Henrique Carlos Fonte Boa Carvalho

Utilização de técnicas de Reconhecimento de Padrões para identificação de ataques de DNS Spoofing em redes locais

Diamantina - MG

09 de março de 2016

Henrique Carlos Fonte Boa Carvalho

Utilização de técnicas de Reconhecimento de Padrões para identificação de ataques de DNS Spoofing em redes locais

Trabalho de Conclusão de Curso apresentada ao Curso de Sistemas de Informação da Universidade Federal dos Vales do Jequitinhonha e Mucuri - UFVJM, como parte dos requisitos exigidos para a obtenção do título de Bacharel em Sistemas de Informação.

Universidade Federal dos Vales do Jequitinhonha e Mucuri – UFVJM

Faculdade de Ciências Exatas

Departamento de Computação

Orientador: Prof. MSc. Eduardo Pelli

Diamantina - MG

09 de março de 2016

Henrique Carlos Fonte Boa Carvalho

Utilização de técnicas de Reconhecimento de Padrões para identificação de ataques de DNS Spoofing em redes locais

Trabalho de Conclusão de Curso apresentada ao Curso de Sistemas de Informação da Universidade Federal dos Vales do Jequitinhonha e Mucuri - UFVJM, como parte dos requisitos exigidos para a obtenção do título de Bacharel em Sistemas de Informação.

Trabalho aprovado. Diamantina - MG, 04 de Março de 2015:

Prof. MSc. Eduardo Pelli
Orientador

Prof Dr. Alessandro Vivas Andrade

Prof Dr. Euler Guimarães Horta

Diamantina - MG
09 de março de 2016

Agradecimentos

Primeiramente gostaria de agradecer aos meus pais Sheila e Fernando, a minha irmã Paloma que sempre me apoiaram e não mediram esforços para que eu chegasse até aqui.

A minha namorada Natalia por sempre ter me auxiliado, me ajudado e ficado ao meu lado.

As minhas primas Cynthia e Lucina por todo apoio e carinho durante todos os momentos em Diamantina. A minha família por todo carinho e apoio.

Aos meus amigos do SIGA, pelo ensinamento e auxílio durante a graduação e serviço.

Aos meus amigos TAE's do Decom/Facet Alan, Oscar, Evandro e Henrique, pela amizade, presteza e auxilio durante a graduação e serviço público.

Aos meus amigos da Sobe e Desce e aos amigos de Diamantina, pela irmandade, bom entendimento, pelos jogos e pelos golos.

Aos professores por me fornecerem conhecimento e caráter durante o processo de formação profissional. Ao meu orientador Eduardo Pelli, pelo suporte no pouco tempo que lhe coube, pelas suas correções e incentivos. Aos professores Alessandro Vivas Andrade e Euler Guimarães Horta pelo tempo, disponibilidade e contribuição neste trabalho.

"O insucesso é apenas uma oportunidade para recomeçar de novo com mais inteligência."
(Henry Ford)

Resumo

A maioria dos dispositivos que estão conectados à Internet usufruem dos serviços de DNS (*Domain Name System*) para resolução de nomes de domínios. A partir da afirmação de que a maioria das redes não restringem o tráfego de pacotes destinados aos serviços de DNS, técnicas de ataques podem ser aplicadas, fazendo com que uma simples requisição de resolução de nome aparentemente normal, possa ocasionar um ataque causando diversos transtornos às vítimas. Os ataques do tipo *Spoofing* consistem em enganar o dispositivo do usuário, fazendo com que o computador identifique o dispositivo do usuário malicioso, de maneira confiável. Desta forma o dispositivo alvo acaba por confiar e receber qualquer tipo de pacote de informações provenientes do atacante. Por se tratar de um tipo de ataque de alta periculosidade, devido à inexistência de pesquisas e técnicas eficientes na sua identificação, procura-se encontrar soluções viáveis para detecção deste tipo de ataque. Este trabalho teve como objetivo a aplicação de técnicas de Reconhecimento de Padrões através da aplicação das técnicas de Seleção de Características como F-Score e Coeficiente de Correlação de Pearson e Classificação de características como as Máquinas de Vetores de Suporte (*SVM*) para detecção de *DNS Spoofing* em redes locais de computadores. Foram obtidos resultados de acurácia média de pelo menos 98,53% $\pm$ 0,93% na detecção na classe de falha da rede, ou seja, quando essas estavam sob ataque de *DNS Spoofing*.

Palavras-chave: redes de computadores; segurança da informação.

Abstract

Most devices that are connected to the Internet use the DNS service (Domain Name System) to resolve domain names. From the statement that most networks do not restrict the packet traffic destined for the DNS service, attacks can be applied, making a simple request apparently normal name resolution, can cause an attack causing several disorders to victims. The type of Spoofing attacks are to trick the device user, causing the computer to identify the malicious user device, reliably. Thus the target device just to trust and receive any type information packet from the attacker. Because it is a type of highly dangerous attack, due to the lack of research and efficient techniques in identification, seeks to find viable solutions to detect this type of attack. This study aimed to apply pattern recognition techniques by applying the features selection techniques as F-score and Correlation Coefficient of Pearson and classification features like Support Vector Machine (SVM) for Spoofing DNS detection in local area networks. Were average results obtained accuracy of at least 98,53% $\pm$ 0,93% in detecting the fault class network, or when these were under DNS spoofing attack.

Keywords: computer networks; information security.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AM	Aprendizado de Máquina
Cert.br	Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil
DNS	Domain Name System
FEBRABAN	Federação Brasileira de Bancos
IP	Internet Protocol
LAN	Local Area Network - Rede Local
MAN	Metropolitan Area Network - Rede Metropolitana
PAN	Personal Area Network - Rede Pessoal
PC	Personal Computer - Computador Pessoal
RNA	Rede Neural Artificial
RTT	Round Trip Time
SO	Sistema Operacional
SVM	Support Vector Machines - Máquinas de Vetores de Suporte
TTL	Time to Live
WAN	Wide Area Network - Rede Geograficamente Distribuida

Sumário

1 Introdução

Tanenbaum (2003) afirma que durante as primeiras duas décadas da existência dos computadores, empresas de médio porte e universidades possuíam apenas 1 ou 2 computadores e que depois de duas décadas haveriam milhões de computadores. Através desta afirmação pode-se perceber que os computadores começam a ser cada vez mais utilizados e mais necessários no cotidiano.

Capron, Johnson e Santos (2006) afirmam que computadores estão cada vez mais presentes na vida da sociedade e são cada vez mais utilizado por apresentar inúmeras vantagens e facilidades nas atividades diárias, onde apresentam as seguintes características:

- Velocidade: Alta capacidade de processamento.
- Confiabilidade: Altamente confiável
- Capacidade de armazenamento: Possibilidade de armazenar uma enorme quantidade de informaçoes.

As primeiras redes de computadores foram concebidas com o intuito de acessar arquivos e programas e compartilhar recursos de outras máquinas. Segundo Tanenbaum (2003) durante os primeiros anos de existência, as redes de computadores foram utilizadas principalmente por pesquisadores universitários e por funcionários de organizações, com a finalidade de enviar mensagens de correio eletrônico, compartilhar impressoras e documentos.

Tanenbaum (2003) e Castells (2003) afirmam que a Internet é um conjunto de redes interligadas. A Internet teve início com o objetivo de mobilizar recursos de pesquisas particularmente universitária para alcançar uma superioridade tecnológica militar, permitindo que vários centros de computadores e diferentes grupos de pesquisa espalhados em diferentes locais pudessem trabalhar e compartilhar informações.

Atualmente a Internet possibilita que seus usuários acessem diversas informações, como dados financeiros, informações pessoais, informações organizacionais entre outras.

O número de usuários trocando informações na Internet está cada vez maior. Segundo ONUBR (2015) o numero de usuários em 15 anos passou de 400 milhões para 3,2 bilhões. Segundo EXAME (2014) 42,5 milhões de brasileiros utilizam celulares para acessarem a Internet, sendo que o número de usuários que acessam a Internet no brasil é de 85,9 milhões. Com esse crescente aumento de usuários utilizando computadores e dispositivos móveis para acessar redes locais e a Internet, maior quantidade de dados e informações são armazenados e transmitidas entre usuários.

Com o aumento de usuários, tráfego de dados e da disseminação da informação

nas redes de computadores, vêm crescendo o número de usuários maliciosos que possuem o objetivo de roubar dados e informações. Eles buscam muitas vezes benefícios próprios ou apenas semear o caos. Esses são conhecidos como *crackers* (VIANNA, 2001; LEMOS, 1999).

A área da computação que possui um intuito de proteger dados, informações e evitar que sejam utilizadas por pessoas sem permissão é a segurança da informação. A segurança da informação é um assunto bastante extenso pois apresenta diversos problemas como roubo de informações, engenharia social dentre outros. Basicamente a segurança da informação trata de impossibilitar que usuários mal-intencionados alterem ou acessem dados e informações destinados a outros usuários. Outro objetivo da segurança da informação é que usuários mal-intencionados não consigam acessar serviços remotos dos quais não estejam autorizados. (TANENBAUM, 2003).

Uma técnica que foi criada para obtenção de dados privilegiados, ou seja, dados que apenas outros usuários possuem permissão para acessar, foi o *Spoofing*. Ele consiste basicamente, em fornecer uma credencial falsa ao destinatário para, através de um conjunto de outros métodos, garantir que consiga obter os dados desejados. Dessa maneira, aparentemente toda conexão realizada foi enviada e recebida corretamente sem que nenhum usuário malicioso tenha acesso. Essa técnica é de alta periculosidade, não apresentando métricas eficazes e defesas definitivas para sua identificação.

As técnicas de Reconhecimento de Padrões podem ser utilizadas com o intuito de identificar padrões, ou seja, elas estudam um grupo de dados e buscam maneiras de agrupá-las e classificá-las através de padrões identificados (DUDA; HART; STORK, 2001).

1.1 Objetivo Geral

Este trabalho teve como objetivo identificar as características mais relevantes em um ataque de *DNS Spoofing* através de técnicas de reconhecimento de padrões.

