FUNDAÇÃO DE ENSINO “EURÍPIDES SOARES DA ROCHA”
CENTRO UNIVERSITÁRIO EURÍPIDES DE MARÍLIA – UNIVEM
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

MÁRIO CLEBER BIDOIA

TESTES DE VULNERABILIDADES DE NEGAÇÃO DE SERVIÇOS EM DIFERENTES VERSÕES DE SISTEMAS OPERACIONAIS PARA SERVIDORES VIRTUALIZADOS

MARÍLIA
2009

MÁRIO CLEBER BIDOIA

TESTES DE VULNERABILIDADES DE NEGAÇÃO DE SERVIÇOS EM DIFERENTES VERSÕES DE SISTEMAS OPERACIONAIS PARA SERVIDORES VIRTUALIZADOS

Trabalho de Curso apresentado ao Curso de Bacharelado em Ciência da Computação da Fundação de Ensino "Eurípides Soares da Rocha", mantenedora do Centro Universitário Eurípides de Marília – UNIVEM, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Orientador Prof. Emerson Alberto Marconato

MARÍLIA
2009

Bidoia, Mário Cleber

Testes de vulnerabilidades de negação de serviços em diferentes versões de sistemas operacionais para servidores virtualizados / Mário Cleber Bidoia; orientador: Emerson Alberto Marconato. Marília, SP: [s.n.], 2009.

78 f.

Trabalho de Curso Bacharel em Ciência da Computação – Curso de Ciência da Computação, Fundação de Ensino "Eurípides Soares da Rocha", mantenedora do Centro Universitário Eurípides de Marília – UNIVEM, Marília, 2009.

1.Segurança 2.Virtualização 3.Vulnerabilidades 4.DoS 5. DDoS.

CDD: 005.8

CENTRO UNIVERSITÁRIO EURÍPIDES DE MARÍLIA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

TRABALHO DE CONCLUSÃO DE CURSO – AVALIAÇÃO FINAL

Mário Cleber Bidóia

TESTES DE VULNERABILIDADES DE NEGAÇÃO DE SERVIÇOS EM DIFERENTES VERSÕES DE SISTEMAS OPERACIONAIS PARA SERVIDORES VIRTUALIZADOS

Banca examinadora da monografia apresentada ao Curso de Bacharelado em Ciência da Computação do UNIVEM/F.E.E.S.R., para obtenção do Título de Bacharel em Ciência da Computação.

Nota: 9,0 (NOVE)

Orientador: Emerson Alberto Marconato

1°. Examinador: Ricardo Petruzza do Prado

2°. Examinador: Fábio Dacêncio Pereira

Marília, 04 de dezembro de 2009.

DEDICATÓRIA

A Deus, por me dar a oportunidade da vida.

Aos amigos, pela força e colaboração.

Aos meus pais, pela educação que me deram.

Às minhas irmãs.

À minha noiva e futura esposa Aline Luiza.

AGRADECIMENTOS

Agradeço primeiramente a Deus, porque além de ter me ofertado o dom da vida colocou em meu caminho excelentes pessoas, que foram fundamentais nesses quatro anos de graduação.

Agradeço ao meu orientador professor Emerson Alberto Marconato, pela dedicação e esforço em “socorrer-me” nos momentos de aperto, muitas vezes deixando de estar no conforto de sua casa com seu filho recém-nascido para comparecer às nossas reuniões.

Agradeço aos Professores Ildeberto de Genova Bugatti e Juliana de Oliveira pelo incentivo a pesquisa e pela atenção e disponibilidade nos momentos em que os procurava em suas respectivas salas para tirar certas dúvidas.

Agradeço aos meus amigos Rafael van Winkel e Marcelo Omoto pela amizade conquistada e pelo incentivo constante.

Agradeço também a um amigo em particular, Lucas Salviano, pois apesar do desafeto no início da graduação, com o passar dos anos cultivou-se uma amizade muito forte e divertidíssima.

Agradeço às minhas irmãs Bianca e Carina, pelo apoio e admiração durante esses quatro anos de intensa batalha.

Agradeço aos meus pais por sempre apoiarem meus estudos e fazerem de tudo para a realização de um sonho: a graduação.

E por último agradeço à mulher que mudou completamente a minha vida, pois sem a sua colaboração, força e incentivo eu jamais teria chego onde estou. Agradeço, então, à minha “professorinha” Aline Luiza (minha Li), noiva e em breve minha esposa.

"Duas coisas são infinitas: o universo e a estupidez humana. Mas, no que respeita ao universo, ainda não adquiri a certeza absoluta".

Albert Einstein

BIDOIA, Mário Cleber. **Testes de vulnerabilidades de negação de serviços em diferentes versões de sistemas operacionais para servidores virtualizados**. 2009. 78 f. Trabalho de Curso Bacharelado em Ciência da Computação – Centro Universitário Eurípides de Marília, Fundação de Ensino "Eurípides Soares da Rocha", Marília, 2009.

RESUMO

Os ataques de Negação de Serviço (*Denial of Service* - DoS) e de Negação de Serviço Distribuído (*Distributed Denial of Service* – DDoS) são uns dos principais ataques aos servidores, sendo que alguns destes ataques exploram vulnerabilidades de algumas versões de sistemas operacionais para servidores. Este trabalho tem como objetivo estudar quais versões destes sistemas operacionais estão vulneráveis. Muitas destas vulnerabilidades são exploradas devido ao fato de más configurações dos sistemas operacionais para servidores. Trata-se de uma pesquisa exploratória e descritiva, com revisão bibliográfica, em que foram realizados diversos ataques às diferentes versões de sistemas operacionais para servidores Windows e Linux, utilizando ferramentas de ataques DDoS. Os sistemas operacionais foram instalados com suas configurações *default.* Com estes testes detectaram-se as possíveis vulnerabilidades e, dessa maneira, demonstrou-se a necessidade de uma correta configuração e o auxílio de software e ferramentas específicas para aumentar a segurança.

Palavras-chave: segurança, virtualização, vulnerabilidades, DoS e DDoS.

BIDOIA, Mário Cleber. **Testes de vulnerabilidades de negação de serviços em diferentes versões de sistemas operacionais para servidores virtualizados**. 2009. 78 f. Trabalho de Curso Bacharelado em Ciência da Computação – Centro Universitário Eurípides de Marília, Fundação de Ensino "Eurípides Soares da Rocha", Marília, 2009.

ABSTRACT

Attacks Denial of Service (DoS) and Distributed Denial of Service (DDoS) are one of the main attacks on the servers, being that some of these attacks exploit vulnerabilities in some versions of operating systems for servers. This work aims to study which versions of these operating systems are vulnerable. Many of these vulnerabilities are exploited due to the fact of bad configurations of operating systems for servers. This is an exploratory and descriptive research, with bibliographical review, carried several attacks out on different operating system versions for Windows and Linux servers, using tools of DDoS attacks. Operating systems was installed with its default settings. With these tests detected possible vulnerabilities and, that way, demonstrated the need for a correct configuration and support of software and specific tools to improve the safety.

Keywords: security, virtualization, vulnerabilities, DoS and DDoS.

LISTA DE FIGURAS

Figura 1 – Arquitetura de Firewall 23
Figura 2 – Funções do IDS na arquitetura do sistema 25
Figura 3 – Arquitetura de um ataque DDoS 38
Figura 4 – Ataques DDOS derrubam sites do Irã 39
Figura 5– Site do Twitter sofre ataque DDoS 40
Figura 6 – Facebook sofre ataque DDoS 41
Figura 7 – Arquitetura do ambiente de teste 47
Figura 8 – Relação das Máquinas Virtuais e o VMM 49
Figura 9 – Arquitetura da Execução do VMWare 51
Figura 10 – Tela de compilação do TFN2K 53
Figura 11 – Resposta do Windows 2003 ao Ataque de ID 4 56
Figura 12 – Resposta do Windows 2003 ao Ataque de ID 5 56
Figura 13 – Resposta do Windows 2003 ao Ataque de ID 6 57
Figura 14 – Resposta do Windows 2003 ao Ataque de ID 8 57
Figura 15 – Resposta do Windows 2008 ao Ataque de ID 4 59
Figura 16 – Resposta do Windows 2008 ao Ataque de ID 5 59
Figura 17 – Resposta do Windows 2008 ao Ataque de ID 6 60
Figura 18 – Resposta do Windows 2008 ao Ataque de ID 8 60
Figura 19 – Resposta do Ubuntu 6 ao Ataque de ID 4 61
Figura 20 – Resposta do Ubuntu 6 ao Ataque de ID 5 62
Figura 21 – Resposta do Ubuntu 6 ao Ataque de ID 6 62
Figura 22 – Resposta do Ubuntu 6 ao Ataque de ID 8 63
Figura 23 – Resposta do Fedora 05 ao Ataque de ID 4 64
Figura 24 – Resposta do Fedora 05 ao Ataque de ID 5 65
Figura 25 – Resposta do Fedora 05 ao Ataque de ID 6 65
Figura 26 – Resposta do Fedora 05 ao Ataque de ID 8 66
Figura 27 – Resposta do Fedora 06 ao ataque ID 04 67
Figura 28 – Resposta do Fedora 06 ao ataque ID 05 68
Figura 29 – Resposta do Fedora 06 ao ataque ID 06 68
Figura 30 – Resposta do Fedora 06 ao ataque ID 08 69
Figura 31 – Resposta do Fedora 09 ao ataque ID 04 70
Figura 32 – Resposta do Fedora 09 ao ataque ID 05 71

Figura 33 – Resposta do Fedora 09 ao ataque ID 06.....71
Figura 34 – Resposta do Fedora 09 ao ataque ID 08.....72

LISTA DE ABREVIATURAS E SIGLAS

ACK: Acknowledge

CERT.br: Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil

DoS: Denial of Service

DDoS: Distributed Denial of Service

ICMP: Internet Control Message Protocol

IDS: Intrusion Detection System

IP: Internet Protocol

RST: Reset

SYN: synchronize

SO: Sistema Operacional

TFN: Tribe Flood Network

TFN2K: Tribe Flood Network 2000

TCP: Transmission Control Protocol

UDP: User Datagram Protocol

LISTA DE GRÁFICOS

Gráfico 1 – Ataques reportados ao CERT de abril a junho de 2009 41
Gráfico 2 – Ataques reportados ao CERT de julho a setembro de 2009 42
Gráfico 3 – Ataques reportados ao CERT no ano 2008 42
Gráfico 4 – Tipo de Ataque x Pacotes Perdidos Windows 2003 58
Gráfico 5 – Tipo de Ataque x Pacotes Perdidos Windows 2008 61
Gráfico 6 – Tipo de Ataque x Pacotes Perdidos Ubuntu 06 63
Gráfico 7 – Tipo de Ataque x Pacotes Perdidos Fedora 05 66
Gráfico 8 – Tipo de Ataque x Pacotes Perdidos Fedora 06 69
Gráfico 9 – Tipo de Ataque x Pacotes Perdidos Fedora 09 72
Gráfico 10 – Resultados obtidos 74

SUMÁRIO

INTRODUÇÃO 14
CAPÍTULO 1 – SEGURANÇA EM REDES DE COMPUTADORES 16
1.1 – Políticas de Segurança 17
1.2 – Vulnerabilidades 18
1.2.1 – Vulnerabilidades de Configuração 19
1.2.2 – Vulnerabilidades de Rede 19
1.2.3 – Vulnerabilidade de software 20
1.2.4 – Ausência de Ferramentas de Segurança 21
1.3 – Mecanismos de Defesas 21
1.3.1 – Criptografia 22
1.3.2 – Firewall 23
1.3.3 – Sistema de Detecção de Intrusão 24
CAPÍTULO 2 – AMEAÇAS E TIPOS DE ATAQUES À REDE DE COMPUTADORES 27
2.1 – Códigos Maliciosos 28
2.2 – *Port Scanning* – Varredura de Portas 29
2.3 – Engenharia Social 31
2.4 – Outras ameaças 32
CAPÍTULO 3 – DoS (*Denial of Service*) – Negação de Serviço 34
3.1 – DDoS (*Distributed Denial of Service*) – negação de serviço distribuído 37
3.2 – Ferramentas de Ataque de Negação de Serviço Distribuído 43
3.2.1 – Trinoo 43
3.2.2 – TFN – Tribe Flood Network 44
3.2.3 – STACHELDRAHT 44
CAPÍTULO 4 – TESTES DAS VULNERABILIDADES 46
4.1 – Virtualização 47
4.2 – Ferramentas Utilizadas 51
4.2.1 – VMWare 51
4.2.2 – TFN2K – Trible Flood Network 2000 52
4.3 – Testes efetuados e resultados obtidos 54
4.3.1 – Ataque contra Windows Server 2003 55
4.3.2 – Ataque contra Windows Server 2008 58
4.3.3 – Ataque contra o Ubuntu 6 61

4.3.4 – Ataque contra o Fedora 5 64
4.3.5 – Ataque contra o Fedora 6 66
4.3.6 – Ataque contra o Fedora 9 70
4.3.7 – Análise dos testes efetuados 73
CONSIDERAÇÕES FINAIS 74
Propostas de trabalhos futuros 75
REFERÊNCIAS 76

INTRODUÇÃO

Atualmente há uma diversidade de redes de computadores, que são utilizadas na execução de diferentes operações. Entre estas redes destaca-se a Internet e os inúmeros serviços por ela oferecidos. Para administrar e oferecer esses serviços aos milhões de usuário da Internet são necessários diversos servidores espalhados pelo mundo e, conseqüentemente, sistemas operacionais dedicados a estes servidores.

Devido o caráter público da Internet não há recursos viáveis para gerenciar todos os seus usuários e, como no organismo social, existem pessoas que se utilizam de diversos meios e instrumentos para prejudicarem outras, no âmbito da Internet não é diferente. O que se pode observar é que existe uma quantidade considerável de usuários maliciosos que utilizam os mais diversos recursos para prejudicar os demais usuários. Por esse motivo a segurança se tornou um dos requisitos mais importantes para um sistema web.

Entre uma variedade de tipos de ataques existentes na atualidade estão o de Negação de Serviço (*Denial of Service* – DoS) e uma evolução deste ataque, o Ataque de Negação de Serviço Distribuído (*Distributed Denial of Service* - DDoS).

Um conhecido ataque que gerava Negação de Serviço é o Ping da Morte (*Ping of Death*), que consistia no fato do antigo Windows 95, em conjunto com o protocolo TCP/IP, não suportar pacotes maiores de 64400 bytes, tornando o sistema vulnerável para um simples comando ping: ping -t -l 65500 <IP da vítima>. Este comando faz com que seja enviado ininterruptamente pacotes com tamanho superior ao suportado pelo Windows 95, fazendo então com que os sistemas reinicializassem ou simplesmente *travassem.*

Na atualidade, esse simples ataque não afeta os Sistemas Operacionais (principalmente o Windows), porém, outros métodos surgiram como SYN *Flooding*, ICMP *flood, Smurf*, entre outros, a partir do momento em que os invasores foram capazes de desvendar as fraquezas da família de protocolos TCP/IP.

Com base nos fatos relatados anteriormente, este trabalho objetivou testar as vulnerabilidades existentes em algumas versões de Sistemas Operacionais para Servidores em relação aos *Distributed Denial of Service* (DDoS), e desta maneira servir como referência para administradores de redes, apresentando as vulnerabilidades oferecidas pelas configurações default desses Sistemas Operacionais.

O presente trabalho foi delimitado e organizado em cinco capítulos essenciais:

O primeiro capítulo apresenta a descrição de fundamentos teóricos sobre segurança, tratando de questões relevantes ao tema, como alguns conceitos, possibilidades de vulnerabilidades e mecanismos de defesas.

No segundo capítulo há a abordagem dos principais tipos de ataques e ameaças existentes atualmente.

O terceiro capítulo apresenta as características do Ataque de Negação de Serviço e do Ataque de Negação de Serviço Distribuído.

A metodologia utilizada é relatada no quarto capítulo, como também os testes efetuados e os respectivos resultados obtidos.

O presente trabalho foi escrito utilizando a nova regra gramatical que passou a vigorar em 2009. Para isso foi utilizado o artigo de Douglas Tufano: Guia Prático da Nova Ortografia.

Considerando a relevância e a extensão do tema pesquisado, este trabalho não pretende apresentar métodos únicos e finitos, mas sim analisar os resultados obtidos com os testes realizados, almejando auxiliar trabalhos futuros acerca do mesmo tema.

CAPÍTULO 1 – SEGURANÇA EM REDES DE COMPUTADORES

Inicialmente as redes de computadores eram utilizadas principalmente por pesquisadores de universidades, com intuito de trocarem conhecimentos através de mensagens (TANENBAUM, 2003). Dessa forma, a questão da segurança da informação nunca foi alvo de atenção especial.

Com a expansão e popularização das redes de computadores e principalmente com o surgimento da Internet, milhares de usuários passaram a utilizá-la, sendo que atualmente diversas operações, como transações bancárias, videoconferências, mensagens instantâneas, compras on-line, entre outras, são efetuadas por meio das redes de computadores. Estas comodidades permitiram que pessoas com más intenções pudessem se aproveitar de falhas para prejudicar alguns usuários, seja por intenção financeira, vingança, simplesmente diversão ou outros motivos. Conseqüentemente houve um aumento da preocupação das empresas em relação ao assunto segurança.

Segundo Kurose, em sua obra 2005, uma solução segura adequada para a questão relatada anteriormente deve ter as seguintes propriedades:

- Confidencialidade: significa proteger informações contra sua divulgação para outra pessoa não autorizada – interna ou externamente. Consiste em proteger a informação contra leitura e/ou cópia por outra pessoa que não tenha sido explicitamente autorizado pelo dono daquela informação. A informação sempre deve ser protegida independente da mídia que a contenha, seja, impressa ou digital. Não se deve atentar somente para da proteção da informação como um todo, mas também de partes da informação que podem ser utilizadas para interferir sobre o todo. Tratando-se de redes de computadores, isto significa que os dados, enquanto em tráfego, não serão visualizados, modificados, ou retirados da rede por indivíduos não autorizados ou capturados por dispositivos ilícitos.

- Autenticidade: Autenticidade está relacionada com a correta identificação de um usuário ou computador. Em um sistema o serviço de autenticação tem a obrigação de assegurar ao receptor que a mensagem é realmente oriunda da origem informada em seu conteúdo. Geralmente, essa garantia pode ser

implementada a partir de mecanismos de senhas ou de assinatura digital. É importante a verificação da autenticidade posteriormente a qualquer processo de identificação, seja de um usuário para um sistema, de um sistema para outro sistema ou de um sistema para o usuário.

- Integridade: Garantir a proteção da informação contra alteração sem a devida permissão. Pode-se descrever como alteração escrita, modificação de conteúdo, alteração de status, criação ou remoção de dados. Integridade significa que os dados vão permanecer da maneira que foi armazenada, até uma possível alteração feita por seus proprietários.

- Disponibilidade: evitar a degradação dos serviços prestados pelo sistema para que dessa maneira o usuário utilize do serviço sempre que necessitar.

Diversas políticas, arquiteturas e soluções de segurança são propostas para aumentar a segurança das informações e tentar garantir essas propriedades.

1.1 – Políticas de Segurança

Política de segurança é um conjunto de regras que tem por objetivo proteger organizações contra possíveis ameaças às suas informações. Essas regras definirão ações que deverão ser efetuadas em prol desse objetivo, impondo restrições e responsabilidades sobre os conteúdos, controlando assim todos os usuários que a estes possuem acesso.

Caruso (1999) destaca que a política de segurança implantada corretamente e seguida rigidamente as regras poderá obter três aspectos como consequência: redução da probabilidade de ocorrência, redução dos danos provocados por eventuais ocorrências e criação de procedimentos para se recuperar de eventuais danos.

Segundo Tittel em *Redes de Computadores* (2003) a política de segurança deve ser aplicada a todos os usuários sem exceções, pois havendo quaisquer exceções a política de segurança inexiste. Dessa forma:

> A partir do momento em que se decide por uma política de segurança, ela precisa ser aplicada a todos os indivíduos, qualquer que seja a função que desempenhem no trabalho. Embora o CEO possa achar inconveniente mudar a senha a cada 30 dias, se fizer uma exceção para ficar mais conveniente e se estender a freqüência dessa, então não existe nenhuma política de segurança.

> Se apolítica determinar que todos os funcionários devem mudar as senhas a cada 30 dias, então isso se aplica a todos ao funcionários, seja qual for a função deles na empresa. (p. 214)

Observa-se a importância de que haja um grupo responsável por planejar e aplicar a política de segurança, assim como revisar e alterar quando se fizer necessário. Destaca-se também a importância da participação de profissionais de cargo elevado dentro da empresa envolvida na elaboração e aplicação da política de segurança, sendo da mais alta gerência a responsabilidade da sua aprovação.

A respeito da implantação de uma política de segurança Spanceski (2004) afirma que esta é frequentemente, a etapa mais difícil de ser concluída. Apesar deste conjunto de regras ser elaborado por um grupo de pessoas com amplo conhecimento em relação à segurança e este ser teoricamente eficiente, uma prática de sucesso pode ser dificultada quando se trata da compreensão por parte dos usuários quanto a sua importância e necessidade de uma correta utilização.

Pesquisadores afirmam que a política de segurança deve ser pública estando assim disponível a todos os que utilizam a infra-estrutura computacional da organização, para que desta maneira, nenhum usuário possa alegar ignorância a respeito das regras e de possíveis sanções aplicadas quando tais normas forem infligidas.

1.2 – Vulnerabilidades

Vulnerabilidades são falhas (*bugs)* nos softwares que na maioria das vezes geram deficiências na segurança do computador ou da rede. As vulnerabilidades também podem ser criadas por configurações incorretas do computador ou de segurança. As ameaças buscam explorar essas vulnerabilidades, o que resulta em possíveis prejuízos para o computador ou dados. A maioria dos ataques aos sistemas computacionais explora vulnerabilidades presentes nestes sistemas.

Atualmente há diversas vulnerabilidades presentes nos sistemas, e a SANS em seu site (www.sans.org), com o artigo titulado *Twenty Most Critical Internet Security Vulnerabilities* (“As Vintes Vulnerabilidades Mais Importantes Segurança da Internet) destaca como as principais vulnerabilidades dos sistemas os seguintes tópicos:

- Instalações default dos sistemas
- Senhas frágeis ou sem conta
- Excesso de portas abertas

- Pacotes não filtrados para endereços corretos de entrada e saída
- Registros de entradas inexistentes ou incompletos
- Programas CGI vulneráveis.

Observa-se que as vulnerabilidades podem ser divididas em alguns tipos como: Vulnerabilidade de Configuração, Vulnerabilidade de Rede, Vulnerabilidade de Software, entre outros. Na seção seguinte será descrito esses tipos de vulnerabilidades com mais detalhes.

A etapa inicial para se tentar garantir os requisitos de segurança de um sistema é a eliminação e/ou prevenção das possíveis vulnerabilidades existentes.

1.2.1 – Vulnerabilidades de Configuração

Com o intuito de evitar a invasão de intrusos os diversos equipamentos de rede devem ser configurados de forma eficiente e restrita. Contudo, a segurança ainda não é prioridade nas organizações e a maioria das ações tomadas tem sido de forma corretiva, ao invés de preventivas. Dessa maneira, diversos equipamentos, como roteadores, *switches*, *gateway*, modems e *firewalls* são erroneamente configurados, permitindo o acesso a usuários e serviços não previstos inicialmente.

A respeito deste assunto é importante destacar uma falha constante cometida por administradores de rede. Muitos destes acreditam que os equipamentos ou softwares de rede vem com suas configurações default corretamente configurados, de forma a garantir a maior segurança da rede ou das máquinas (JUNIOR, 2002).

Partindo dessas premissas, pode-se concluir que não basta uma organização conter diversos softwares e equipamentos projetados para garantir a segurança, se suas configurações não forem corretamente efetuadas.

1.2.2 – Vulnerabilidades de Rede

Devido ao fato do protocolo TCP/IP ter sido desenvolvido sem preocupação em garantir os requisitos de segurança e ser nativamente vulnerável, ele se tornou alvo de diversos ataques ocorridos na Internet. Muito desses ataques obtêm sucessos simplesmente

porque as máquinas aceitam pacotes sem verificar a veracidade do endereço IP de origem, e sem avaliar se é ou não uma fonte confiável (KONRATH, 2001).

Um ataque muito utilizado e que faz uso dessa vulnerabilidade é o IP *Spoofing*. Essa técnica se utiliza da fragilidade e facilidade de alteração do cabeçalho do protocolo IP. Um atacante pode forjar o seu endereço de IP origem e como o receptor acredita fielmente na origem do pacote, o atacante pode através de manipulações do cabeçalho forjar seu endereço para outro host e tentar obter acesso a determinados recursos.

Esse tipo de vulnerabilidade é muito explorado e por esse motivo se faz necessário uma maior atenção, tomando as devidas precauções para evitar esses ataques.

1.2.3 – Vulnerabilidade de software

A maioria dos softwares em uso na atualidade contém erros (*bugs*), em muitos casos estes são desconhecidos dos usuários e até de seus desenvolvedores. Estes erros podem levar à interrupções abruptas no seu funcionamento ou até permitir o acesso não autorizado aos sistemas onde eles executam.

O erro mais comum de desenvolvimento de sistemas é a falta de validação de dados de entrada, que pode dar acesso ao ataque conhecido como *buffer overflow*. Nesse ataque, uma entrada de dados inválida é enviada ao sistema, causando uma execução diferente daquela prevista originalmente. A entrada de dados inválida pode conter código malicioso previamente preparado para iniciar um ataque ao sistema afetado. Esse código pode reescrever regiões de memória destinada a código executável, sendo executada em substituição ao código anteriormente presente.

A possibilidade de execução de código malicioso no computador vítima permite a um intruso iniciar um ataque.

Os problemas nos diversos sistemas, na sua maioria, quando identificados, são solucionados e uma nova versão ou correção é disponibilizada para atualização. O tempo desde a descoberta do problema até sua correção pela atualização deixa o sistema vulnerável. E diversos usuários passam muito tempo sem atualizar seus sistemas. Alguns não fizeram nunca as atualizações necessárias, principalmente quando o software instalado é uma cópia Ilegal, que normalmente tem restrições para atualizações. Mesmo quando não existem as restrições, os usuários tem medo de serem descobertos durante a atualização.

A política de atualização dos sistemas pode ser dificultada, ainda, mais por restrições orçamentárias. Várias empresas ainda utilizam versões de sistemas lançadas há mais de 10 anos, muitas vezes sem suporte por parte dos fabricantes. As atualizações dos sistemas podem ser muito caras para os usuários, que preferem manter os sistemas antigos.

1.2.4 – Ausência de Ferramentas de Segurança

Outro tipo de vulnerabilidade explorado pelos atacantes é a falta da utilização de ferramentas ou softwares de segurança. Apesar de haver no mercado grande número dessas ferramentas, algumas gratuitas, outras necessitam ser alugadas ou compradas, muitos usuários ignoram suas utilizações, deixando suas máquinas ainda mais vulneráveis.

Como vem sendo destacado somente a utilização dessas ferramentas não garantirão a segurança do sistema. Para aumentar ainda mais a segurança deve-se levar em considerações os outros assuntos já discutidos, como uma boa política de segurança, configurações corretas entre outras.

Contudo a falta dessas ferramentas facilita em muito o trabalho de invasão, pois o atacante não terá que se preocupar com sua detecção. Se o objetivo é proteger uma rede, essas ferramentas devem ser instaladas em todas as maquinas pertences a ela, caso contrário a partir de uma maquina vulnerável o invasor pode obter acesso às outras máquinas.

Outro fator que leva a essa vulnerabilidade é que grande maioria dos usuários limita-se a ter um antivírus, nem sempre atualizado, principalmente porque diversas opções disponíveis na forma de shareware (pagas) permitam atualizações apenas por um determinado período de tempo. Demais ferramentas, como firewall, detecção de intrusos e anti-spyware são poucas ou completamente desconhecidas pelos usuários.

1.3 – Mecanismos de Defesas

Nesta seção será descrita os principais mecanismos de defesas utilizados atualmente para tentar evitar possíveis ataques.