1.1.1 Objetivos Especificos

1: Aplicar técnicas de seleção de características para selecionar as características mais relevantes;
2: Aplicar um classificador para obter uma elevada taxa de acerto para novos ataques;

2 Revisão de Literatura

2.1 Redes de Computadores

Uma rede de computadores pode ser definida como um conjunto de dois ou mais dispositivos (também nomeados como nós ou *hosts*) que utilizam diversas regras (protocolos) para compartilhar recursos. Esses nós devem estar interligados através de uma estrutura cabeada ou sem fio podendo ser: cabo par trançado, cabo de cobre, fibra ótica, ondas de rádio, via satélite entre outros.

As redes de computadores podem ser divididas de acordo com a distância entre seus nós apresentado na Tabela 1 (ROSS, 2008).

Tabela 1 – Classificação das Redes de Computadores

Distância	Localização dos processadores	Nome
1 m	Metro Quadrado	*PAN*
10 m	Quarto	*LAN*
100 m	Prédio	*LAN*
1 km	Campus	*LAN*
10 km	Cidade	*MAN*
100 km	País	*WAN*
1000 km	Continente	*WAN*
10.000 km	Planeta	*WAN*

Fonte: Adaptado de Tanenbaum (2003)

As Redes Locais (*LAN*), são redes que conectam computadores numa mesma sala, edifício ou campus universitário com até alguns quilômetros de extensão. São amplamente usadas para conectar computadores pessoais e estações de trabalho em escritórios e instalações industriais, permitindo a troca de recursos e dados entre os nós. As Redes Metropolitanas (*MAN*), abrangem computadores interligados em uma cidade, as vezes, interligam até computadores de cidades vizinhas próximas. As Redes Geograficamente Distribuida (*WAN*), abrangem computadores conectados em uma grande área geográfica, como cidades, estados, países e continentes (TANENBAUM, 2003).

Quando um dispositivo conecta a uma rede ele recebe um endereço *IP*, este é um endereço virtual, basicamente é a identificação do dispositivo na rede que ele esta conectado. O *IP* pode ser atribuido automaticamente pelo dispositivo de conexão a rede ou pelo administrador do sistema.

Quando um dispositivo deseja enviar informações para outro dispositivo pertencente a mesma rede de computadores, é necessário que ele o endereço *MAC* do dispositivo o

qual deseja se comunicar. A maneira para realizar identificar o endereço *MAC* de um dispositivo é através do protocolo *ARP* (CANTÚ, 2003).

O protocolo *ARP*, definido na RF826, envia pacotes para todas as máquinas conectadas à rede, perguntando qual dispositivo possui o endereço *IP* e qual o endereço *MAC*, sendo que apenas o dispositivo alvo responde indicando seu endereço *MAC* (TANENBAUM, 2003).

O endereço *MAC* é o endereço físico associado ao dispositivo, é um endereço gravado em *hardware*, é um endereço único, cada dispositivo possúi um endereço *MAC* por componente que possa acessar alguma rede (ROSS, 2008).

Conforme o aumento de *hosts* na Internet, tornou-se dificil localizar *hosts*, criou-se assim o *DNS*, cujo o objetivo é mapear e traduzir nomes de de domínios em endereços *IP*. Isso facilita pois não ocorre a necessidade de decorar o endereço de *IP* de cada *host*, mas os nomes de domínio. O *DNS* é um sistema de bancos de dados distribuídos, capaz de armazenar informações referentes a nomes de domínios (TANENBAUM, 2003).

Para resolução dos nomes de domínio, o *DNS* realiza uma busca em um banco de dados distríbuido, armazenados em sistemas independentes, chamados de resolvedores de nomes, os quais realizam a tradução do nome para o endereço de *IP*.

2.1.1 Segurança da Informação

Com a utilização dos computadores inicialmente voltados apenas para transferência de dados e compartilhamento de recursos a segurança das redes nunca precisou de maiores cuidados. Porém, a utilização de computadores tem se tornado cada vez mais necessária e imprescindível para realização de diversas tarefas, possibilitando que seus usuários como cidadãos comuns e organizações empreguem as redes para executar operações financeiras, adquirir conhecimento e armazenar informações. Apesar disso as redes de computadores vem se tornando um problema (TANENBAUM, 2003).

Diversas organizações sofrem ameaças constantes em seus ativos, o que poderia representar prejuízo de milhões de dólares. Ativos são considerados qualquer elemento que tenha valor para a organização (ISO/IEC17799, 2005). Os ativos estão agrupados nas seguintes categorias (LYRA, 2008; ISO/IEC17799, 2005):

- Informação
- *Software*
- Físico
- Serviços
- Pessoas
- Intangíveis

As informações, atualmente, são os objetos de maior valor para as empresa, mediante a necessidade de proteger os ativos das organizações e principalmente as informações presentes em diferentes formas, podendo ser impressas em papel, armazenados em dispositivos de armazenamento ou nas memórias das pessoas.

A segurança da informação se refere à proteção existente sobre as informações de uma determinada empresa ou pessoa, ou seja, aplica-se tanto para as informações empresariais quando as pessoais. A informação é um ativo que, como qualquer outro ativo importante, é essencial para os negócios de uma organização e consequentemente necessita ser adequadamente protegida (ISO/IEC17799, 2005).

A segurança da informação apresenta três propriedades básicas (Figura 1) (LYRA, 2008; ISO/IEC17799, 2005):

- Confidencialidade
- Integridade
- Disponibilidade

Figura 1 – Pilares da Segurança da Informação
Fonte: www.aes.inf.br

Segundo Lyra (2008) a confidencialidade é definida basicamente como capacidade de um sistema permitir que uma determinada informação será acessada somente por pessoas autorizadas. A integridade refere-se a informação estar correta, neste caso, ser verdadeira e não adulterada ou corrompida. Disponibilidade refere-se a informação estar disponível para ser acessada no momento necessário.

Além das três propriedades básicas a segurança da informação possui outras propriedades (LYRA, 2008):

- Autenticação
- Não-Repudio
- Legalidade

- Privacidade
- Auditoria

Segundo Lyra (2008) a autenticação busca garantir que um usuário é de fato quem alega ser; não-repúdio é a capacidade de um sistema em garantir que um determinado usuário executar uma ação; legalidade é a possibilidade de garantir que o sistema esteja de acordo com a legislação vigente; privacidade é a capacidade de um sistema manter o usuário anônimo, realizando ações sem que seja identificado; e auditoria é a capacidade do sistema auditar tudo que for realizado, detectando tentativas de ataque ou fraudes.

Com o crescimento da utilização das redes de computadores para diversos fins, aumenta-se o número de ataques sofridos por usuários e empresas e também a variedade de ataques. Diversos ataques não podem ser contabilizados, pois alguns passam desapercebidos pelos alvos. Segundo CERT.br (2015) o número de ataques vem aumentando ao longo dos anos, de 2013 a 2014 triplicou o número de incidentes reportados, conforme apresentado na Figura 2.

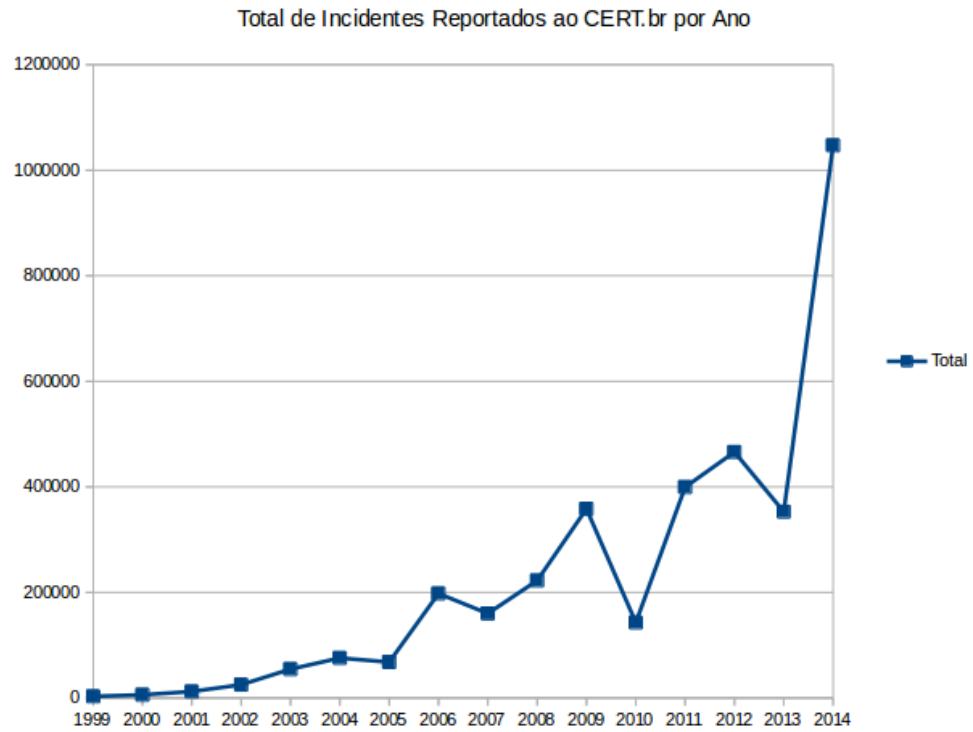

Figura 2 – Incidentes reportados ao CERT.br.
Fonte: www.cert.br

Visando a obter benefícios, alguns usuários tendem a manipular as informações que são transmitidas na rede, desta maneira prejudicando os usuários que receberam dados incorretos durante a troca de pacotes pela rede. Uma técnica utilizada por *crackers* para obter informações privilegiadas e/ou alterar dados é conhecido por *Spoofing*.

Os *crackers* são os usuários de computadores que invadem sistemas e violam a privacidade alheia. Possuem objetivos financeiros ou caóticos, e possuem o ideal de romper

com a sociedade e sabotar ao máximo os grandes sistemas de computadores. (VIANNA, 2001; LEMOS, 1999).

2.1.2 Kali Linux

Kali Linux é uma distribuição baseada no Debian GNU/Linux, este sistema operacional possui o foco na perícia forense computacional e em testes de penetração. Ele é o sucessor da distribuição Backtrack (PRITCHETT; SMET, 2013).