1.3.1 – Criptografia

Criptografia é a técnica utilizada para codificar e decodificar mensagens ou dados a serem transmitidos, para que se estes forem interceptados por algum indivíduo não autorizado a acessar o seu conteúdo, este não conseguirá visualizá-lo. O intuito maior desta técnica surge pela necessidade de segurança e privacidade individual e de sistemas computacionais, pois somente o receptor da mensagem ou dos dados será capaz de decodificá-los. (NORTHCUTT et al., 2002).

De acordo com Northcutt (2002), os agentes envolvidos na troca destas mensagens detêm de valores secretos denominados Chaves, para codificar e decodificar as mensagens. Esses valores variam de tamanho, sendo que esta variação interfere diretamente no nível de segurança da codificação da mensagem.

Gomes em seu livro Gestão da cadeia de suprimentos integrada à tecnologia da informação (2004) oferece a seguinte descrição quanto às chaves utilizadas na criptografia: "uma chave é uma cadeia aleatória de bits utilizada em conjunto com um algoritmo. Cada chave distinta faz com que o algoritmo trabalhe de forma ligeiramente diferente." (p. 211).

Existem dois tipos de algoritmos de criptografia: o simétrico e o assimétrico. Em relação à criptografia simétrica a mesma chave é utilizada para codificar e decodificar a mensagem. Esta chave é usada para cifrar os dados e criar um documento novo, totalmente ilegível para quem não possui a senha. Esse método criptográfico apresenta alta velocidade de execução, em razão da matemática aplicada para gerar o texto cifrado não ser complexa. Contudo, a criptografia simétrica não é eficiente, considerando que um possível intruso poderia interceptar uma mensagem cifrada, como também ter acesso à chave e decifrar os dados que deveriam ser seguros. (ALBUQUERQUE, 2002).

Atualmente há diversos algoritmos simétricos disponíveis, como Data Encryption Standard (DES), ROT13, Blowfish e International Data Encryption Algorithm (IDEA).

O algoritmo de criptografia assimétrico utiliza um método de criptografia mais seguro e complexo. Este algoritmo faz uso de duas chaves: uma pública e uma privada. A chave pública é utilizada para codificar os dados e deve ser disponibilizada para que os usuários possam enviar mensagens com segurança para o responsável pela chave privada; esta por sua vez, é utilizada para decodificar as mensagens enviadas. Um dos principais algoritmos baseado no método assimétrico é o RSA, criado por Ron L. Rivest, Adi Shamir e Leonard Adelman.

Para desenvolver algoritmos de criptografia são aplicados métodos matemáticos, geralmente complexos, que conseqüentemente efetuam determinadas transformações nos dados da mensagem.

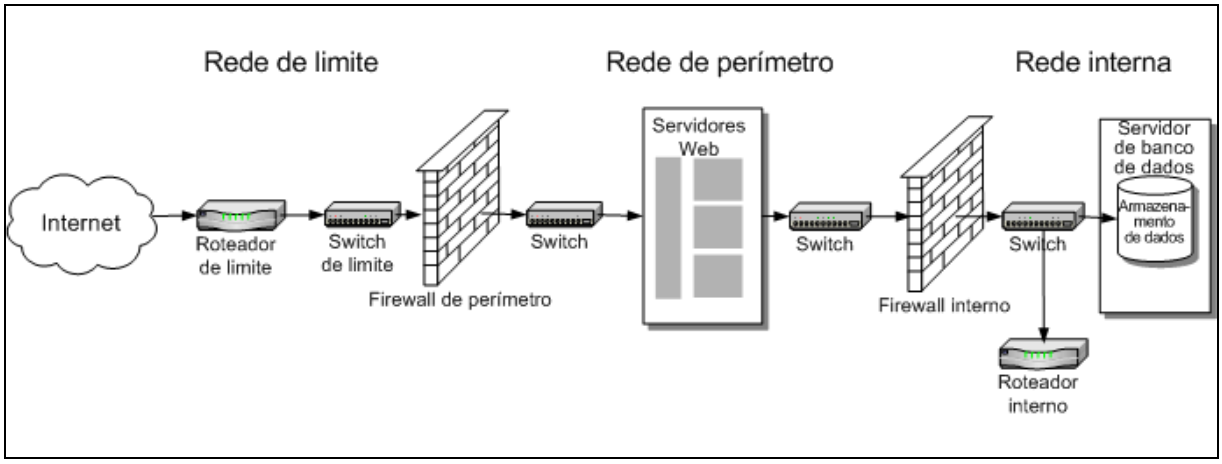
> Um algoritmo é definido na criptografia como uma transformação matemática. Logo, o algoritmo converte uma mensagem clara em uma mensagem cifrada e vice-versa. Quando cifra uma mensagem, a origem utiliza um algoritmo de ciframento para transformar o conteúdo claro da mensagem em texto cifrado. Quando decifra uma mensagem, o destinatário utiliza o algoritmo de deciframento correspondente para converter o texto cifrado de novo em uma mensagem clara. (Gomes, 2004. p. 210).

De acordo com Albuquerque (2002) o mais inteligente não é tentar desenvolver um algoritmo novo com o intuito de aumentar a segurança, mas sim utilizar os algoritmos públicos mais conhecidos, atribuindo maior importância na escolha da chave que será utilizada, sabendo-se que o segredo da criptografia está neste valor e não no algoritmo.

1.3.2 – Firewall

O Firewall é um dispositivo que contém regras para controlar o tráfego na rede, permitindo ou negando o acesso aos dados ou sistemas. O termo Firewall é utilizado na terminologia das redes para relacionar com o firewall (parede contra fogo) utilizado nos edifícios (TITTEL, 2003). Em um edifício a idéia de firewall é evitar que o fogo se propague para outros apartamentos, da mesma forma que em uma rede de computadores o Firewall tem o objetivo de impedir que dados não autorizados trafeguem por sua rede (Figura 1).

Figura 1 – Arquitetura de Firewall



> O Firewall cria uma parede entre a rede pública e a privada, dessa forma é capaz de proteger o acesso não autorizado às informações.

> Os computadores na rede privada não estão diretamente expostos ao mundo "externo". Qualquer tentativa maliciosa para acessá-los exigiria passar pelo firewall. Portanto, um firewall atua como um buffer entre uma rede privada e uma rede pública como a Internet. Em outras palavras, um firewall protege uma rede do acesso não-confiável (Scrimger, 2002. p. 255).

Um requisito importante que um Firewall precisa apresentar é a confiabilidade, pois havendo a possibilidade de um comprometimento seria como se não houvesse Firewall instalado.

Grande parte dos especialistas em segurança indica o uso de um sistema dedicado somente para o Firewall, ou seja, sua execução deve ser realizada em um sistema à parte. A exemplo desta questão, especialistas apontam que se deve evitar a execução do Firewall em servidores WEB da empresa, servidores de dados ou qualquer outro aplicativo.

Contudo, não se pode afirmar que sua rede estará totalmente segura simplesmente por ter um Firewall instalado (CHESWICK, 2005). É necessário definir uma arquitetura de rede eficiente e com outras ferramentas de defesa instaladas, como IDS (Intrusion Detection System) e Antivírus, para que assim estas ferramentas possam trabalhar em conjunto tornando a rede mais segura.

1.3.3 – Sistema de Detecção de Intrusão

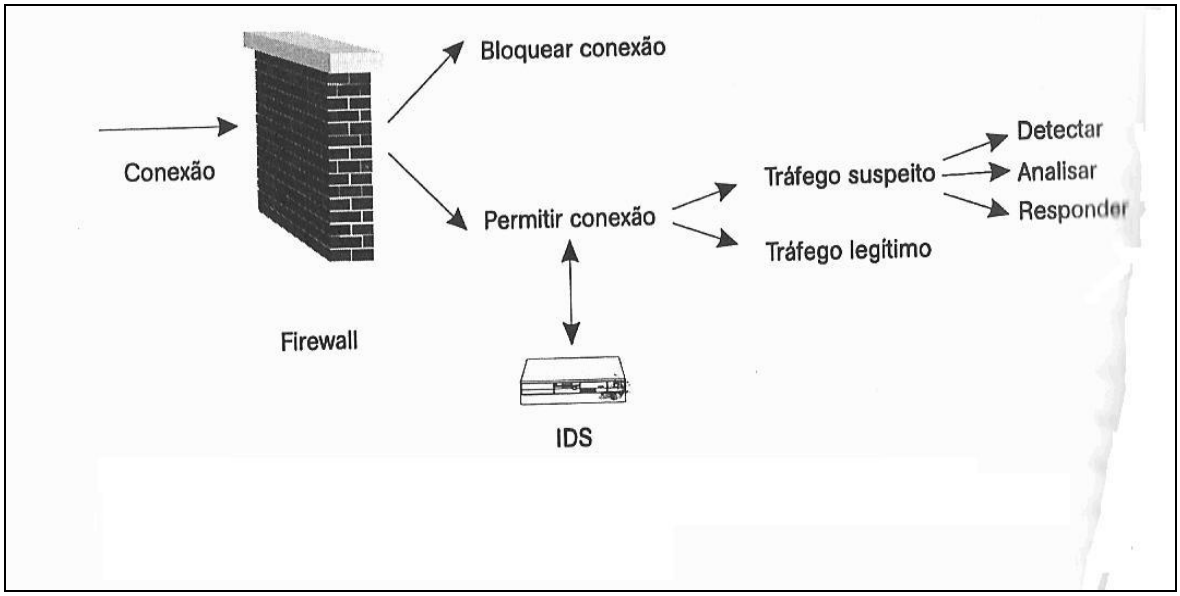
Sistemas de Detecção de Intrusão (IDS – Intrusion Detection System) são aplicativos responsáveis pelo monitoramento de uma rede ou host e pela detecção de qualquer comportamento suspeito baseado em sua configuração, sendo assim, capaz de detectar um ataque ou uma invasão (NORTHCUTT et al. , 2002).

Dois conceitos importantes a respeito de IDS são: falsos positivos e falsos negativos. Falsos positivos ocorrem quando o IDS identifica pacotes normais como possíveis tentativas de ataque, gerando dessa maneira arquivos de log com muitos falsos positivos, evitando que o administrador da rede perceba um verdadeiro ataque.

Falsos negativos, por sua vez, ocorrem quando o IDS não detecta possíveis ataques, identificando esses pacotes intrusos como normais. Uma situação como esta pode apresentar problemas mais graves do que os causados pelos falsos positivos.

Os IDSs não impedem os ataques de forma direta, eles simplesmente alertam os administradores a respeito de comportamentos anormais (Figura 2), sendo da competência dos responsáveis as devidas medidas contentoras.

Figura 2 – Funções do IDS na arquitetura do sistema



O IDS pode ser classificado de duas maneiras:

1. Quanto ao sistema monitorado:
    a. Baseado em Rede (NIDS): através de sensores espalhados pela rede monitorada coletam e analisam todos os pacotes endereçados ao segmento de rede.
    b. Baseado em Nós da Rede: da mesma maneira que os NIDS, esses sistemas também capturam e analisam pacotes a procura de comportamentos suspeitos. Contudo, seus sensores, só se preocupam com os pacotes endereçados ao nó de rede no qual pertence.
    c. Baseado em Host (HIDS): fazem o monitoramento de máquinas (hosts), tomando como fonte das informações: auditoria do sistema operacional e logs do sistema.
2. Quanto ao método de análise:
    a. Por Assinatura: baseiam-se em características comuns de ataques bem conhecidos para classificar os dados capturados como normais ou suspeitos.

b. Por Anomalias: criam padrões de uso normal dos sistemas e detectam quaisquer desvios significativos desses padrões como suspeita de possíveis atividades maliciosas.

Como se pode observar existem diversas maneiras para tentar garantir a segurança dos sistemas computacionais. O uso destas técnicas e ferramentas é de grande importância, pois atualmente existem diversas ameaças e ataques contra os computadores e redes de computadores.

CAPÍTULO 2 – AMEAÇAS E TIPOS DE ATAQUES À REDE DE COMPUTADORES

Para tentar garantir a segurança das informações é necessário conhecer os tipos de ataques existentes, observando ainda que esses ataques podem vir de ações externas quanto internas.

> Essas ameaças podem ter origem interna ou externa, pois, ao contrário do que se pensa, nem sempre o principal "inimigo" está fora da rede, como um hacker ou cracker, mas sim dentro dela, como um funcionário mal intencionado ou muito insatisfeito, que geralmente possui livre acesso aos recursos disponíveis e que pode comprometer a integridade e a privacidade de informações estratégicas da empresa, como, por exemplo, através da destruição ou alteração de informações e o envio de informações sigilosas a concorrentes (GUIMARÃES, 2006, p. 15).

Um ataque pode acorrer de diferentes maneiras na tentativa de acesso não autorizado. Segundo Heady (1990) um ataque pode ser formalmente definido como: "qualquer conjunto de ações que tentem comprometer a integridade, confidencialidade ou disponibilidade dos dados e/ou do sistema".

Os ataques podem ser classificados em duas categorias: ataques ativos e ataques passivos (KUROSE e ROSS, 2005).

Ataques ativos são aqueles que, quando conseguem obter alguma vantagem sobre uma vulnerabilidade do sistema, provocam-lhes alterações. Caso ocorram essas alterações no sistema há também a possibilidade de alteração ou destruição de determinados dados.

Os ataques passivos não provocam alteração no sistema, pois geralmente a intenção é o roubo das informações e na maioria dos casos o sistema não detecta o ataque.

De acordo com observações de manchetes e pesquisas na área de segurança, podem ser destacados como os principais tipos de ataques à rede de computadores: vírus, *worms*, *port scanning*, DoS (*Denial of Service*) – negação de serviço e DDoS *(Distributed Denial of Service*) – negação de serviço distribuído.

Esses ataques e outras ameaças aos computadores e redes de computadores serão descritos na sequência dessa pesquisa.

2.1 – Códigos Maliciosos

Vírus, *worms* (vermes) e cavalos de tróia (*Trojans horses*), típicos exemplos de códigos maliciosos, são considerados os maiores problemas dos administradores de redes (CERT 2006).

Os vírus são pequenos programas de computadores (ou fragmentos de programas), normalmente anexados junto a um software ou arquivo, para que possam espalhar-se por computadores. Os vírus não se auto-executam, ou seja, é necessário que o usuário execute o determinado arquivo ou software infectado para que o vírus se propague. Além de serem capazes de se reproduzirem, eles podem também corromper arquivos e sistemas.