A forense computacional é a ciência que estuda a aquisição e análise de dados livres de distorções ou alterações de maneira que seja possível reconstruir os dados ou o que aconteceu no passado no sistema.

Teste de penetração são métodos utilizados por profissionais de tecnologia da informação para avaliar a segurança de sistemas de computadores ou de redes, através da simulação de um ataque de uma fonte maliciosa, tentando identificar possíveis falhas e erros no sistema com o intuito de prevenir que um futuro ataque realizado por um usuário malicioso obtenha informações e dados confidenciais (GIAVAROTO; SANTOS, 2013).

O Kali Linux, por ser uma distribuição focada para a segurança da informação, apresenta a maioria das aplicações mais conhecidas e utilizadas para a auditoria de sistemas pré-instalados, dentre os mais conhecidos estão o NMAP, Wireshark entre outros.

O Nmap é uma ferramenta utilizada para exploração da rede e auditoria de sistemas. Foi desenvolvido para realizar uma rápida análise sobre os dispositivos da rede.Ele utiliza pacotes para determinar quais dispositivos estão conectados na rede, quais serviços estão disponíveis, qual SO está utilizando entre outras características Lyon e Fyodor (2009).

O Wireshark é uma ferramenta para analisar pacotes que circulam na rede. Ele tenta capturar os pacotes da rede e mostrar de forma mais detalhada possível (LAMPING; WARNICKE, 2004).

2.1.3 Spoofing

A maioria das redes que estão interligadas à Internet utilizam os serviços de DNS para resolução de nomes de domínios, ou seja, a conversão do nome fornecido pelo aplicativo para o endereço de IP do servidor. Mesmo com todos os sistemas defensivos e proteções criadas para a segurança, as redes não podem ignorar estes serviços, possibilitando a troca de pacotes que se destinam a aplicação DNS. Consequentemente a maioria das redes não restringem o tráfego de pacotes destinados a esses serviços. Desta maneira é possível que técnicas de ataque mais sofisticadas como *DNS Spoofing* possam ser aplicadas, fazendo com que uma simples requisição de resolução possa sofrer uma alteração vinda de um ataque de rede e retornar com um valor incorreto.

O *Spoofing* é uma técnica que consiste em fornecer uma identidade falsa, que por sua vez é considerada confiável aos usuários alvos que estão conectados à rede. Essa técnica consegue por meio da identidade falsa modificar e enviar pacotes falsos ou ilegais aos alvos. O *Spoofing* possui algumas variações no enfoque do ataque, podendo ser usado basicamente: o *IP Spoofing*, o *DNS Spoofing* e também o *ARP Spoofing*.

O *IP Spoofing* consiste em enganar o dispositivo do usuário alvo, fazendo com que o computador alvo identifique o dispositivo do usuário malicioso de maneira confiável. Desta forma o dispositivo alvo acaba por confiar e receber qualquer tipo de pacote de informações provenientes do atacante. Com a utilização é possível que o atacante invada o computador do usuário alvo, receba os pacotes provenientes da comunicação entre os dispositivos e cometa um *hijacking*, que é um sequestro da seção do usuário. Tanase (2003) define o IP Spoofing da seguinte maneira:

> IP spoofing é uma das formas mais comuns de se camuflar on-line. No IP spoofing, o atacante ganha um acesso não autorizado a um computador ou a uma rede por fazer parecer que a mensagem maliciosa foi enviada por uma máquina confiável pois falsificou o endereço de IP desta máquina.

O *ARP Spoofing* é um ataque cujo o objetivo é alterar a resposta *ARP* enviada a uma requisição original através de uma resposta falsa. Enviando uma resposta falsa, o roteador pode expedir dados destinados ao computador da vítima para o computador do usuário malicioso e este redireciona os dados para a vítima. Se for bem sucedido o computador da vítima não tem ideia que o redirecionamento das informações esta ocorrendo.

O *DNS Spoofing* consiste em enganar o dispositivo do usuário alvo, conseguindo redirecionar as requisições realizadas ao servidor de DNS dos aplicativos Web que o dispositivo alvo deseja acessar, propiciando um redirecionamento para página definida pelo usuário malicioso. O *DNS Spoofing* consiste de basicamente 4 etapas:
1º O usuário alvo realiza uma requisição ao servidor de DNS;
2º O invasor retorna um IP falso para a requisição do usuário alvo antes de chegar a resposta do servidor de DNS com o IP verdadeiro;
3º A resposta do servidor de DNS com o IP verdadeiro é descartada pois o alvo ja recebeu uma resposta anterior;
4º O usuário acessa o endereço fornecido pelo invasor;
Na Figura 3 pode-se observar um ataque do tipo *DNS Spoofing*. Pode-se observar a falsificação de uma requisição, onde, a cada pacote enviado estará geralmente associada uma resposta do servidor de DNS e essa falsa resposta será enviada para a vítima.

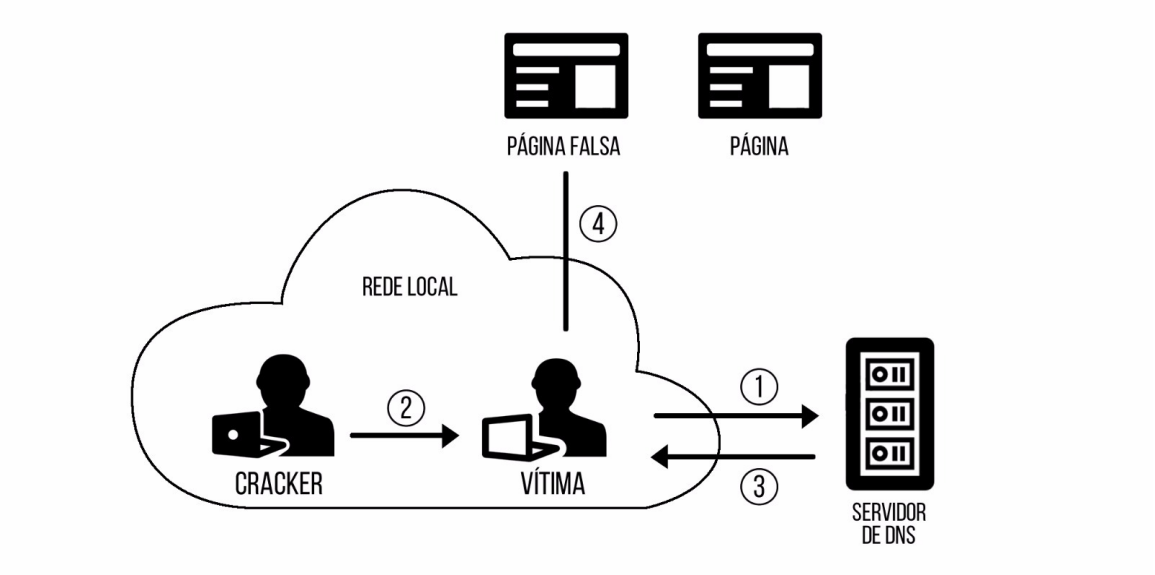

Figura 3 – Esquema representativo de um ataque de *DNS Spoofing*.

Existem três mecanismos utilizados atualmente para bloqueio de pacotes falsos provenientes de *IP Spoofing*, são eles: - filtragem de pacotes que chegam e saem da rede origem (Ingresso / Egresso); -Rastreamento inverso (*Trace back*); e tentativas de descarte de pacotes no destino. Bremler-Barr e Levy (2004) propuseram um método de prevenção do *Ip Spoofing (Spoofing Prevention Method - SPM)*, com precisão de 99%. Tal método se baseia na análise de autenticidade do endereço da fonte de pacotes por roteadores próximos aos dispositivos destino destes pacotes. A avaliação é dada de uma chave única associada a um registro temporal para cada conexão de rede com origem e destino definidos.

Essas soluções não se aplicam ao *DNS Spoofing*, pois esses atuam nas redes locais, não sendo possível a filtragem dos pacotes por um roteador ou *Firewall*. Portanto, esse tipo de ataque é considerado de alta periculosidade, pois, não existem técnicas eficientes para identificação, sendo necessário o estudo de todas as variáveis relacionadas ao ataque, a fim de se tentar criar estratégias de defesa

Ferguson (2000) cita ataques de *DNS Spoofing* em dispositivos localizados em locais públicos, para obtenção de informação sobre cartões de crédito de bancos e afirma que, para prevenir um ataque deste tipo, deve-se atentar ao contexto e às decisões de segurança.

De acordo com Bremler-Barr e Levy (2005) em 2005 foram contabilizados pelo menos 4 mil ataques de *Spoofing* a cada semana na *Internet*, isso devido a facilidade com que o ataque é gerado. Atualmente, com um nível de acesso muito maior, pode-se presumir que o nível de ataque é bastante superior.

2.2 Inteligência Computacional

Segundo Iyoda (2000) a inteligência computacional compreende paradigmas computacionais que procuram desenvolver sistemas que apresentam alguma forma de inteligência similar à exibida por determinados sistemas biológicos, sendo sistemas que produzam algum tipo de comportamento encontrados em sistemas naturais.

Algumas técnicas de inteligencia computacional e sua inspiração (Tabela 2).

Tabela 2 – Tabela de Representação de Técnicas da Inteligência Computacional

Técnica Computacional	Inspiração na Natureza
Redes Neurais	Neurônios Biológicos
Computação Evolucionária	Evolução Biológica
Lógica Fuzzy	Processamento linguístico
Sistemas Especialistas	Processo de Inferência

Bezdek (1994) sugere que um sistema é computacionalmente inteligente quando ele exibe ou começa a exibir:

- Reconhecimento de padrões.
- Adaptabilidade computacional.
- Tolerância computacional a falhas.
- Velocidade de processamento comparável a de processos cognitivos humanos.
- Taxas de erro que se aproximam do desempenho humano.

Duch (2007) sugere que os sistemas inteligentes devem exibir:

- Autonomia
- Aprendizagem
- Raciocínio

Eberhart, Simpson e Dobbins (1996) sugerem que sistemas inteligentes são capazes de:

- Aprender
- Tratar novas situações utilizando
  - Raciocínio
  - Generalização
  - Assiciação
  - Abstração
  - Descoberta

2.2.1 Aprendizado de Máquina

As técnicas de Aprendizado de Máquina utilizam um princípio de inferência denominado indução. Através deste princípio é possível obter conclusões genéricas a partir de um determinado conjunto de dados fornecidos como exemplos. O aprendizado indutivo é dividido em dois tipos, sendo eles, o aprendizado supervisionado e o aprendizado não-supervisionado.