A propagação dos vírus se dá através de disquetes, CD-ROM, pen drives, documentos infectados que são executados, de e-mails (com anexos infectados), de programas piratas, da execução de *downloads* de procedência duvidosa (*orkut, msn, Skype*), mais recentemente por meio de comunicação P2P, entre outros. Em todos esses exemplos, faz-se necessária a presença e atuação do ser humano (MOURA *et al.*, 1999).

Alguns vírus são pré-programados para danificar o computador de forma a corromper os programas, excluir arquivos ou simplesmente construídos para transmitir mensagens cujo intuito é chamar a atenção de um número grande de pessoas. Independentemente do intuito, constituem, normalmente, situações que acabam por culminar na queda ou parada temporária do sistema.

Os *worms* podem, por outro lado, ser considerados uma espécie de vírus, que devido a sua forma de reprodução são considerados mais desastrosos que os vírus normais (MOURA *et al.*, 1999).

Por definição os *worms* são programas capazes de se reproduzirem de um computador para o outro, assim como os vírus, mas, diferentemente dos vírus, os *worms* não utilizam um programa como hospedeiro. Eles possuem a capacidade de se propagarem automaticamente no próprio computador ou de computador para computador, realizando assim a infecção (CERT.br, 2006).

Um exemplo de propagação de *worms* é o do Conficker[1], que se propaga por meio de compartilhamento de arquivo e via unidades removíveis, como dispositivos USB, e já infectou mais de 15 milhões de maquinas segundo estatísticas da F-Secure (TERRA, 2009).

[1] Disponível em: http://www.microsoft.com/brasil/protect/computer/viruses/worms/conficker.mspx

Os *worms* são considerados mais perigosos que os vírus comuns, pois, além da não necessidade da intervenção humana, eles fazem uso de situações normais de operação do sistema (por parte do usuário) para efetuarem sua propagação.

Os *trojans*, ou mais conhecidos como cavalos de tróia, são um exemplo de código malicioso que aparentemente se mostra inofensivos. Eles vêem escondidos em cartões virtuais, protetores de tela, fotos. Esses programas além de executarem funções para as quais foi aparentemente projetado, também executam outras funções normalmente maliciosas e sem o conhecimento do usuário (CERT.br 2006). Esse código malicioso costuma executar automaticamente diversas funções como captura de dados, alterações de características e configurações de um sistema ou permitir acesso remoto em uma base cliente/servidor.

Os métodos utilizados para a instalação desses programas no sistema são muito variados, normalmente é feito o uso da Engenharia Social para atrair o usuário a executar o arquivo infectado e eles também podem ser inseridos por meio de acesso não autorizado.

De um modo geral, as formas de se combater os vírus, *worms* e cavalos de tróia são:

- Utilizar (e manter atualizado) antivírus para detectar a presença dos mesmos;
- Não abrir anexos de *e-mail*s de procedência duvidosa;
- Não efetuar *download* de *softwares* e documentos de sites duvidosos;
- Estar atento sempre para as novas ameaças.

Uma vez que as ameaças, atualmente, são produzidas de formas variadas, fazendo uso de técnicas cada vez mais sofisticadas e audaciosas, existe a preocupação em se prover técnicas mais eficientes no combate aos diferentes tipos de ataques e estar sempre atento às atualizações dos sistemas.

2.2 – *Port Scanning* – Varredura de Portas

Port scanning é uma técnica que consiste em contatar através de requisições de conexão TCP ou via datagrama UDP as portas de um sistema e observar quais destas respondem. Com base nas respostas é possível determinar quais serviços são oferecidos e procurar possíveis brechas nessas portas abertas. Dessa maneira, o atacante poderá optar por qual método de invasão utilizará com base na lista de portas que possibilitam conexões (KUROSE e ROSS, 2005).

O *port scanning* analisando as reações do sistema aos eventos gerados na varredura, pode ser capaz também de obter outras informações de interesse do atacante, como por

exemplo, qual o sistema operacional em execução e assim pesquisar suas vulnerabilidades. Atualmente há diversos tipos de varreduras e entre estas podem ser destacadas:

- **Varredura padrão**: essa varredura envia uma requisição TCP SYN para o sistema alvo e dessa forma executa completamente o *handshake* de 3 etapas (SYN, SYN/ACK e ACK). Esse método é pouco utilizado devido à facilidade em detectá-lo.
- **Varredura TCP SYN**: esse tipo de varredura também é iniciado com o envio de um pacote TCP SYN para o sistema alvo, todavia, nesse caso a conexão não é completada. Com base na resposta à solicitação (ACK: se está em modo escuta; RST: caso contrário), o port scanner reconhece o estado da porta e encerra a conexão enviando um RST/ACK. Este método tem a vantagem de dificultar a detecção da varredura.
- **Varredura Stealth Scanning**: nesta variação a varredura não é iniciada com o envio de um pacote TCP SYN, mas sim com um pacote simulando uma conexão já estabelecida. Assim, além de obter informações sobre o estado da porta, ela também coleta informações adicionais sobre o sistema alvo. O objetivo desta varredura é tentar evitar que filtros de pacotes, como Firewall, tornem-na inviável por meio do bloqueio de pacotes TCP SYN.

Seja qual for o tipo de varredura, de um modo geral, elas são consideradas técnicas de ataque que se aproveitam das vulnerabilidades do protocolo TCP/IP (Internet Protocol), que gerencia as conexões de forma automatizada e com um nível de segurança extremamente simplista.

Sendo verdadeira essa premissa de que as varreduras são técnicas oportunistas, um *host* responde a qualquer solicitação de abertura de conexão endereçada às suas portas, sem se quer avaliar se a origem do pedido é confiável. Outra característica muito explorada pelos port scanner é a utilização de portas conhecidas associadas a serviços padrões.

Existem diversos softwares específicos para explorar essas características capazes de enviar pacotes endereçados às portas do host alvo, monitorar as respostas emitidas pelo alvo e gerar relatórios da análise da varredura executada. Entre as ferramentas utilizadas para a realização do port scanning uma das mais populares é o nmap (INSECURE.ORG), tendo sua popularidade advinda do fato de reunir as mais diversas técnicas de varredura.

2.3 – Engenharia Social

A Engenharia social é um conjunto métodos e técnicas que tem como objetivo obter informações sigilosas e importantes utilizando a exploração da confiança das pessoas, de técnicas psicológicas e investigativas, de enganação, entre outras. Esses tipos de ataques podem ser realizados através de envio de e-mails, telefonemas, salas de bate-papo (chats), e até mesmo pessoalmente (ULBRICH, 2006).

Entre as diversas maneiras de se obter informações através de engenharia social, Kevin Mitnick, considerado um dos maiores hackers da história, em uma entrevista dada a revista Época em 15 de setembro de 2003, destaca:

> [...]Em vez de ficar se descabelando para encontrar uma falha no sistema, o hacker pode largar no banheiro um disquete infectado, com logotipo da empresa e uma etiqueta bem sugestiva: 'Informações Confidenciais. Historico Salarial 2003'. É provável que alguém encontre e insira na máquina (Mitnick, 2003, p ND).

Podem ser definidos basicamente dois tipos de ataques: o direto e o indireto.

O ataque direto é caracterizado pelo contato pessoal. Estes contatos podem ocorrer de diversas maneiras como um telefonema, fax, ou até mesmo pessoalmente, e necessitam de um planejamento antecipado e detalhado, um plano de emergência para qualquer eventualidade, e por fim, o atacante deve ter certo dom artístico, isto porque ele deve ser um bom articulador para evitar qualquer suspeita.

O ataque indireto é caracterizado pela utilização de ferramentas de invasão, como trojans, vírus e sites com códigos maliciosos, e de impostura como cartas, e-mails e falsos sites. Normalmente a intenção não é atacar esses usuários, mas sim obter informações da organização a que ela pertence.

O atacante utiliza-se principalmente de duas técnicas para efetuar o ataque: pesquisa e impostura.

No método da pesquisa o atacante faz a coleta de materiais, como relatórios e folha de pagamentos para ter uma visão da organização da empresa. Após o término da pesquisa inicia-se o método da impostura que normalmente se faz por meio de um ataque direto com o atacante passando-se por outra pessoa.

Portanto, não basta a empresa ter uma excelente arquitetura de segurança com diversos softwares para garantir a segurança das suas informações se ela não se preocupar com o que normalmente é a maior vulnerabilidade de um sistema: o usuário.

2.4 – Outras ameaças

Além dessas principais ameaças citadas anteriormente, vale ressaltar outras ameaças que não afetam diretamente o funcionamento do sistema, mas auxilia os invasores a obter informações sobre os usuários.

Na Cartilha de Segurança disponibilizada pela CERT.br (2006) essas ameaças são as seguintes:

- Bots: são programas capazes de se reproduzir através da rede. Eles têm a capacidade de se comunicar com os invasores e estes, por sua vez, podem orientá-los a realizar outros tipos de ataque.
- BotsNet: uma rede de computadores infectados com bots. O atacante responsável por essa rede, às vezes formada por centena ou até milhares de computadores, pode aumentar a potência de seus ataques.
- Backdoor: como o nome já diz são programas que funcionam como uma "porta traseira" e possibilitam que um invasor retorne a um computador comprometido;
- Keylogger: são programas capazes de gravar as ações dos usuários capturando os caracteres que são digitados;
- Spyware: monitora as atividades do sistema e envia para seu proprietário, na maioria das vezes comprometendo a privacidade do usuário;
- Rootkits: conjunto de ferramentas para esconder a presença de um invasor em um sistema;

Observa-se que, por essas ameaças não afetarem diretamente o desempenho do sistema, muitas vezes os usuários não notam sua presença. Por isso, para tentar evitar a infecção por meio dessas ameaças, é imprescindível manter os softwares de segurança sempre atualizados.

A maioria das ameaças e ataques descritos neste capítulo tem a intenção de roubar e/ou destruir informações. Contudo há outros tipos de ataques que não objetivam diretamente

o roubo ou a destruição dos dados, mas apenas tornar o sistema indisponível. Este é o caso do ataque DoS (*Denial of Service)*, que será detalhado no próximo capítulo.

CAPÍTULO 3 – DoS (*Denial of Service*) – Negação de Serviço

Além dos ataques descritos nos capítulos anteriores existe outro tipo que constitui uma ameaça à segurança das redes de computadores, o ataque de negação de serviço.

Essa categoria de ataque tem como finalidade, sugerido pelo próprio nome, tornar impossível a utilização dos recursos de uma rede ou de um determinado *host*. Normalmente esse tipo de ataque sobrecarrega a infra-estrutura sob ataque, ficando esta impossibilitada de realizar os trabalhos legítimos (KUROSE e ROSS, 2005). Constituindo-se em um ataque baseado na sobrecarga da capacidade ou em uma falha não prevista.

Este tipo de ataque tem como objetivo esgotar os recursos do sistema alvo, forçando uma interrupção total ou parcial dos serviços. A capacidade de processamento, de armazenamento de dados e a largura de banda são alguns dos recursos visados pelas técnicas de negação de serviços.

O problema principal está focado no protocolo IP, que é altamente vulnerável a ataques DoS. Além disso, muitas ferramentas de ataques estão disponíveis para o acesso público e são relativamente fáceis de utilizar.

O problema principal deste tipo de ataque é a alta vulnerabilidade do protocolo TCP/IP, a base da Internet. Além de essas vulnerabilidades permitirem este ataque, elas também tornam muito difíceis a detecção da origem do ataque, devido à facilidade de se forjar os endereços IP de origem (McCLURE, 2003). Além disso, existe a imensa facilidade em encontrar na Internet ferramentas disponíveis e de fácil utilização, para efetuar esses ataques.

Mesmo existindo técnicas de negação de serviço com objetivo de atacar computadores pessoais, a maioria dos ataques busca atingirem sistemas maiores, os quais têm objetivos fornecer serviços a um grande número de usuários, como os servidores WEB.

Observa-se também que há outros motivos para existirem essas falhas nos sistemas: erro básico de programadores, falhas na implementação e bugs (erros), desatualizações dos sistemas operacionais, a partir do momento em que passa a oferecer oportunidades para comprometimento do funcionamento do sistema, devido ao fato de não tratar determinados erros. Assim, o invasor parte do pressuposto de que erros existem e que deve efetuar diversos tipos de testes de falhas, até acontecer um erro e o sistema parar sua execução.

Mesmo não sendo um ataque que visa roubar ou destruir informações, o ataque DoS é extremamente preocupante, pois pode causar prejuízos financeiros tornando indisponíveis por tempo indeterminado os serviços oferecidos pelo sistema (ASSUNÇÃO, 2002).

Um ataque DoS pode ainda ser visto como um *worm*, que se prolifera entre servidores infectados procurando novos computadores para se manifestar. Entretanto, como o programa não se autodetecta ele se reinstala consumindo recurso das máquinas já infectadas, exaurindo assim os recursos das máquinas.

Ataques de DoS podem ser lançados contra roteadores de borda, *bastion hosts*, e *firewalls*, sendo que roteadores, servidores DNS (*Domain Name Service*) e *firewalls* costumam ser os alvos preferenciais.

Em ataques direcionados aos equipamentos de redes, o objetivo principal é tirar uma rede ou sub-rede inteira do ar e não somente algumas máquinas específicas. Existem *bugs* em vários tipos de roteadores e em outros equipamentos de conectividade que podem facilitar esses ataques. Ataques aos *firewalls* fazem com que esses percam a função de filtro, deixando passar conexões TCP para qualquer porta, sendo que um ataque DoS pode, e normalmente é utilizado para facilitar uma invasão.

É importante ainda salientar que as técnicas de negação de serviços são frequentemente adotadas como uma etapa intermediária de métodos de ataque mais complexos. Dessa maneira, elas servem como uma armadilha para deixar um *host* (sistema ou servidor) fora do ar a fim de que outro *host* assuma sua identidade ou até mesmo interrompa o funcionamento de um sistema que execute funções de segurança e controle da rede.

Observa-se a existência de três tipos principais de ataques de negação de serviço:

Exploração de falhas: exploram as vulnerabilidades no software do sistema alvo causando falhas em seu processamento ou extinguindo seus recursos.