No aprendizado supervisionado é fornecido ao algoritmo de aprendizado, um conjunto de exemplos de treinamento para os quais o resultado já é conhecido, sabendo a qual classe os dados pertencem, ou seja, o aprendizado supervisionado necessita de um "professor externo", para apresentar conhecimento ao algoritmo através dos dados conhecidos. O algoritmo, a partir do conhecimento fornecido pelos exemplos, consegue extrair as informações com o objetivo de produzir os resultados corretos para novas entradas que não foram fornecidas previamente.

No aprendizado não-supervisionado não são fornecidos nenhum tipo de dados previamente ao algoritmo, ou seja, não há a presença de um "professor". Desta forma, o algoritmo não possui nenhum conhecimento sobre os dados que serão fornecidos. O algoritmo analisa os dados fornecidos e tenta agrupar de alguma maneira (LORENA; CARVALHO, 2007). Esse tipo de aprendizagem é utilizada principalmente quando deseja-se identificar padrões ou tendências que possam influenciar nos dados estudados, apresentando assim um maior entendimento (SOUTO et al., 2003).

Para Lorena e Carvalho (2007) as técnicas de Aprendizado de Máquina devem possuir alguns requisitos básicos, dentre eles, ser capazes de lidar com ruídos, que são dados imperfeitos. Diversos conjuntos de dados possuem dados imperfeitos, como, por exemplo, a presença de dados com rótulos e/ou atributos incorretos. Deseja-se e minimizar a influência de outliers durante o processo, sendo esses, dados muito discrepantes que raramente ocorrem no conjunto, podendo ser dados incorretos ou muito particulares.

O principal objetivo ao utilizar técnicas de AM é o de generalização de um classificador, que pode ser definido como a capacidade de predizer novos dados fornecidos para "teste" corretamente em suas classes. Durante a aplicação de um classificador, pode ocorrer um *overfitting* ou *underfitting*. *Overfitting* ocorre quando o classificador se especializa no conjunto de dados fornecido para treinamento, apresentando um alto índice de acerto para o conjunto de treinamento e um baixo índice de acerto para os dados de teste. *Underfitting* ocorre quando ele apresenta uma baixa taxa de acerto no conjunto de treinamento.

Os conceitos referentes à geração de um classificador são apresentados de forma simplificada na Figura 4.

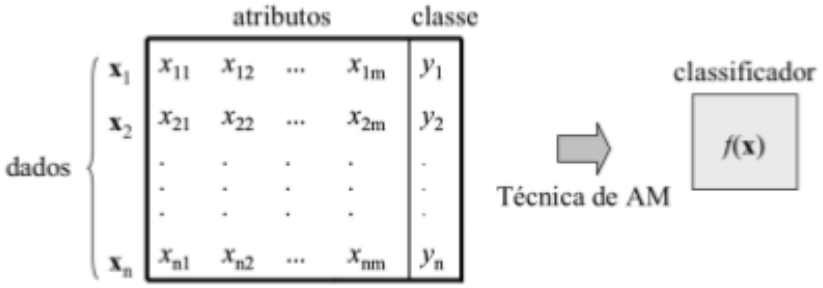

Figura 4 – Indução de um Classificador
Fonte: Lorena e Carvalho (2007)

2.2.2 Reconhecimento de Padrões

De acordo com Bianchi (2006), Reconhecimento de Padrões é a área que estuda como as máquinas podem observar dados, aprender e distinguir os dados fornecidos para então classificar e categorizar os dados em classes através de características comuns.

Existem duas formas de reconhecer e classificar um padrão: o reconhecimento supervisionado, quando os dados fornecidos já foram identificados com uma classe fornecida ou o reconhecimento não supervisionado, quando não há informações fornecidas previamente por um agente externo. Desta forma o sistema não apresenta informações conhecidas como dados de entrada (LORENA; CARVALHO, 2007).

Na primeira etapa define-se como e quais os dados serão obtidos para posteriormente serem analisados, nesta etapa ocorre a extração de características, além de ocorrer o pré-processamento dos dados, que consiste em aplicar determinadas técnicas visando a obter ajustes na base de dados. Desta maneira é possível aumentar ou diminuir algumas características.

Na segunda etapa são utilizadas técnicas de seleção de características, para identificadar quais características da base de dados são mais relevantes para posteriormente classificar e agrupar os dados em classes. Para tarefa de seleção pode-se citar os métodos *F-Score* e Coeficiente de Correlação de Pearson.

Segundo Duda, Hart e Stork (2001), o método *F-Score* (*Fisher Score*) para seleção de característica disponibiliza uma medida dada pela distância entre as médias das distribuições de duas classes (C_1 e C_2) em relação às suas variâncias. Quanto maior é o valor da classe calculado pelo *F-Score*, mais discriminativa e relevante é a característica para o experimento. A métrica é definida pela Equação 2.1 (GU; LI; HAN, 2012):

$$F(g) = \frac{\sum_{k=1}^{n} n_k(\mu_k^j - \mu^j)^2}{(\sigma^j)^2} \tag{2.1}$$

Onde:

- $(\sigma^j)^2 = \sum_{i=1}^{n} n_k(\sigma_k^j)^2$
- *F(g)* é a função que calcula o valor *F-score* para a característica g;

O Coeficiente de Correlação de *Pearson* é outra metodologia para rankear características relevantes. Ela mede o grau de relação da distribuição de duas classes. O cálculo desse coeficiente é definido pela Equação 2.2 (DUDA; HART; STORK, 2001):

$$\rho_j = \frac{\frac{1}{n-1}\sum_{i=0}^{n}(x_{ij} - \bar{x}) \times (y_{ij} - \bar{y})}{\sigma_{xj} \times \sigma_y} \tag{2.2}$$

Onde:

- -1 $\leq \rho_j \leq$ 1;
- Se $\rho_j = 1$, existe total relação positiva entre as distribuições;
- Se $\rho_j = -1$, existe total relação negativa entre as distribuições;
- Se $\rho_j = 0$, as distribuições não possuem relação.

Na terceira etapa, é realizada a classificação, onde consiste em utilizar as características que foram obtidas nas etapas anteriores para conseguir diferenciar novos dados fornecidos em classes.

O classificador SVM foi desenvolvido baseado nas ideias originadas na teoria de aprendizagem estatística de Vapnik e Vapnik (1998). Segundo Lorena e Carvalho (2007) a teoria de aprendizagem estatística estabelece uma série de métricas que devem ser seguidas com o intuito de obter um classificador com boa generalização, definida como a sua capacidade de identificar corretamente a classe de novos dados do mesmo domínio em que o aprendizado ocorreu.

As Máquinas de Vetores de Suporte constituem uma técnica de aprendizado que vem atraído uma grande atenção do público acadêmico, especialmente a atenção da comunidade de Aprendizado de Máquina. Os resultados da aplicação de SVMs são muitas vezes melhores do que os obtidos por outras técnicas de AM, como as RNAs. As SVMs apresentam uma boa generalização, podendo ser utilizadas em diversasas áreas do conhecimento, trabalhos de destaque podem ser encontrados na categorização de textos segundo Joachims (2002), na análise de imagens segundo Kim et al. (2002) e Pontil e Verri (1998) e em Bioinformática de acordo com Noble et al. (2004) e Scholkopf, Guyon e Weston (2003).

As SVMs vêm sendo difundida através da comunidade acadêmica não apenas por apresentar melhores resultados que as Redes Neurais Artificiais, segundo Soares (2008), outras características que vem contribuindo para que seu uso seja amplamente difundido são:

- Boa capacidade de generalização
- Robustez diante de objetos de dimensões elevadas
- Convexidade da função objetivo, ou seja, possui apenas um mínimo global

- Capacidade de lidar com dados ruidosos
- Uma base teórica bem estabelecida na Matemática e Estatística

Segundo Chaves (2006) a SVM pode ser descrita da seguinte maneira: determinada base da dados com duas classes e um contjunto de dados pentencentes a uma dessas classes, a SVM encontra o hiperplano que separa os dados de ambas as classes (Figura 5). O hiperplano deve cometer poucos erros marginais, minimizando assim o erro sobre os dados de teste e de treinamento, respectivamente. Desta forma o hiperplano é denominado ótimo (LORENA; CARVALHO, 2007), (SMOLA; SCHÖLKOPF, 1998).

O hiperplano é determinado por uma amostra da base de dados, nomeado vetores de suporte. O conceito por traz do SVM é a maximização da margem, ou seja, maximizar a distância dos dados de treinamento.

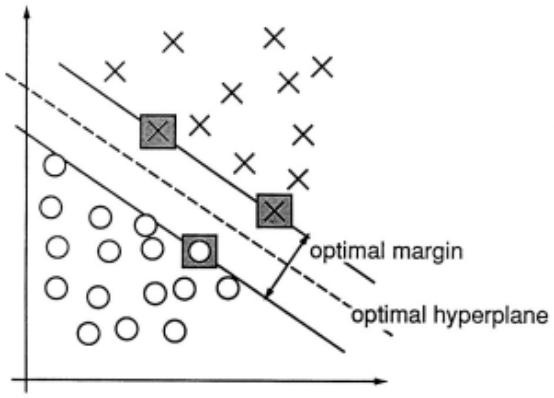

Figura 5 – Distribuição do hiperplano.
Fonte: Cortes e Vapnik (1995)

A decisão pelo *SVM* é dado pela (Equacao 2.3) (CORTES; VAPNIK, 1995).

$$W(\alpha) = \sum_{i}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \tag{2.3}$$

$$\text{Sujeito a} = \sum_{i=1}^{N} y_i \alpha_i; \forall_{i=1}^{N} : 0 \leq \alpha_i \leq C \tag{2.4}$$

Onde:

- $W(\alpha)$ é o vetor de pesos;
- $K(x_i, x_j)$ é o Kernel utilizado;
- α é o multiplicador de Lagrange;
- C é especificado pelo usuário;
- X é o vetor de características;
- Y é o rótulo da classe;

O parâmetro C é a relação entre a complexidade do algoritmo e o número de amostras de treinamento incorretos (HORTA, 2008). De acordo com (SEMOLINI, 2002) pode ser visto como um parâmetro de penalização .

As funçoes *kernel* é o que diferenciam um modelo de SVM de outro, ou seja, a função realiza o mapeamento dos dados (HORTA et al., 2011). Segundo Lima (2002) o *kernel* realiza implicitamente o mapeamento nos dados de características e depois usa um algoritmo para classificar o espaço.

O *SVM* possui 4 funçoes *kernel*:

- Linear = $(x' * y)$
- Polinomial = $(\gamma * x' * v + 1)^d$
- Radial Basis = $\exp(-\gamma * |u - v|^2)$
- Sigmoidal = $\tanh(\gamma * u' * v)$

3 Material e Métodos

Para realização desta pesquisa, foram utilizados dois computadores, o primeiro computador utilizado era um notebook da marca Fácil, composto por um processador Intel Dual Core de 1.8 Ghz, 4 Gb de Memória RAM DDR II, 320 Gb de Disco Rígido e equipado com o S.O. Kali Linux 1.9 e o segundo computador foi um desktop, composto por um processador Intel I5 de 3.1 Ghz, 8 Gb de Memória RAM DDR III, 500 Gb de Disco Rígido e equipado com o S.O. Ubuntu 15.04.

Inicialmente foi utilizado o Notebook contendo o SO Kali Linux e o aplicativo desenvolvido para Linux já pré-instalado no SO denominado *Ettercap* para simular um ataque de *DNS Spoofing* em um ambiente controlado para realizar a pesquisa, tendo sido utilizado especificamente com este intuito.

Posteriormente a simulação do ataque, foi utilizado o Desktop contendo Ubuntu para a realização das etapas posteriores, sendo essas, extração das características influenciadas pela simulação, realização da coleta da base de dados, pré-processamento das características, seleção das características e classificação das características aplicando a *SVM*, desta maneira, este computador ficou com os processamentos mais pesados.

Após a identificação das características relevantes para o estudo, passou-se para a etapa de coleta de dados que ocorreu em uma rede local durante um dia rotineiro de utilização da rede. Durante a coleta estavam conectados na rede 7 dispositivos. Para a coleta de dados, foi utilizado um *script* baseado na linguagem *Bash* desenvolvido especificamente para este projeto, sendo obtidos e armazenados os dados das características identificadas durante uma simulação de um ataque e durante uma utilização sem nenhuma ameaça na rede.

Após a coleta de dados foi utilizada a linguagem R em conjunto com o R Studio, uma IDE para desenvolvimento desta linguagem, para realizar o desenvolvimento do restante dos *scripts* para serem utilizados nesse projeto. Foi criado o *script* para realizar o pré-processamento, a seleção, a classificação e a avaliação das características.

Depois de criado os scripts foi realizado o pré-processamento das características, com o intuito colocá-las em um padrão, para posteriormente utilizá-las, livrando-os de ruídos. Após o pré-processamento aumentou-se o número de características, passando de 4 para 8. Depois de obtidos os novos dados, foi aplicados as técnicas de Seleção de Características, *F-Score* e *Coeficiente de Correlação de Pearson*.

Através das técnicas de seleção, foi obtido um ranqueamento dos dados, onde foram obtidas as características que apresentavam maior relevância para o estudo, e

posteriormente foi realizado a etapa de Classificação de Características, onde foi utilizado o Classificador *SVM*. Para a utilização do classificador foi usado uma parte dos dados para realizar o seu treinamento e o resto dos dados para realizar o teste. Após a aplicação do classificador era necessário avaliar a taxa de acerto do classificador para nova entrada dos dados, onde foi realizada a avaliação dos resultados.

3.1 Extração de características

As características identificadas através da simulação foram:

- Saltos: Basicamente é a quantidade de saltos que o pacote demora para chegar ao destino, ou seja, a quantidade de roteadores, servidores que um pacote passa até chegar ao servidor de destino. Também conhecido na literatura por TTL (*Time To Live*), este foi identificado através do comando *traceroute*, que envia pequenos pacotes, por cada nó da rede até que o pacote chegue ao destino, e posteriormente ele retorna para a máquina solicitante, para obter uma grande quantidade de dados é dispendido uma enorme quantidade de horas, pois ele precisa aguardar uma resposta de um nó para enviar o novo pacote para o próximo nó da rede (Figura 6).

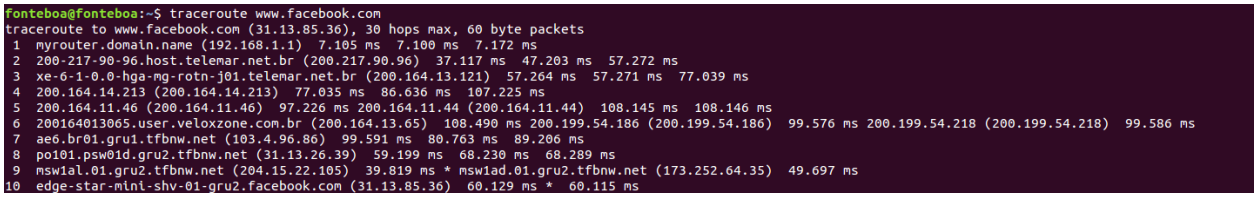

Figura 6 – Comando *traceroute* aplicado ao site www.facebook.com.

- Tempo de Resposta: Basicamente é o tempo que um pacote leva chegar ao servidor de destino e retornar ao requisitante. Também conhecido na literatura como RTT *(Round Trip Time)*, foi obtido através do comando *ping* (Figura 7).
- Erros: Foi definido que era um "erro"quando o pacote passava do tempo máximo permitido pelo comando *ping*, onde, após esse tempo ele era descartado, desta maneira, se ele fosse descartado o valor do tempo de resposta era 2000 ms e erros era 1 (Figura 7).

```
fonteboa@fonteboa:~$ ping www.facebook.com
PING star-mini.c10r.facebook.com (31.13.85.36) 56(84) bytes of data.
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=1 ttl=86 time=47.5 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=2 ttl=85 time=72.8 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=3 ttl=85 time=95.2 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=4 ttl=86 time=34.0 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=5 ttl=86 time=62.8 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=6 ttl=86 time=100 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=7 ttl=86 time=39.0 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=8 ttl=86 time=69.1 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=9 ttl=85 time=117 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=11 ttl=86 time=106 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=12 ttl=86 time=35.0 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=13 ttl=85 time=53.8 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=14 ttl=86 time=93.1 ms
64 bytes from edge-star-mini-shv-01-gru2.facebook.com (31.13.85.36): icmp_seq=15 ttl=86 time=42.5 ms
^C
--- star-mini.c10r.facebook.com ping statistics ---
15 packets transmitted, 14 received, 6% packet loss, time 14027ms
rtt min/avg/max/mdev = 34.024/69.323/117.315/27.741 ms
```

Figura 7 – Comando *traceroute* aplicado ao site www.facebook.com.

- Velocidade: Foi obtida através do comando *wget*, ele faz o download de um arquivo e identifica a velocidade instantânea do *download* (Figura 8).

```
fonteboa@fonteboa:~$ wget /temp/speedtest_data/ http://ftp.icm.edu.pl/pub/Linux/opensuse/distribution/13.2/iso/openSUSE-13.2-DVD-x86_64.iso
/temp/speedtest_data/: O esquema está faltando.
--2016-01-20 12:44:41--  http://ftp.icm.edu.pl/pub/Linux/opensuse/distribution/13.2/iso/openSUSE-13.2-DVD-x86_64.iso
Resolvendo ftp.icm.edu.pl (ftp.icm.edu.pl)... 193.219.28.2
Conectando-se a ftp.icm.edu.pl (ftp.icm.edu.pl)|193.219.28.2|:80... conectado.
A requisição HTTP foi enviada, aguardando resposta... 200 OK
Tamanho: 4678746112 (4,4G) [application/octet-stream]
Salvando em: “openSUSE-13.2-DVD-x86_64.iso”

openSUSE-13.2-DVD-x86_64.iso                0%[                                                                                ] 211,98K  43,9KB/s   eta 28h 55m
```

Figura 8 – Comando *wget*.

3.2 Coleta de Dados

Após as etapas de simulação e identificação das características envolvidas a um ataque de *DNS Spoofing*, foi desenvolvido um *script* na linguagem *Bash* para a obtenção dos dados. O script foi aplicado durante dois momentos: durante uma ocorrência de ataque, onde foram coletados 2500 dados de cada característica e aplicado durante nenhuma ocorrência de ataque na rede, sendo obtidos também 2500 dados, onde foi armazenando os dados referente as características: Velocidade, Tempo de Resposta, Saltos, Erros e, por ultimo, foi acrescentada uma característica denominada Classe, que é responsável por realizar o rótulo correto de cada amostra. O rótulo “1” foi definido para os dados que estavam sofrendo um ataque de *DNS spoofing* e “2” para os que não estava sofrendo nenhum tipo de ataque.

3.3 Pré-processamento dos Dados

De posse dos 5 mil dados obtidos através do *script*, foi necessário realizar o pré-processamento dos dados, desta forma foram obtidos dados mais homogêneos sem grande dispersão por possíveis erros.

Para a realização desta etapa foi desenvolvido um script em Linguagem R. Cada característica foi transformada em sua média e seu desvio padrão. Para obtenção desses dados o *script* armazenava 10 dados de cada característica e gerava sua média e seu desvio padrão. Após o cálculo realizado nas características foi obtido uma nova base de dados, contendo 8 características e 500 repetições de cada.

As novas características foram: Média de Saltos, Desvio padrão de Saltos, Média do Tempo de Resposta, Desvio padrão do Tempo de Resposta, Média da Velocidade, Desvio padrão da Velocidade, Média de Erros, Desvio padrão de Erros. A etapa é apresentada na Figura 9.