Flooding: esse é um tipo de ataque muito utilizado. Este consiste em inundar os recursos do sistema alvo, enviando mais informações do que o sistema é capaz de manipular. Dessa maneira o atacante pode ser capaz de tornar exclusiva a conexão da rede do alvo, impedindo assim qualquer tipo de uso destes recursos.

Ataques de negativa de serviço distribuído (DDoS): um dos mais perigosos e devastadores ataques de negação de serviço. Este consiste em montar previamente uma estrutura contendo diversas máquinas, denominadas zumbis, localizadas em qualquer parte do mundo e que, ao comando de um mestre, inicia o ataque baseado no ataque *flooding* sobre o alvo determinado. Estes são ataques mais complexos e eficientes e de difícil detecção, chegando algumas vezes a tornar impossível esta detecção, pelos ataques partirem de qualquer parte do mundo.

CANEDO (2006) destaca como as formas de ataque do tipo DoS mais conhecidas:

SYN Flooding: esse ataque explora o tipo de conexão *three way handshake*. O atacante envia requisições enviando pacotes TCP SYN tentando iniciar uma conexão, porém, o IP de origem é falsificado (IP Spoofing), desta maneira a resposta a solicitação TCP SYN/ACK enviadas pela vítima nunca é respondida. Como são enviadas muitas solicitações e estas nunca são finalizadas, a fila de conexão chega ao seu limite rapidamente e passa a negar novas requisições. Podem-se adotar configurações especiais para tentar minimizar a vulnerabilidade de um sistema a este ataque, uma delas seria reduzir o tempo limite para excluir uma conexão solicitada e não estabelecida.

UDP Packet Storm: neste tipo de ataque um computador faz solicitações através de pacotes de UDP. Como o protocolo UDP não garante a entrega do pacote, o atacante simplesmente envia pacotes de UDP constantes à máquina vítima. Com isso a máquina fica sobrecarregada e consequentemente, não consegue executar as funções para as quais foi previamente designada.

Smurf: considerado um dos mais devastadores ataques DoS, consiste em enviar requisições ICMP *echo request* a endereços de broadcast de redes, forjando o endereço de origem com o endereço da vítima. Assim, as resposta de todas as máquinas da rede será enviada para a máquina vítima. Em razão do elevado número de pacotes enviados à máquina vítima, pode haver congestionamento da rede, impossibilitando sua utilização.

LAND: este tipo de ataque explora a vulnerabilidade dos datagramas IPs. O atacante envia solicitações de conexão, em um procedimento normal o endereço IP de resposta seria o do solicitante, contudo no LAND Attack, o endereço de IP é forjado para que, o endereço de origem e destino sejam o da vítima. Isso passa a produzir um loop e a máquina da vítima recebe os pacotes enviados por ela mesma, levando a uma queda de desempenho ou até mesmo ao travamento do sistema.

Ping of Death: este tipo de ataque gera um *buffer overflow* (estouro de pilha) quando a máquina atacada recebe um datagrama com tamanho além do limite de 65535 bytes. Uma vez que o datagrama ultrapassa o limite de tamanho, ele é fragmentado por não poder ser roteado, chegando à origem na forma de vários datagramas. A partir do momento que o sistema inicia o processo de remontagem acaba por ter queda de desempenho ou a paralisação no sistema.

Pode-se observar que há diversas maneiras de efetuar ataque de negação de serviço. A partir do momento em que os atacantes notaram que poderiam usar outras máquinas para auxiliar e aumentar a potência de seus ataques, este tipo de ataque se tornou ainda mais

devastador. Deu-se então o surgimento do Ataque de Negação Distribuída (*Distributed Denial of Service* – DDoS) que será comentado na sequência.

3.1 – DDoS (*Distributed Denial of Service*) – negação de serviço distribuído

Ataques DDoS são problemas sérios que afetam os usuários de *Internet*, uma vez que consomem os recursos de um *host* ligado à rede que prestam serviços, tais como *e-mail* (SMTP) e páginas *Web* (HTTP).

Basicamente, os ataques DDoS são coordenados por um atacante que de posse de *hosts* dedicados, conhecidos como *zumbis*, lança um ataque coordenado sobre uma rede ou *host* denominado vítima (KUROSE e ROSS, 2005).

Para entender o funcionamento do ataque DDoS é importante definir os atores envolvidos na arquitetura deste ataque, para isso será usada a nomenclatura mais aceita nas literaturas atuais. A arquitetura deste ataque é composta de quatro atores e dois processos. Os atores são: atacante, master, zumbi e vítima. Os processos: cliente e daemon (CANEDO, 2006).

O atacante é o responsável por coordenar o ataque. Este ator possui uma lista com os endereços de todos os masters. O atacante passa os parâmetros para os masters informando o endereço de uma ou de várias máquinas vítimas.

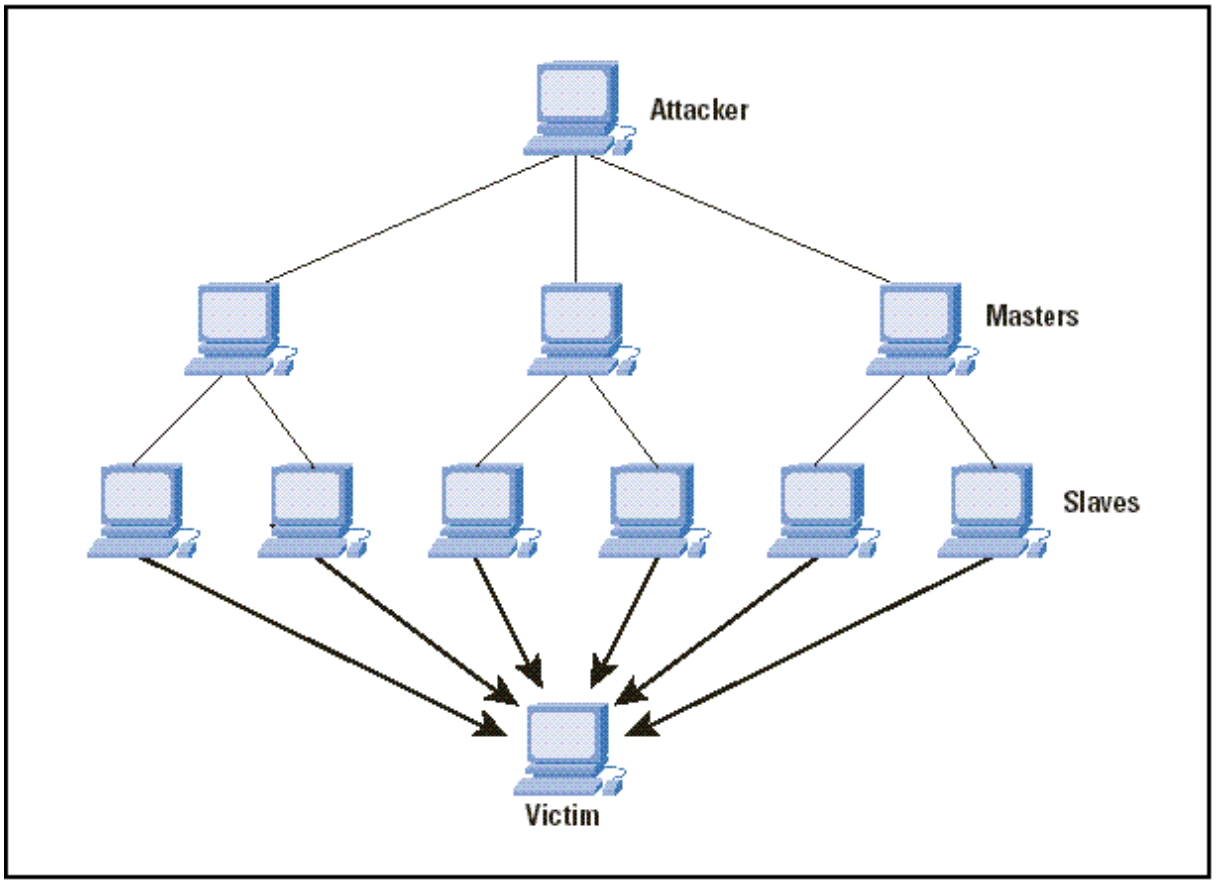
O master é o ator em que o processo cliente é executado e recebe os parâmetros do atacante, podendo manter sob controle centenas ou até milhares de zumbis.

O zumbi é o participante que efetua o ataque diretamente. Ele é o responsável por enviar os pacotes para uma ou mais vítimas de acordo com a especificação do atacante.

A vítima é o ator que sofrerá os ataques, podendo esta ser somente um host ou uma rede. Geralmente são escolhidas vítimas de grande porte que oferecem serviços a milhares de usuários.

A arquitetura do DDoS pode ser observada mais detalhadamente na Figura 3:

Figura 3 – Arquitetura de um ataque DDoS



O DoS tem a característica de realizar o ataque através de uma única máquina. Contudo, com o passar dos tempos observou-se que, utilizando a arquitetura de sistemas distribuídos, a potência dos ataques poderia ser expandida por meio da utilização de uma série de computadores atacantes ao mesmo tempo. Poder-se-ia então obter resultados mais eficientes e desastrosos, a partir desse momento surge a idéia de Ataque de Negação de Serviço Distribuído. Em verdade este ataque potencializa os danos causados pelos ataques de negação de serviço.

Os ataques DDoS ocorrem em três etapas principais: a etapa de intrusão em massa, a etapa de instalação dos softwares DDoS e a etapa de efetivação do ataque.

Na etapa de intrusão em massa, o atacante busca, geralmente através de uma varredura, máquinas vulneráveis que ele possa ter remotamente o controle total. Concluída esta etapa, é gerado um arquivo contendo os endereços de todas as máquinas controladas.

Na etapa seguinte, de instalação dos softwares DDoS, a partir do momento em que o atacante possui o controle das máquinas, inicia-se o processo de instalação dos softwares que

auxiliará a execução do ataque. Nesta etapa são escolhidos os masters e zumbis. Completada essas duas etapas, dá-se início a terceira e última etapa, a de execução do ataque.

Tanto para efetuar como para tentar evitar um ataque DDoS são necessárias ferramentas com um alto nível de sofisticação, integrando recursos avançados que vão desde mecanismos de distribuição automatizados dos módulos clientes até comunicações criptografadas entre os servidores e os clientes.

Recentemente diversos sites têm sido vítimas deste tipo de ataque. Na sequência, podem ser observados alguns ataques que se destacaram.

No mês de junho do ano de 2009 uma onda de ataques de negação de serviço deixou fora do ar os principais sites do governo iraniano[2]. Ataques a sites do governo não são raros, normalmente são manifestos de protestos contra o governo vigente.

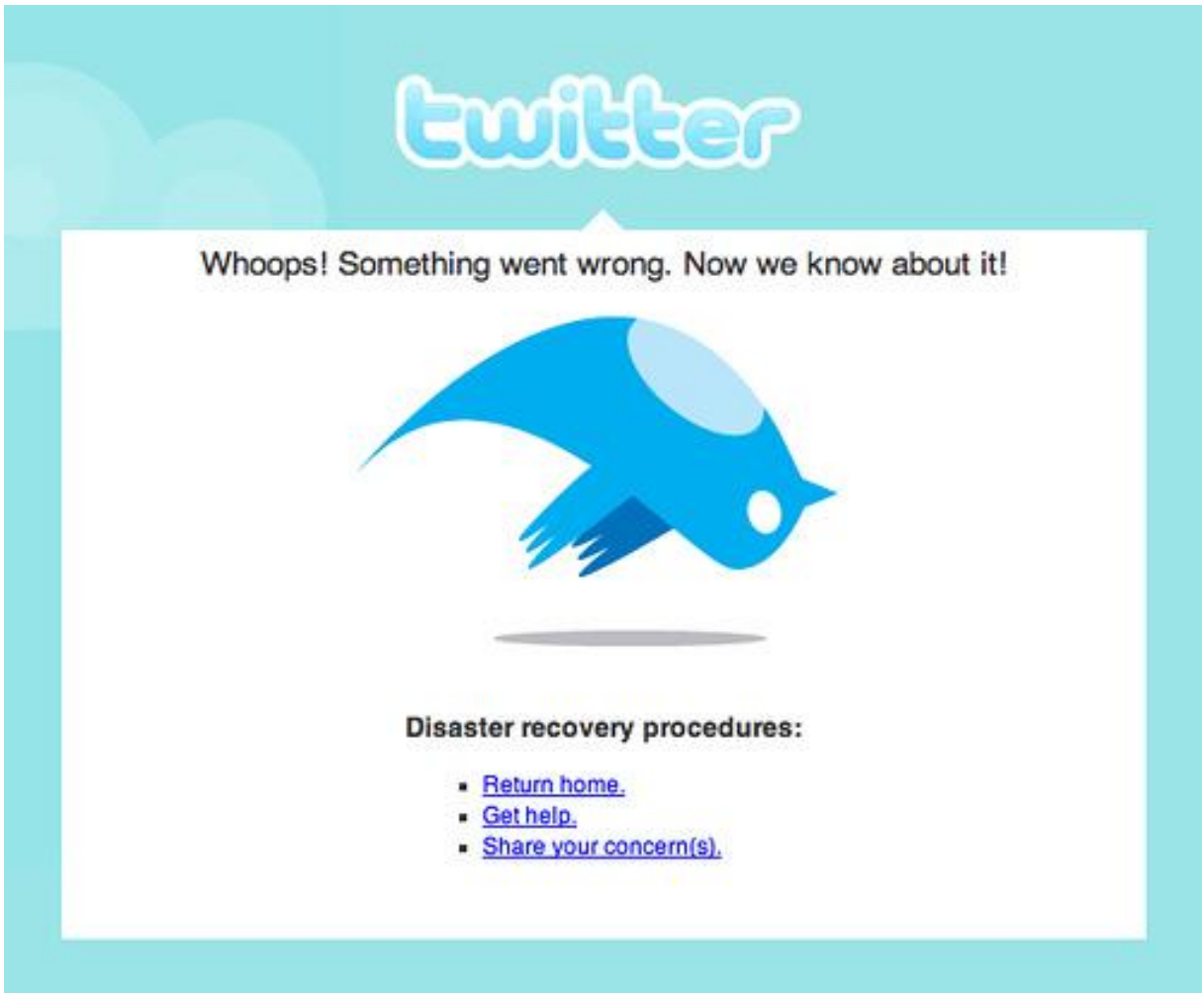
Figura 4 – Ataques DDOS derrubam sites do Irã

Fonte: Site da INFO[6]

[2] Fonte: http://info.abril.com.br/noticias/ti/ataques-ddos-derrubam-sites-do-ira-16062009-24.shl

No mês de agosto do ano de 2009 a vítima foi um dos mais famosos sites da atualidade, o Twitter. Na manhã do dia 06 do já referido mês o site ficou fora do ar e em seu blog de informação de status[3], o Twitter divulgou que um ataque de negação de serviço era o responsável pelo problema.