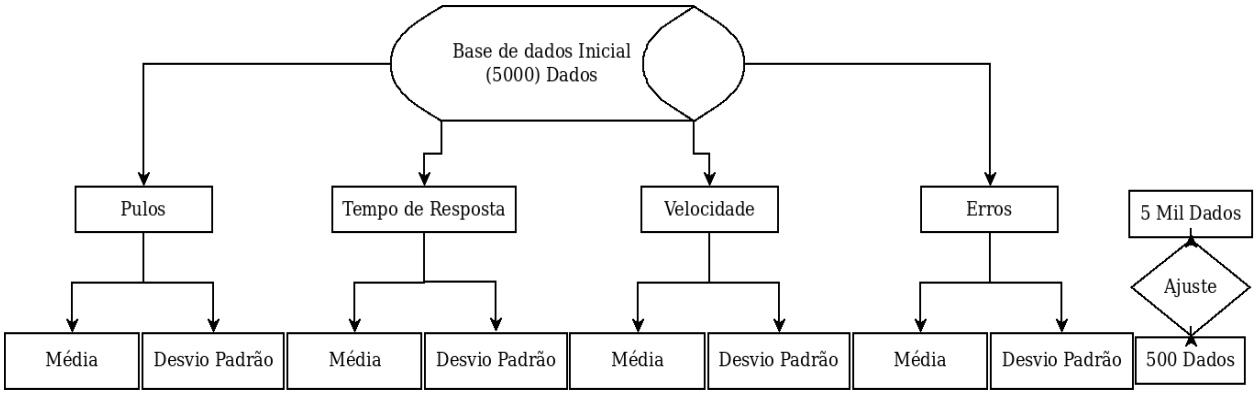

Figura 9 – Ajuste dos dados realizado na base de dados original

3.4 Seleção de Características

Após o ajuste realizado na base de dados contendo os dados originais, foram aplicadas as técnicas de seleção de características como *F-score* e o *Coeficiente de Correlação de Pearson*, com o intuito de identificar as características que possuem maior discrepância no conjunto de dados sendo identificadas quais são as mais relevantes. Durante a aplicação das técnicas foi utilizada a totalidade dos dados, ou seja, foram utilizadas todas as 8 características e suas 500 repetições.

Com a aplicação das técnicas foi obtido uma tabela contendo o ranqueamento dos dados ordenados pelo valor de cada uma, onde são apresentadas as características relevantes para o estudo. Após a aplicação da seleção foi realizada uma alteração nas Classes dos dados, ou seja, foi alterado o rótulo dos dados, para o resultado ser mais claro e mais conciso, desta forma, os dados anteriores que possuíam o rótulo de “1” passaram para “S”, neste caso que estava sendo realizado um ataque de Spoofing e os que possuiam o rótulo de “2” passaram para “N”, onde não ocorria nenhum tipo de ataque.

3.5 Classificação de Características

Em posse dos dados e do resultado de quais são as características mais relevantes, foi iniciado o desenvolvimento de um *script* em Linguagem R para a aplicação do classificador *SVM*. O aprendizado escolhido para esse classificador foi o supervisionado e, neste caso, era necessário fornecer ao *SVM* os dados para o seu "treinamento" e "teste".

A base de dados fornecida para o "treinamento" continham as características e apresentavam o rótulo de cada dado, ou seja, continha a qual classe os dados pertencem. A base de "treinamento" possui a função de fornecer conhecimento para o classificador. Para o "teste" foi utilizado o restante dos dados, sem a utilização dos rótulos. Eles não estão contidos na base de "treinamento", desta maneira, a base de "teste" era uma base de dados "nova", que não foram fornecidas previamente ao classificador. A base de "teste" é utilizada para verificar qual o percentual de acerto do classificador para novos dados, assim sendo, ela é utilizada para verificar se o classificador consegue prever corretamente a qual classe os dados desconhecidos pertencem.

Com o intuito de obter dados imparciais e não tendenciosos durante a classificação e posteriormente a avaliação dos resultados da taxa de acerto do classificador SVM, foi aplicado o classificador 10 vezes para obtenção final dos resultados de predição correta para novos dados. Para cada repetição da SVM foram utilizados dados diferentes dos anteriores, para isso os dados eram aleatórios para cada repetição, desta maneira, a base de dados de treinamento que possuía 70% e teste que possuía 30% dos dados e a cada repetição eram diferentes. Toda aplicação da SVM foi através da utilização do *Kernel* "linear".

Foram realizados 4 experimentos diferentes com a aplicação classificador SVM para identificar com quais características ele apresentaria uma maior acurácia, ou seja, uma maior taxa de acerto para novas entradas, as aplicações foram das seguintes maneiras:

- Experimento 1 - SVM para as 8 características
- Experimento 2 - SVM para Média de Saltos e Média do Tempo de Resposta
- Experimento 3 - SVM para as 8 características exceto Média de Saltos e Desvio Padrão de Saltos
- Experimento 4 - SVM para a Média e Desvio Padrão do Tempo de Resposta

A Figura 10, apresenta o que ocorreu em cada experimento. Ela exibe desde a escolha das características utilizadas para a aplicação do classificador até a etapa de avaliação dos resultados. Para chegar à avaliação dos resultados foram realizadas 10 repetições do classificador contendo as mesmas características e avaliando o resultado da matriz de confusão, para posteriormente calcular a média dos valores.

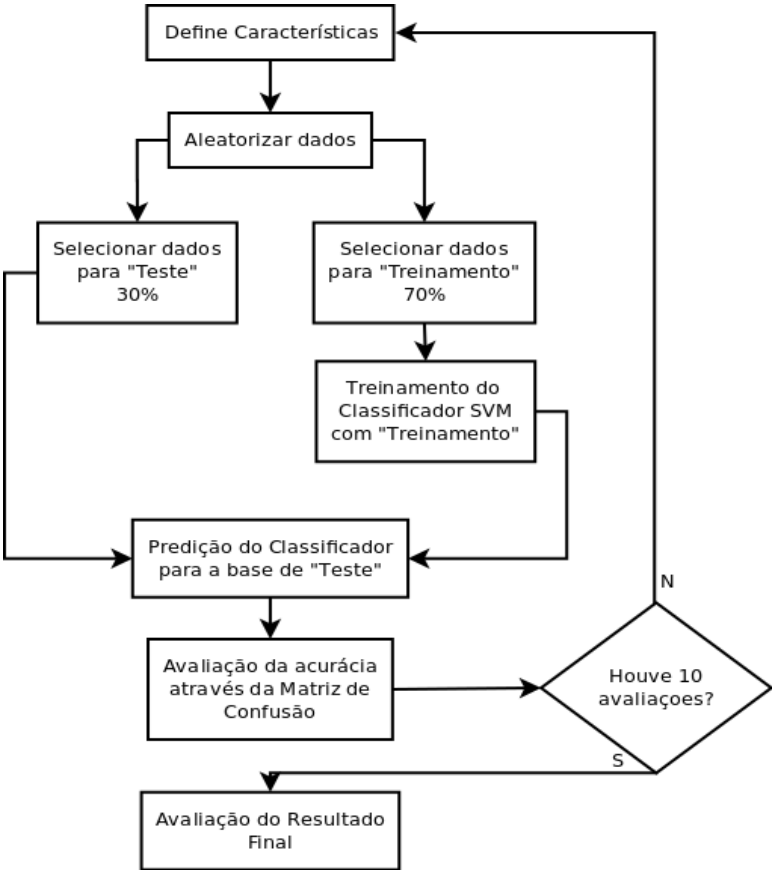

Figura 10 – Representação das etapas do projeto

3.6 Métrica de avaliação dos resultados

Foi avaliada a precisão do classificador após 10 execuções sobre cada base de dados composta pela média e o desvio padrão das características originais. Para poder analisar a precisão do nosso classificador de forma não tendenciosa, a cada repetição eram utilizados novos dados para "treinamento" e "teste", desta maneira todas as avaliações do classificador foram obtidos através de dados diferentes, acarretando resultados não viciados.

Após o "treinamento" dos dados foi realizada a predição pelo classificador onde ele tentava identificar a qual classe os dados de "teste" pertencem, onde foi gerada a Matriz de Confusão gerada pelos resultados do classificador *SVM*. Segundo Senna (2010), a matriz de confusão oferece métricas efetivas de identificar se as predições realizadas pelo classificador foram corretas ou incorretas. A matriz de confusão pode ser representada segundo a Figura 11. Através da análise da matriz pode-se obter as seguintes informações:

- Acurácia: Taxa (%) das amostras que foram classificados corretamente.
- Precisão: Taxa (%) com que o classificador oferece resultados similares após várias execuções.
- Sensibilidade: Taxa (%) de classificação positiva das amostras realmente positivas.

- Especificidade: Taxa (%) de classificação negativa das amostras realmente negativas.

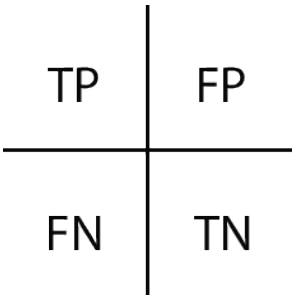

Figura 11 – Representação de uma Matriz de Confusão
Fonte: Adaptado de Senna (2010)

Onde:

- TP: True Positive ou Verdadeiro Positivo
- FP: False Positive ou Falso Positivo.
- FN: False Negative ou Falso Negativo.
- TN: True Negative ou Verdadeiro Negativo.

A medida analisada neste trabalho foi a acurácia, precisão, sensibilidade e especificidade. Os resultados analisados foram obtidos através das seguintes Equações:

$$Acurácia = \frac{TP + TN}{TP + TN + FN + FP} \tag{3.1}$$

$$Precisão = \frac{TP}{TP + FP} \tag{3.2}$$

$$Sensibilidade = \frac{TP}{|P|} = \frac{TP}{TP + FN} \tag{3.3}$$

$$Especifidade = \frac{TN}{|N|} = \frac{TN}{TN + FP} \tag{3.4}$$

4 Resultados e Discussão

Os resultados foram divididos em 6 seções, onde cada uma delas apresentará resultados e discussões e posteriormente será realizada uma consideração sobre os resultados.

4.1 Seleção de Características

Os resultados obtidos através da métrica de seleção de características *F-Score* é exibido na Tabela 3.