Figura 5– Site do Twitter sofre ataque DDoS



Fonte: Twitter[7]

No mesmo dia em que o Twitter sofreu um ataque DDoS, o site de relacionamento Facebook passou pela mesma situação[4]:

[3] http://status.twitter.com/post/157191978/ongoing-denial-of-service-attack

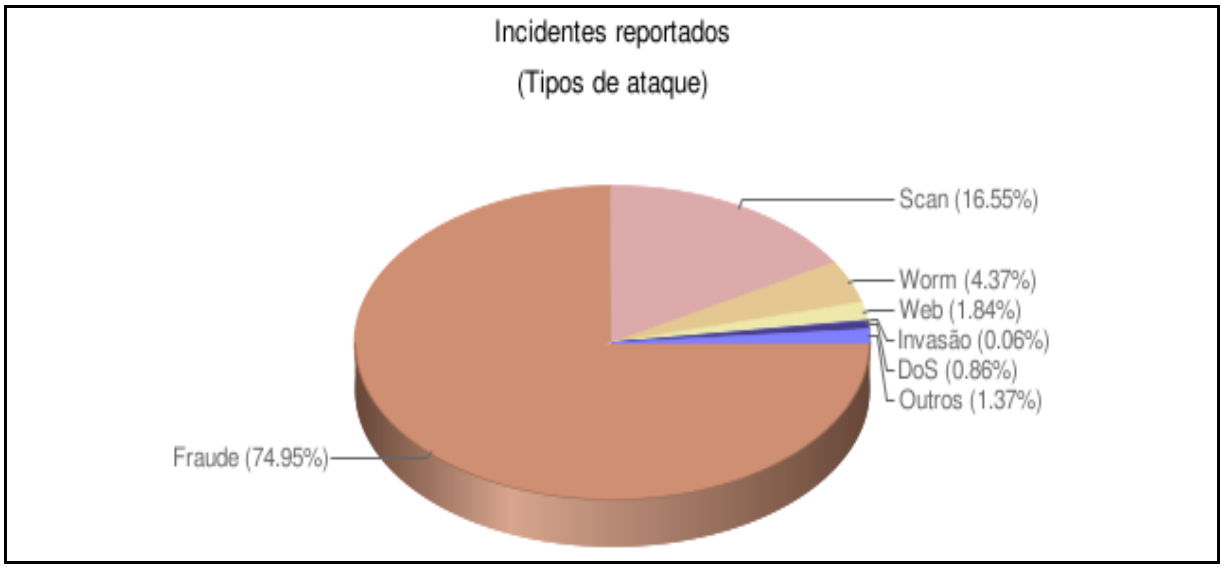
[4] Fonte: http://www.band.com.br/jornalismo/tecnologia/conteudo.asp?ID=163360

Figura 6 – Facebook sofre ataque DDoS

Fonte: Site da BAND[8]

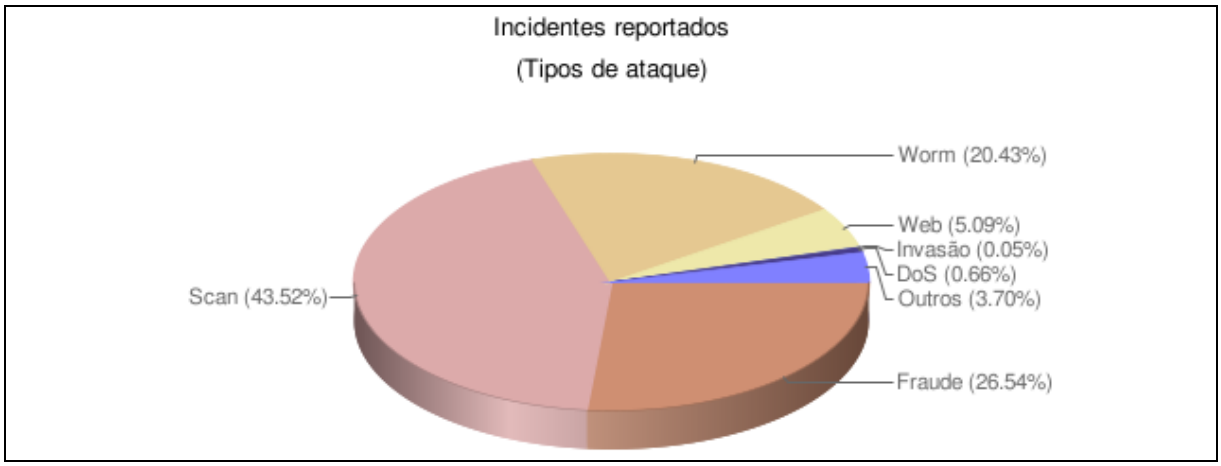
Como se pode observar, de acordo com a frequência de ataques e com base nas estatísticas do CERT (Gráfico 1 e Gráfico 2) houve uma aumento dos ataques DoS no ano de 2009 em relação ao ano anterior (Gráfico 3).

Gráfico 1 – Ataques reportados ao CERT de abril a junho de 2009



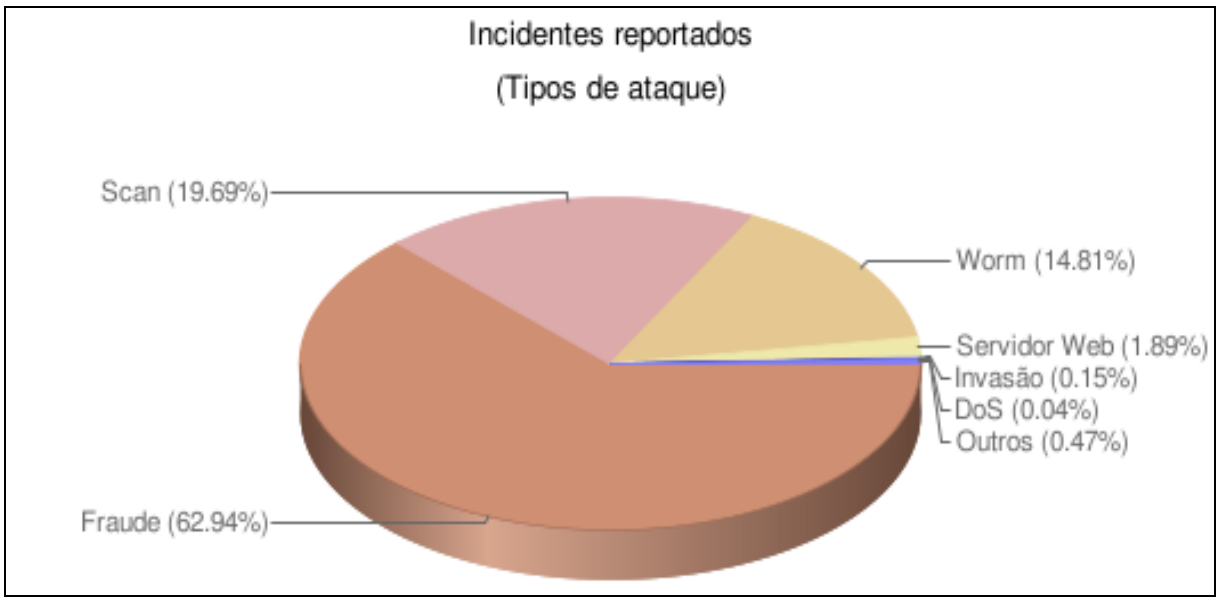
Fonte: http://www.cert.br/stats/incidentes/2009-apr-jun/tipos-ataque.html

Gráfico 2 – Ataques reportados ao CERT de julho a setembro de 2009



Fonte: http://www.cert.br/stats/incidentes/2009-jul-sep/tipos-ataque.html

Gráfico 3 – Ataques reportados ao CERT no ano 2008



Fonte: http://www.cert.br/stats/incidentes/2008-jan-dec/tipos-ataque.html

Como se pode observar estes tipos de ataque ainda são um grande problema para os especialistas em segurança e para os administradores de rede. Dessa forma, os ataques DDoS merecem especial atenção, não apenas pela eficácia, mas também por estabelecer um novo modelo de ataque distribuído. Por isso se faz necessário conhecimento de possíveis vulnerabilidades nos sistemas operacionais para servidores, para que assim, o administrador da rede possa na fase de configuração já se prevenir desses possíveis ataques.

3.2 – Ferramentas de Ataque de Negação de Serviço Distribuído

Atualmente são disponibilizadas diversas ferramentas que têm por objetivos organizar esse ataque de negação de serviço distribuído. Estas ferramentas gerenciam o controle entre os atores envolvidos durante o ataque e, na maioria das vezes, é de uso tão simples que até mesmo usuários leigos são capazes de efetuar ataques contra uma determinada máquina.

Apesar de alguns atacantes preferirem desenvolver sua própria ferramenta de ataque, muitas são disponibilizadas na Internet e podem ser utilizadas por qualquer usuário. Entre essas ferramentas destacam-se: o TFN, TFN2K, o Stacheldraht e o Trinoo (CERT-07, 1999).

Neste capítulo será descrito o funcionamento dessas ferramentas, abstendo-se apenas a descrição do TFN2K para o capítulo 4, no qual serão descritas as ferramentas utilizadas para a realização dos testes. Essas ferramentas são instaladas em alguns hosts que atuarão como servidores (masters). Paralelamente, outros hosts recebem também componentes de software, passando a representar o papel de clientes (zumbis).

As instalações desses softwares são feitas sempre de forma não autorizada, ou seja, o usuário nem as percebe. Na maioria das vezes esses processos vêm anexados a outros arquivos aparentemente inofensivos. Ao comando do atacante, os servidores (*masters*) se comunicam com os clientes (*zumbis*), determinando o início do ataque, seguidos pelos hosts que executam o módulo cliente que lançam ao mesmo tempo uma série de ataques contra o alvo ou os alvos especificados.

3.2.1 – Trinoo

O Trinoo é uma ferramenta que efetua ataques coordenados do tipo UDP *Flooding*. Geralmente uma rede Trinoo é formada por um pequeno número de masters e grande número de zumbis (CERT-07, 1999).

O atacante controla o master por meio de uma conexão TCP via porta 27765/tcp. E os masters se comunicam com os zumbis via pacotes UDPs por meio da porta 27444/udp, podendo essa comunicação também ser feita por pacotes TCP via porta 1524/tcp.

A comunicação entre o master Trinoo e os agentes é feita via pacotes UDP na porta 27444/udpou via pacotes TCP na porta 1524/tcp. A senha padrão para usar os comandos é "l44adsl" e somente comandos que contêm a *substring* "l44" serão processados.

Geralmente, o processo cliente que roda no master tem sido encontrado sob o nome de master, enquanto que os *daemons,* que rodam na máquina zumbi, têm sido encontrados com uma variedade de nomes, entre eles: ns, http, rpc.trinoo, rpc.listen, trinix, etc. Ambos os processos podem ser inicializados sem privilégios de usuário *root*.

3.2.2 – TFN – Tribe Flood Network

A TFN é uma poderosa ferramenta que efetua de ataque DDoS coordenado. Esta ferramenta é capaz de gerar ataques do tipo UDP Flood, SYN Flood, ICMP Flood e Smurf. Uma vantagem desta ferramenta com relação à Trinoo é a possibilidade de forjar o endereço de IP das origens dos pacotes enviados a vítima (CERT-07, 1997).

O atacante acessa os masters por meio de linha de comando, o qual passa como parâmetros o IP da vítima ou vítimas, o tipo de ataque que será efetuado e um arquivo de texto contendo os endereços das máquinas zumbis.

Para executar os comandos do cliente depois de efetuada a conexão não é necessária utilizar senha. Não há comunicação TCP ou UDP entre os clientes e os zumbis, essa é feita por meio de pacotes ICMP_ECHOREPLY.

O processo cliente, geralmente chamado *tribo* e o *daemon* nomeado comumente como ***td,*** devem ser instalados utilizando privilégio de *root*.

3.2.3 – STACHELDRAHT

A Stacheldraht também é uma ferramenta para efetuar ataques DDoS coordenados. Seu desenvolvimento baseou-se nas características principais da TFN e Trinoo. Por esse motivo apresenta características parecidas com estas ferramentas de ataque, porém contendo alguns aspectos adicionais (STAFF, 1999).

Entre as características adicionais destacam-se a criptografia na comunicação entre o atacante e o *master*, e a atualização automática dos agentes.

A necessidade da criptografia surgiu devido ao fato das ferramentas anteriores efetuarem comunicação de forma insegura, em razão das vulnerabilidades do protocolo TCP.

As atualizações automáticas permitiam, por exemplo, que um *daemon* apagasse sua imagem atual substituindo-a por uma nova.

O processo *cliente*, que roda na máquina *master*, é frequentemente encontrado sob o nome ***mserv***, enquanto que o processo *daemon* é habitualmente encontrado com o nome ***lef*** ou ***td.***

Como se pode observar são diversas as ferramentas disponíveis atualmente para se efetuar ataques de Negação de Serviço, além de suas utilizações serem, na maioria dos casos, extremamente fáceis.

CAPÍTULO 4 – TESTES DAS VULNERABILIDADES

Neste capítulo serão descritos os testes realizados, assim como o ambiente construído para esta realização e as ferramentas utilizadas para auxiliar os testes.