Tabela 3 – Características ranqueadas através do F-Score

Característica	Nome	F-score
5	mediasaltos	162.0368
1	mediavelocidade	2.3421
3	mediatempoderesposta	0.1539
4	stdtempoderesposta	0.0423
6	stdsaltos	0.0268
8	stderros	0.0094
7	mediaerros	0.0074
2	stdvelocidade	0.0051

Os resultados obtidos através da métrica de seleção de características Coeficiente de Correlação de Pearson, são exibidos na Tabela 4.

Tabela 4 – Características ranqueadas através do Coeficiente de correlação de Pearson

Característica	Nome	Pearson
5	mediasaltos	0.9969
1	mediavelocidade	0.8376
3	mediatempoderesposta	0.3659
4	stdtempoderesposta	0.2020
6	stdsaltos	-0.1621
8	stderros	0.0970
7	mediaerros	0.0861
2	stdvelocidade	-0.0714

Observa-se que os resultados apresentados pelas duas métricas de seleção de características F-Score e Coeficiente de Correlação de Pearson apresentaram a mesma ordem de relevância para as seguintes características médiasaltos, mediavelocidade, mediatempoderesposta, stdtempoderesposta, stdsaltos, stderros, mediaerros, stdvelocidade. A diferença é apresentada no Coeficiente de Correlação de Pearson, onde duas características apresentam valores negativos, o que significa que apresentou uma correlação negativa.

As características que apresentaram maior relevância para o estudo foram mediasaltos e mediavelocidade, através do F-Score foram obtidos 162,0368 e 2,3421 respectivamente. Através do Coeficiente de Correlação de Pearson as mesmas características obtiveram os valores de 0,9969 e 0,8376, respectivamente.

4.2 Experimento 1 - SVM para as 8 características

Os resultados obtidos através das avaliações da aplicação do classificador *SVM*, utilizando a base de dados contendo todas as 8 características (Tabela 5).

Tabela 5 – Avaliação do resultado do classificador para as 8 características

Repetição	Acurácia	Precisão	Sensibilidade	Especifidade
1	100%	100%	100%	100%
2	100%	100%	100%	100%
3	100%	100%	100%	100%
4	100%	100%	100%	100%
5	100%	100%	100%	100%
6	100%	100%	100%	100%
7	100%	100%	100%	100%
8	100%	100%	100%	100%
9	100%	100%	100%	100%
10	100%	100%	100%	100%
Média	100%	100%	100%	100%
Desvio padrão	0%	0%	0%	0%

Observa-se que quando o classificador SVM foi treinado e testado com as 8 características utilizadas no estudo, apresentou-se uma acurácia média de 1, ou seja, conseguiu identificar corretamente 100% dos dados corretamente, apresentando um resultado perfeito.

4.3 Experimento 2 - SVM para Média de Saltos e Média do Tempo de Resposta

Os resultados obtidos através das avaliações da aplicação do classificador *SVM*, utilizando a base de dados contendo as características Media de Saltos e Média do tempo de resposta (Tabela 6).

Tabela 6 – Avaliação do resultado do classificador para Média Saltos e Média Tempo de Resposta

Repetição	Acurácia	Precisão	Sensibilidade	Especifidade
1	100%	100%	100%	100%
2	100%	100%	100%	100%
3	100%	100%	100%	100%
4	100%	100%	100%	100%
5	100%	100%	100%	100%
6	100%	100%	100%	100%
7	100%	100%	100%	100%
8	100%	100%	100%	100%
9	100%	100%	100%	100%
10	100%	100%	100%	100%
Média	100%	100%	100%	100%
Desvio padrão	0%	0%	0%	0%

Observa-se que foi obtido uma acurácia média de 1, ou seja, conseguiu identificar corretamente 100% de novos ataques de *DNS Spoofing*. Foram utilizados essas duas características pois ambas são obtidas através de apenas um comando e não apresentam nenhuma relação direta, desta maneira, uma não causa influência sobre a outra.

4.4 Experimento 3 - SVM para as 8 características exceto Média de Saltos e Desvio Padrão de Saltos

Os resultado obtidos através das avaliações da aplicação do classificador *SVM*, utilizando a base de dados contendo as 8 características avaliadas no estudo exceto Média e Desvio Padrão de Saltos (Tabela 7).

Tabela 7 – Avaliação do resultado do classificador para as 8 características exceto Média e Desvio Padrão de Saltos

Repetição	Acurácia	Precisão	Sensibilidade	Especifidade
1	100%	100%	100%	100%
2	99,33%	98,63%	100%	98,72%
3	98,67%	100%	97,44%	100%
4	98,67%	97,46%	100%	97,26%
5	99,33%	100%	98,51%	100%
6	100%	100%	100%	100%
7	100%	100%	100%	100%
8	99,33%	100%	98,59%	100%
9	98,67%	97,26%	100%	97,47%
10	99,33%	98,68%	100%	98,67%
Média	99,33%	99,20%	99,45%	99,21%
Desvio padrão	0,54%	1,11%	0,98%	1,11%

Foi obtido uma acurácia de 0,99933, ou seja, conseguiu identificar corretamente 99,33% de novos ataques corretamente.

4.5 Experimento 4 - SVM para a Média e Desvio Padrão do Tempo de Resposta

O resultado obtidos através da aplicação do classificador apenas para as caracteristicas Média de Tempo de Resposta e Desvio Padrão do Tempo de Resposta (Tabela 8).

Tabela 8 – Avaliação do resultado do classificador para Média e Desvio Padrão do Tempo de Resposta

Repetição	Acurácia	Precisão	Sensibilidade	Especifidade
1	98%	95,94%	100%	96,20%
2	99,33%	98,76%	100%	98,57%
3	100%	100%	100%	100%
4	98%	97,5%	98,73%	97,18%
5	97,33%	95,94%	98,61%	96,15%
6	98%	96,42%	100%	95,65%
7	98%	96,2%	100%	95,95%
8	98%	96,05%	100%	96,1%
9	96,7%	97,29%	100%	97,44%
10	100%	100%	100%	100%
Média	98,53%	97,41%	99,73%	97,32%
Desvio padrão	0,93%	1,62%	0,56%	1,65%

Quando foram utilizadas as características Média do Tempo de Resposta e Desvio Padrão do Tempo de Resposta, foi obtida uma acurácia média de 0,9853, ou seja, conseguiu-se identificar corretamente a classe dos dados fornecidos em 98,53% das vezes. Foram utilizadas apenas as características sobre o tempo de resposta, pois elas são obtidas através de um único comando, onde o mesmo apresenta pouco gasto computacional e pouco tempo para a coleta.

4.6 Considerações sobre os resultados

As acurácias médias obtidas através das Tabelas anteriores foram ranqueadas (Tabela 9).

Tabela 9 – Avaliação da acurácia das utilizações do *SVM*

Características	Acurácia Média	Diferença
SVM para as 8 características	100%	-
Média Saltos, Media Tempo de Resposta	100%	-
SVM para as 8 características exceto Saltos	99,33% ± 0,54%	-0,67%
Média e Desvio Padrão do Tempo de Resposta	98,53% ± 0,93%	-1,47%

O resultado mostra que a acurácia mínima obtida com o classificador *SVM* para a identificação de novos ataques de *DNS Spoofing* é de 98,53%, desta maneira, observa-se que o classificador apresenta para qualquer base de dados utilizada uma elevada taxa de acerto para a identificação de novos ataques.

A melhor acurácia alcançada pelo classificador foi obtida através de dois experimentos, sendo que ambos possuiam a característica Saltos, sendo obtida uma acurácia média de 100%. É importante ressaltar que apesar de ter obtido 100% de acurácia, o gasto computacional é extremamente elevado, desta maneira para realizar as coletas dos dados referentes a característica Salto é despendido uma enorme quantidade de tempo, o que pode significar que não seja viável a captação da característica se for necessário uma resposta rápida de uma ocorrencia de ataque de DNS Spoofing.

As características relacionadas ao tempo de resposta, as quais são obtidas apenas com um comando, rápidas de serem coletadas e armazenadas, obteve uma acurácia média de 98,53%, sendo a 4ª melhor base de dados utilizada para treinamento e teste do classificador, obtendo uma piora de 1,47% comparada aos experimentos que possuiam a característica Saltos.

A utilização de todas as características exceto a Média de Saltos e Desvio Padrão de Saltos, obtiveram uma acurácia média de 99,33%, sendo a 3ª melhor base de dados utilizada para treinamento e teste do classificador, onde obteve uma piora de 0,67% comparada aos experimentos que apresentavam a característica Média e Desvio Padrão de Saltos.

Jin, Wang e Shin (2003), propõem um mecanismo nomeado *HCF - HOP Count Filtering*, esse mecanismo apresenta o funcionamento através do número de Saltos padrão até o servidor de destino, para ataques de negação de serviço. Templeton e Levitt (2003) criaram um mecanismo para detecção de Spoofing através da utilização também do número de saltos para ataques de negação de serviço, onde basicamente, o servidor saberia da ocorrência de spoofing se ouvesse uma alteração no número de saltos entre ele e o cliente Guo, Chen e Chiueh (2006) afirmam que é uma boa solução, mas que pode apresentar problemas sobre a aprendizagem falsa sobre o número de saltos. Guo, Chen e Chiueh (2006) em seu trabalho para Detecção de *Spoofing* para prevenir ataques de negação de serviço, utiliza duas características, sendo elas TTL e RTT.

Na literatura não foram encontrados trabalhos que realizavam a aplicação do

classificador SVM para casos de *DNS Spoofing*. Foram encontrados trabalhos com a aplicação do SVM em diversas áreas do conhecimento.

5 Conclusão

Através deste trabalho observa-se que a aplicação de reconhecimento de padrões com a aplicação de técnicas de seleção de características como F-Score e Coeficiente de correlação de Pearson e a técnica de classificação de características como *SVM* é possível predizer com 100% de acurácia novos ataques de *DNS Spoofing*.

A característica Média de Saltos, obteve o melhor resultado através das técnicas de seleção de características, sendo considerada a mais relevante para o estudo. Através da utilização das característica Média de Saltos e Desvio Padrão de Saltos foram obtidos os melhores resultados sendo possível predizer com perfeição em todos os casos de novos ataques, ou seja, com a utilização das características Pulos obtem-se uma acurácia de 100% para novos ataques.