O ambiente de teste é composto de 12 máquinas, sendo 11 usadas como escravas e 1 como mestre, com as seguintes configurações:

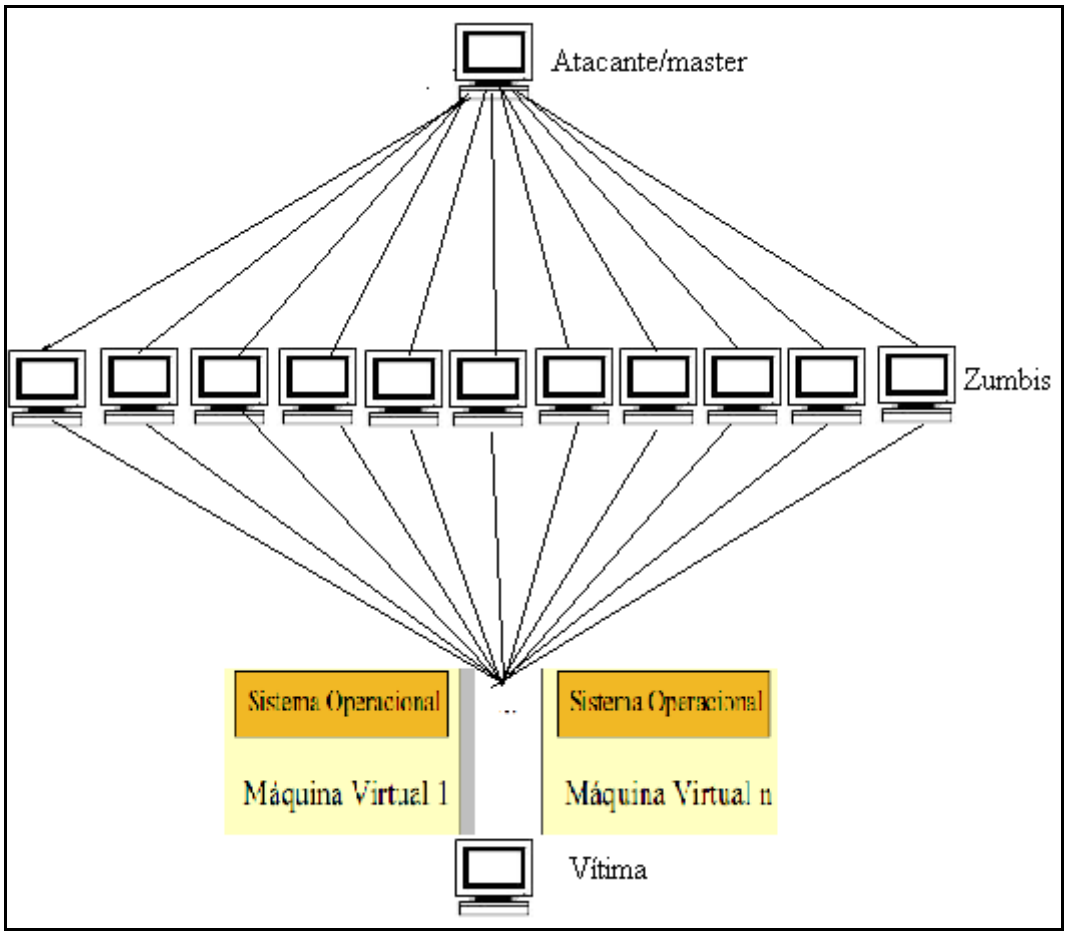
- Sistema Operacional Linux Fedora 9;
- Kernel Linux 2.6.25-14.fc9.i686;
- GNOME 2.22.1
- Memória RAM 494,6 MB
- Processador: Intel Pentium 4 2.80 GHz

A máquina utilizada como vítima tem com a seguinte configuração:

- Sistema Operacional Windows XP Professional
- Memória 958MB
- Processador: Intel Celeron M 1.46 GHz

Esta máquina será utilizada para virtualizar os Sistemas Operacionais para servidores que serão testados. A Figura 7 exibe a arquitetura do ambiente construído para desenvolver o trabalho.

Figura 7 – Arquitetura do ambiente de teste



4.1 – Virtualização

O termo virtualização é utilizado desde os tempos dos *mainframes* e é definido como a possibilidade de executar simultaneamente dois ou mais ambientes distintos e isolados em uma mesma máquina física[5]. Atualmente as principais vantagens que se busca com a virtualização são segurança, confiabilidade e disponibilidade, custo, balanceamento de carga e suporte a aplicações legadas.

De acordo com Mattos inicialmente deve-se definir dois conceitos em relação à virtualização: instruções privilegiadas e não privilegiadas.

As instruções não-privilegiadas são aquelas em que os recursos compartilhados por diversos processos não tem sua alocação ou estado alterados.

[5] Disponível em: http://info.vmware.com/content/GLP_PTBR_Virt_LP1

As instruções privilegiadas são o oposto, ou seja, podem modificar a alocação e o estado desses recursos.

Há dois modos distintos que um computador pode operar: o modo de usuário ou o de supervisor.

O modo de usuário, comumente denominado de espaço de aplicação, é o modo no qual as aplicações geralmente são executadas. Neste modo, é possível executar somente as instruções não-privilegiadas.

O modo de supervisor exerce controle total sobre a CPU, tendo a possibilidade de executar todas as instruções do processador em questão, tanto as não-privilegiadas como as privilegiadas. O sistema operacional é executado no modo de supervisor. Para não haver conflito, os sistemas possuem um bit de controle, deste modo antes de o sistema operacional passar o controle da CPU para uma aplicação do usuário, este bit é configurado para o modo de usuário.

A respeito do ambiente virtualizado, é preciso definir mais dois conceitos, o de sistema operacional hospedeiro e o de sistema operacional visitante.

O sistema operacional hospedeiro (Host Operating System) diz respeito ao sistema operacional nativo da máquina física na qual ocorrerá a virtualização, ou seja, este é o sistema operacional executado diretamente sobre o hardware físico.

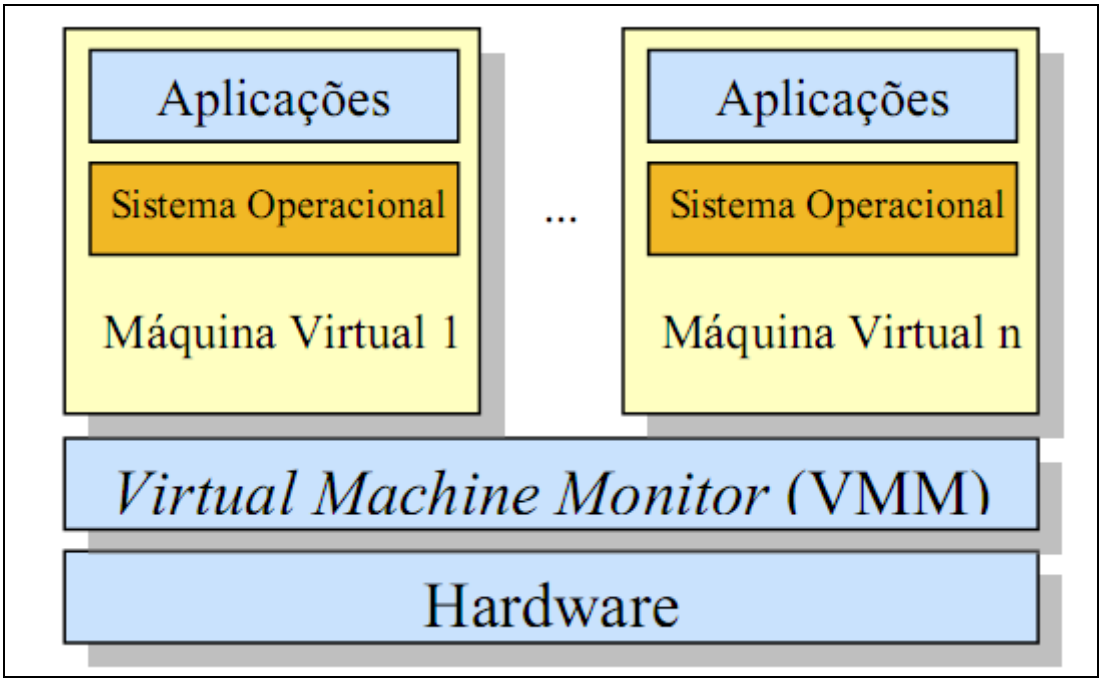
O sistema operacional visitante (Gest Operating System), diz respeito ao sistema operacional que é executado sobre o hardware virtualizado, isto é, o sistema operacional executado na máquina virtual.

A máquina na qual é efetuada a virtualização pode contar com apenas um SO hospedeiro sendo executado de cada vez. No entanto, podem ser executados diversos SOs visitantes simultaneamente.

Um conceito que merece maior atenção para o melhor entendimento da virtualização é o do *Virtual Machine Monitor* (VMM- Monitor de Máquina Virtual), também chamado de *Hypervisor*.

O *Virtual Machine Monitor* é um componente de software responsável por hospedar as máquinas virtuais (Figura 7). O VMM é responsável pela virtualização e controle dos recursos compartilhados pelas máquinas virtuais, entre eles processadores, dispositivos de entrada e saída, memória, armazenagem. Outra função do VMM é escalonar qual máquina virtual será executada a cada momento, semelhante ao escalonador de processos do Sistema Operacional.

Figura 8 – Relação das Máquinas Virtuais e o VMM



O VMM é executado no modo de supervisor, no entanto as máquinas virtuais são executadas em modo de usuário. As máquinas virtuais são executadas em modo de usuário, quando estas necessitam executar uma instrução privilegiada, é gerada uma interrupção e o VMM se encarrega de emular a execução desta instrução.

A virtualização é um recurso interessante que está sendo muito utilizado, mas como qualquer tecnologia possui suas vantagens e desvantagens. Podem ser citadas como suas principais vantagens:

- **Segurança:** podem-se dividir os recursos e processos assim a vulnerabilidade de uma máquina não afeta as demais.
- **Confiabilidade e disponibilidade:** a falha de uma máquina não prejudica as demais.
- **Custo:** consolidação de diversos pequenos servidores em um menor número de servidores, porém mais poderosos;
- **Adaptação às diferentes cargas de trabalho:** é possível tratar as variações na carga de trabalho. Ferramentas autônomas podem realocar recursos de uma máquina virtual para a outra.

- **Balanceamento de carga:** As máquinas virtuais estão encapsuladas no VMM. Dessa maneira é fácil trocar a máquina virtual de plataforma a fim de aumentar o seu desempenho.
- **Suporte a aplicações legadas:** Quando uma empresa decide migrar para um novo Sistema Operacional, é possível manter o sistema operacional antigo sendo executado em uma máquina virtual, o que reduz os custos com a migração. Vale ainda lembrar que a virtualização pode ser útil para aplicações que são executadas em hardware legado, que está sujeito a falhas e tem altos custos de manutenção. Com a virtualização desse hardware, é possível executar essas aplicações em hardwares mais novos, com custo de manutenção mais baixo e maior confiabilidade.

Como desvantagens da virtualização podem ser citados os seguintes itens:

- **Segurança:** como o VMM é uma camada de software está sujeito à vulnerabilidade e se o SO hospedeiro tiver alguma vulnerabilidade todas as máquinas virtuais também estarão vulneráveis.
- **Gerenciamento:** os ambientes virtuais necessitam ser instanciados, monitorados, configurados e salvos;
- **Desempenho:** atualmente não existem métodos consolidados para medir o desempenho de ambientes virtualizados. No entanto, a introdução de uma camada extra de software entre o sistema operacional e o hardware gera um custo de processamento superior ao que se teria sem a virtualização. Outro ponto importante a ser ressaltado é que não se sabe exatamente quantas máquinas virtuais podem ser executadas por processador, sem que haja o prejuízo da qualidade de serviço.

4.2 – Ferramentas Utilizadas

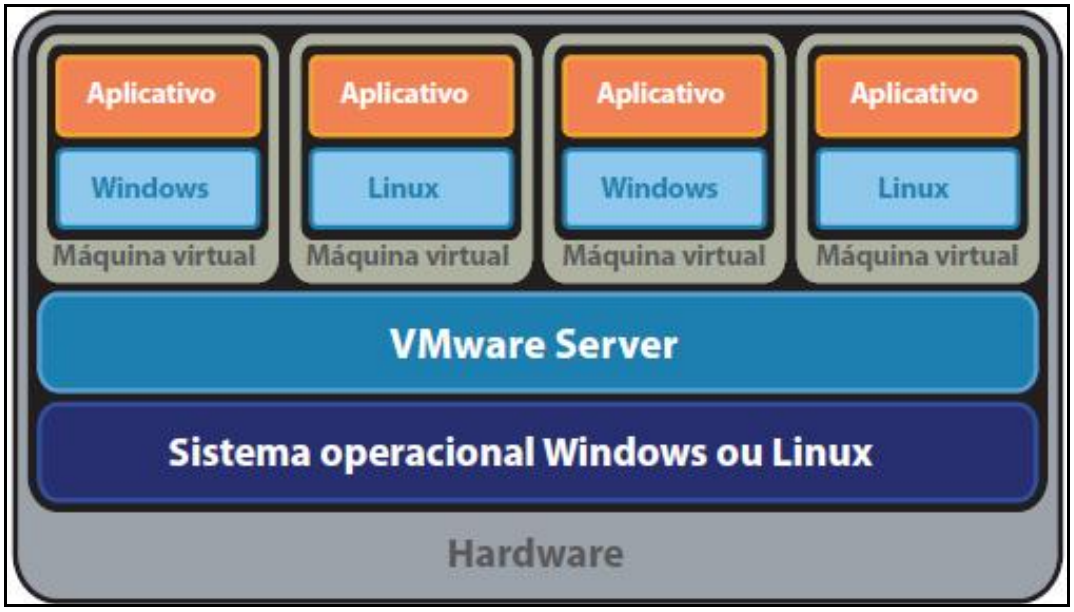
Para a realização dos testes efetuados durante esse trabalho foram necessárias duas ferramentas específicas: VMWare e TFN2K, que serão detalhadas nos tópicos seguintes.

4.2.1 – VMWare

O VMWare é uma das mais populares ferramentas de virtualização, fornecendo software para virtualização desde ambientes desktops a servidores.

O VMWare é executado como se fosse um programa, no espaço de aplicação, dentro de um sistema operacional hospedeiro, o qual fica responsável pela abstração dos dispositivos que serão disponibilizados para o sistema operacional visitante (Figura 8).