Apesar do bom resultado com a utilização da Média de Saltos deve-se levar em consideração que captação desses dados demanda um alto gasto computacional.

As características Média do Tempo de Resposta foi a 3ª melhor característica através das técnicas de seleção de características. Em conjunto com o Desvio Padrão do Tempo de Resposta obteve um alto acerto para predição de novos ataques, considerado o 4º melhor experimento, com 98,53% $\pm$ 0,93%. As características Média do Tempo de Resposta e Desvio Padrão do Tempo de Resposta são obtidas através de um único comando e apresentam baixo gasto computacional o que a torna excelente para uma captação e decisão rápidas para analise de uma ocorrência de ataque.

6 Trabalhos Futuros

Para estudos futuros existem outros temas que podem ser explorados: identificar novas caratrísticas que possam apresentar uma maior acurácia para identificação de novos ataques; aplicar outras funçoes de *Kernel* (RBF, sigmóide e polinomial) do algoritmo *SVM*, identificar se a rede mais sobrecarregada de dispositivos irá influenciar no resultado das técnicas de reconhecimento de padrões; aplicar novas técnicas de classificação de características para verificar qual obtém melhor predição.

Referências

BEZDEK, J. C. What is computational intelligence. *Computational Intelligence: Imitating Life*, Piscataway, NJ: IEEE Press, p. 1–12, 1994. Citado na página 21.

BIANCHI, M. F. de. *Extração de características de imagens de faces humanas através de Wavelests, PCA e IMPCA*. Dissertação (Mestrado), 2006. Citado na página 23.

BREMLER-BARR, A.; LEVY, H. *Spoofing Prevention Method*. 2004. Citado na página 20.

BREMLER-BARR, A.; LEVY, H. Spoofing prevention method. In: IEEE. *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*. [S.l.], 2005. v. 1, p. 536–547. Citado na página 20.

CANTÚ, E. Redes de computadores e a internet. *São José*, 2003. Citado na página 15.

CAPRON, H.; JOHNSON, J. A.; SANTOS, J. C. B. dos. *Introdução à informática*. [S.l.]: Pearson Prentice Hall, 2006. Citado na página 12.

CASTELLS, M. *A Galáxia Internet: reflexões sobre a Internet, negócios e a sociedade*. [S.l.]: Zahar, 2003. Citado na página 12.

CERT.BR. *Estatísticas dos Incidentes Reportados ao CERT.br*. 2015. Disponível em: <http://www.cert.br/stats/incidentes/>. Acesso em: 15 fev. 2016. Citado na página 17.

CHAVES, A. d. C. F. *Extração de Regras Fuzzy para Máquinas de Vetores Suporte (SVM) para Classificação em Múltiplas Classes*. Tese (Doutorado) — PUC-Rio, 2006. Citado na página 25.

CORTES, C.; VAPNIK, V. Support-vector networks. *Machine learning*, Springer, v. 20, n. 3, p. 273–297, 1995. Citado na página 25.

DUCH, W. What is computational intelligence and where is it going? In: *Challenges for computational intelligence*. [S.l.]: Springer, 2007. p. 1–13. Citado na página 21.

DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification*. 2. ed. New York: Wiley, 2001. Citado 3 vezes nas páginas 13, 23 e 24.

EBERHART, R.; SIMPSON, P.; DOBBINS, R. *Computational intelligence PC tools*. [S.l.]: Academic Press Professional, Inc., 1996. Citado na página 21.

EXAME. *Mais da metade dos brasileiros são usuários da internet*. 2014. Disponível em: <http://exame.abril.com.br/tecnologia/noticias/mais-da-metade-dos-brasileiros-sao-usuarios-da-internet>. Acesso em: 15 fev. 2016. Citado na página 12.

FERGUSON, P. Network ingress filtering: Defeating denial of service attacks which employ ip source address spoofing. 2000. Citado na página 20.

GIAVAROTO, S. C. R.; SANTOS, G. R. *Backtrack Linux Auditoria e Teste de Invasão em Redes de Computadores*. [S.l.]: Ciência Moderna, 2013. Citado na página 18.

GU, Q.; LI, Z.; HAN, J. Generalized fisher score for feature selection. *arXiv preprint arXiv:1202.3725*, 2012. Citado na página 23.

GUO, F.; CHEN, J.; CHIUEH, T.-c. Spoof detection for preventing dos attacks against dns servers. In: IEEE. *Distributed Computing Systems, 2006. ICDCS 2006. 26th IEEE International Conference on*. [S.l.], 2006. p. 37–37. Citado na página 38.

HORTA, E. G. Previsores para a eficiência da quimioterapia neoadjuvante no câncer de mama. *M. Sc., Universidade Federal de Minas Gerais (UFMG)*, 2008. Citado na página 26.

HORTA, E. G. et al. Extração de características e casamento de padrões aplicados à estimação de posição de um VANT. *UFMG*, 2011. Citado na página 26.

ISO/IEC17799. *Tecnologia da informação — Técnicas de segurança — Código de prática para a gestão da segurança da informação*. [S.l.]: ABNT, 2005. Citado 2 vezes nas páginas 15 e 16.

IYODA, E. M. Inteligência computacional no projeto automático de redes neurais híbridas e redes neurofuzzy heterogêneas. Biblioteca Digital da Unicamp, 2000. Citado na página 21.

JIN, C.; WANG, H.; SHIN, K. G. Hop-count filtering: an effective defense against spoofed ddos traffic. p. 30–41, 2003. Citado na página 38.

JOACHIMS, T. *Learning to classify text using support vector machines: Methods, theory and algorithms*. [S.l.]: Kluwer Academic Publishers, 2002. Citado na página 24.

KIM, K. I. et al. Support vector machines for texture classification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 24, n. 11, p. 1542–1550, 2002. Citado na página 24.

LAMPING, U.; WARNICKE, E. Wireshark user's guide. *Interface*, v. 4, n. 6, 2004. Citado na página 18.

LEMOS, A. L. Ciber-rebeldes. *Universidad Federal de Bahia, http://www. cfh. ufsc. br/˜ cso5421/bibliografias/rebelde. html*, 1999. Citado 2 vezes nas páginas 13 e 18.

LIMA, A. R. G. Máquinas de vetores suporte na classificaçao de impressoes digitais. *Universidade Federal do Ceará, Departamento de Computação, Fortaleza-Ceará*, 2002. Citado na página 26.

LORENA, A. C.; CARVALHO, A. C. de. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, v. 14, n. 2, p. 43–67, 2007. Citado 4 vezes nas páginas 22, 23, 24 e 25.

LYON; FYODOR, G. *Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning*. USA: Insecure, 2009. ISBN 0979958717, 9780979958717. Citado na página 18.

LYRA, M. R. *Segurança e auditoria em sistemas de informação*. [S.l.]: Rio de Janeiro: Ciência Moderna, 2008. Citado 3 vezes nas páginas 15, 16 e 17.

NOBLE, W. S. et al. Support vector machine applications in computational biology. *Kernel methods in computational biology*, p. 71–92, 2004. Citado na página 24.

ONUBR. *Em 15 anos, número de usuários de internet passou de 400 milhões para 3,2 bilhões, revela ONU*. 2015. Disponível em: <https://nacoesunidas.org/em-15-anos-numero-de-usuarios-de-internet-passou-de-400-milhoes-para-32-bilhoes-revela-onu/>. Acesso em: 15 fev. 2016. Citado na página 12.

PONTIL, M.; VERRI, A. Support vector machines for 3d object recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 20, n. 6, p. 637–646, 1998. Citado na página 24.

PRITCHETT, W. L.; SMET, D. D. *Kali Linux Cookbook*. [S.l.]: Packt Publishing Ltd, 2013. Citado na página 18.

ROSS, J. *Redes de Computadores*. [S.l.]: Antenna Edições Técnicas, 2008. Citado 2 vezes nas páginas 14 e 15.

SCHOLKOPF, B.; GUYON, I.; WESTON, J. Statistical learning and kernel methods in bioinformatics. *Nato Science Series Sub Series III Computer and Systems Sciences*, IOS press, v. 183, p. 1–21, 2003. Citado na página 24.

SEMOLINI, R. *Support vector machines, inferência transdutiva e o problema de classificação*. Tese (Doutorado) — Universidade Estadual de Campinas, 2002. Citado na página 26.

SENNA, S. L. de. *Computação evolucionária Aplicada ao Diagnóstico de Falhas Incipientes em Transformadores de Potência Utilizando Dados de Cromatografia*. Dissertação (Mestrado), Setembro 2010. Citado 2 vezes nas páginas 32 e 33.

SMOLA, A. J.; SCHÖLKOPF, B. *Learning with kernels*. [S.l.]: Citeseer, 1998. Citado na página 25.

SOARES, H. B. *Análise e classificação de imagens de lesões da pele por atributos de cor, forma e textura utilizando máquina de vetor de suporte*. Tese (Doutorado), 2008. Citado na página 24.

SOUTO, M. de et al. Técnicas de aprendizado de máquina para problemas de biologia molecular. *Sociedade Brasileira de Computaçao*, 2003. Citado na página 22.

TANASE, M. Ip spoofing: an introduction. *Security Focus*, v. 11, 2003. Citado na página 19.

TANENBAUM, A. S. *Redes de computadores*. [S.l.]: Pearson Educación, 2003. Citado 4 vezes nas páginas 12, 13, 14 e 15.

TEMPLETON, S. J.; LEVITT, K. E. Detecting spoofed packets. In: IEEE. *DARPA Information Survivability Conference and Exposition, 2003. Proceedings*. [S.l.], 2003. v. 1, p. 164–175. Citado na página 38.

VAPNIK, V. N.; VAPNIK, V. *Statistical learning theory*. [S.l.]: Wiley New York, 1998. v. 1. Citado na página 24.

VIANNA, T. L. Hackers: um estudo criminológico da subcultura cyberpunk. *Revista do Centro Acadêmico Afonso Pena*, v. 6, n. 1, 2001. Citado 2 vezes nas páginas 13 e 18.