Figura 9 – Arquitetura da Execução do VMWare



Os recursos de hardware também são virtualizados. O suporte para os diversos dispositivos é fornecido pelo próprio sistema operacional hospedeiro. Para ter acesso aos dispositivos, o VMWare instala um driver de dispositivo, o VMDriver. Este driver põe a placa de rede em modo promíscuo, recebendo todos os quadros *ethernet*, e cria uma ponte (*bridge*), que encaminha os quadros para o sistema hóspede ou para a máquina virtual especificada.

Entre os produtos fornecidos pela VMWare, pode-se encontrar o VMWare Workstation, Server, Fusion e Player, que são plataformas de virtualização que são executadas em um sistema operacional hospedeiro. Para este trabalho foi utilizado o VMWare Server para executar as máquinas virtuais utilizadas como vítima dos ataques .

4.2.2 – TFN2K – Trible Flood Network 2000

A TFN2K é uma ferramenta utilizada para efetuar ataques DDoS sincronizados (CERT-17, 1999). É uma evolução da ferramenta TFN, que pode efetuar ataque do tipo SYN flood, ICMP flood e Smurf[6]. Essa ferramenta utiliza-se de alguns mestres ou somente um e vários agentes (zumbis). Uma das vantagens dessa ferramenta é que ela torna possível forjar o endereço de IP de origem dos pacotes enviados às vitimas, dificultando assim qualquer tentativa de identificação.

Esta ferramenta foi desenvolvida em linguagem C. No presente trabalho foi utilizada a versão para Linux, porém, sua utilização é idêntica em plataforma Windows. A versão utilizada foi baixada do site da Packet Storm[7]. Depois de baixado o arquivo, geralmente este é copiado para o diretório /usr/src.

Utiliza-se o comando **tar xzvf tfn2k.gz** (note que o arquivo baixado chama-se tfn2k.gz) para extrair os arquivos do pacote e gerar o diretório tfn2k. Entre os arquivos extraídos há um denominado README que traz informações a respeito da ferramenta TFN2K, dentre essas informações estão modo de funcionamento, compilação e instalação.

A maneira mais simples de compilação é simplesmente digitar o comando **make.** Será apresentada uma tela avisando sobre o propósito educacional do software, que irá utilizá-lo para fins educacionais, que os créditos são do autor, entre outras. Em seguida solicitará a senha, de no mínimo 8 e no máximo 32 caracteres, que será utilizada na comunicação criptografada entre os processos, conforme Figura 10.

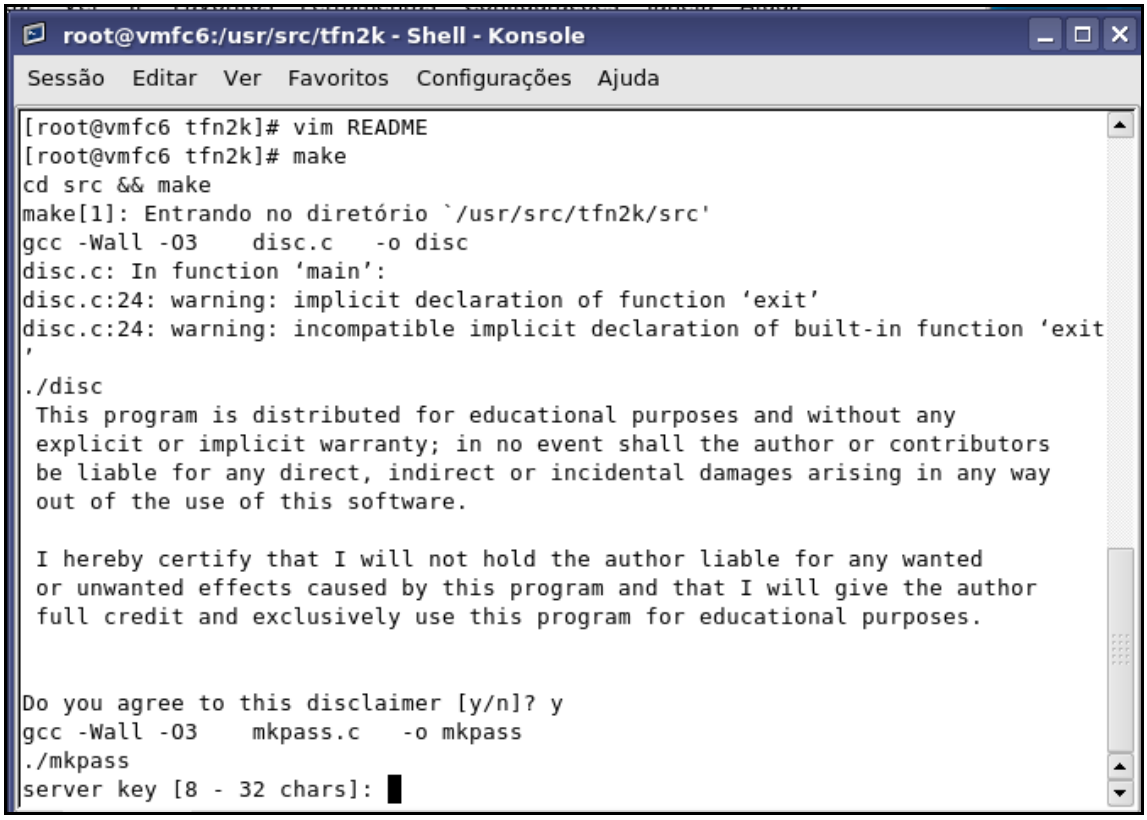
Terminada a compilação são gerados os dois arquivos binários **tfn**, que roda na máquina *master* e ***td***, que roda nas máquinas *zumbis.*

---

[6] InformaBR Segurança da Informação Disponível em: http://www.informabr.com.br/exploits.htm

[7] Disponível em: http://packetstormsecurity.com/distributed/tfn2k. tgz

Figura 10 – Tela de compilação do TFN2K



São diversas as maneira como a atacante fará para o usuário executar o *td*: o atacante pode ter acesso a uma máquina e executá-lo manualmente, ele pode criar sites que contenha links falsos que ao serem clicados iniciem a execução do *td* ou simplesmente por meio de trojans, ou seja, o atacante envia às vítimas arquivos infectados com o *td* e quando esta os executar também iniciará a execução do *td*. Esta é a maneira mais utilizada pelo atacante para que o usuário execute o *td*.

O *master* contém um arquivo com a lista dos IPs de todas as máquinas *zumbis* por ele controladas. Este arquivo é passado como parâmetro no momento da execução do tnf. Dessa maneira o *master* envia a mensagem de ataque a todos os IPs listados no arquivo.

Um dos parâmetros passado para o *master* é o tipo de ataque que será efetuado.

A lista abaixo descreve esses tipos de ataque e seus códigos:

- ID 1 – Anti Spool Level: o ataque ocorrerá com o endereço IP origem *spoofado*.

- ID 2 – Change Packet Size: os ataques ICMP/8, SMURF, e UDP utilizam pacotes de tamanho mínimo por default. Com este comando pode-se determinar o tamanho de cada pacote em bytes.
- ID 3 – Bind root Shell: inicia uma sessão de *root Shell* quando o usuário se conecta à uma porta especificada.
- ID 4 – UDP flood attack: efetua um ataque do tipo UDP Flood.
- ID 5 – SYN flood attack: efetua um ataque do tipo SYN Flood.
- ID 6 – ICMP echo reply (ping) attack: este ataque envia pedidos de ping com IP de origem falsos, para o qual a vítima responde com pacotes de resposta igualmente grandes.
- ID 7 – SMURF attack: envia requisições *broadcast* de *ping* com o endereço de origem da vítima, os hosts que responderem enviarão respostas para o endereço da vítima.
- ID 8 – MIX attack: envia pacotes UDP, SYN e ICMP intercalados com relação de 1:1:1.
- ID 9 – TARGA3 attack: envia pacotes aleatórios com valoeres IPs criticos que podem causar falhas no protocolo IP.
- ID 10 – Remote command execution: executa remotamente comandos no *shell*.

Linha de comando que exemplifica o uso do tfn:

tfn -f zumbis.txt -i 192.168.0.35 -c 8

Explicando as opções utilizadas:

-f especifica o nome de um arquivo texto com os números IPs, um em cada linha do arquivo, das maquinas zumbis.

-i o número IP da maquina vítima do ataque.

-c o ID que especifica o tipo do ataque que deve ser realizado pelas maquinas zumbis.

4.3 – Testes efetuados e resultados obtidos

Nesta seção serão descritos todos os testes efetuados e seus respectivos resultados. Todos os testes foram efetuados utilizando a ferramenta TFN2K e cada um deles durou 3 minutos. Para verificar o desempenho do sistema durante o ataque foram efetuadas

requisições pings e a observação do tempo de resposta e/ou os pacotes perdidos, ou seja, as requisições negadas pelo sistema.

Para uma melhor verificação e detalhamento das possíveis vulnerabilidades, foram efetuados quatro tipos de ataques: ID 4, ID 5, ID 6 e ID 8. Estes ataques foram selecionados por serem os mais significativos de acordo com o objetivo desse trabalho.

Os sistemas operacionais para servidores testados foram:

- Windows Server 2003;
- Windows Server 2008;
- Linux Fedora 5;
- Linux Fedora 6;
- Linux Fedora 9;
- Linux Ubuntu 6;

Todos estes sistemas operacionais foram virtualizados utilizando o software VMWare Server, na máquina vítima descrita anteriormente.

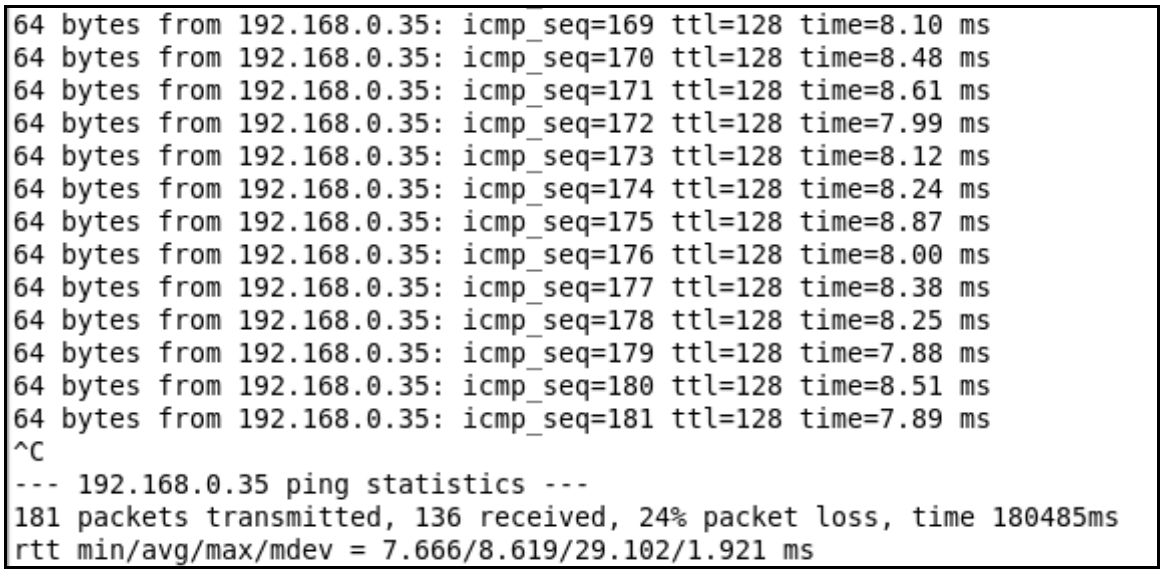
4.3.1 – Ataque contra Windows Server 2003

O primeiro teste foi feito com o Sistema Operacional da Microsoft Windows Server 2003 Enterprise Edition Service Pack 1.

Após a bateria de testes os resultados obtidos foram os seguintes:

Ataque com ID 4: neste primeiro ataque o sistema sofreu uma leve queda de desempenho. Como pode ser visto na Figura 9 o sistema não conseguiu responder a todas as solicitações e as que foram atendidas tiveram o tempo de resposta um pouco elevado.

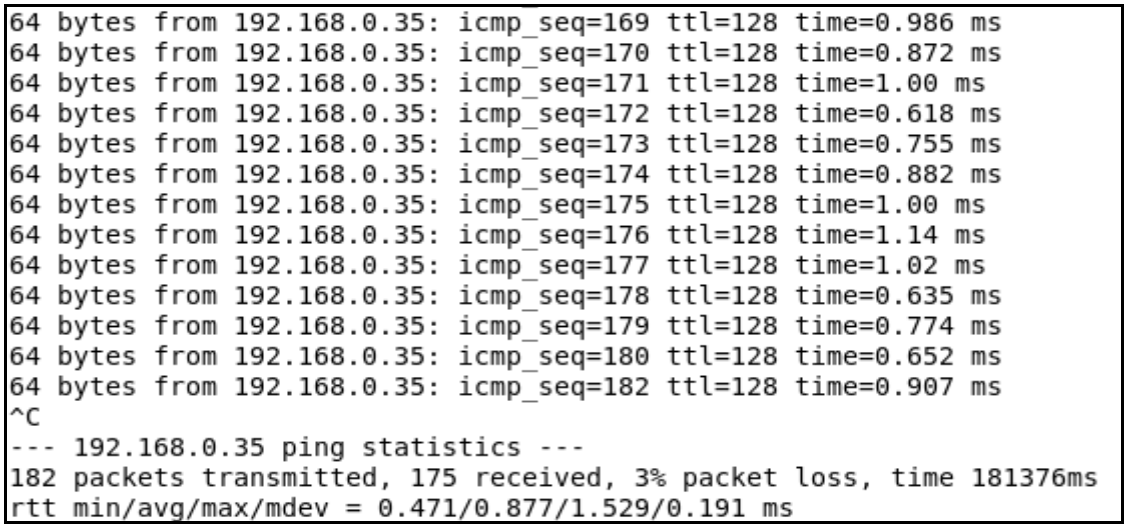
Figura 11 – Resposta do Windows 2003 ao Ataque de ID 4



Ataque com ID 5: Neste tipo de ataque o sistema obteve sucesso, conseguindo tratar com eficiência as requisições e não sofrendo queda de desempenho. Por se tratar de um ataque do tipo SYN Flood, observou-se que o sistema possui boa configuração em relação ao tempo de espera para a confirmação da conexão, consequentemente não torna saturada sua fila de conexão e não apresenta reposta negativa às novas solicitações.

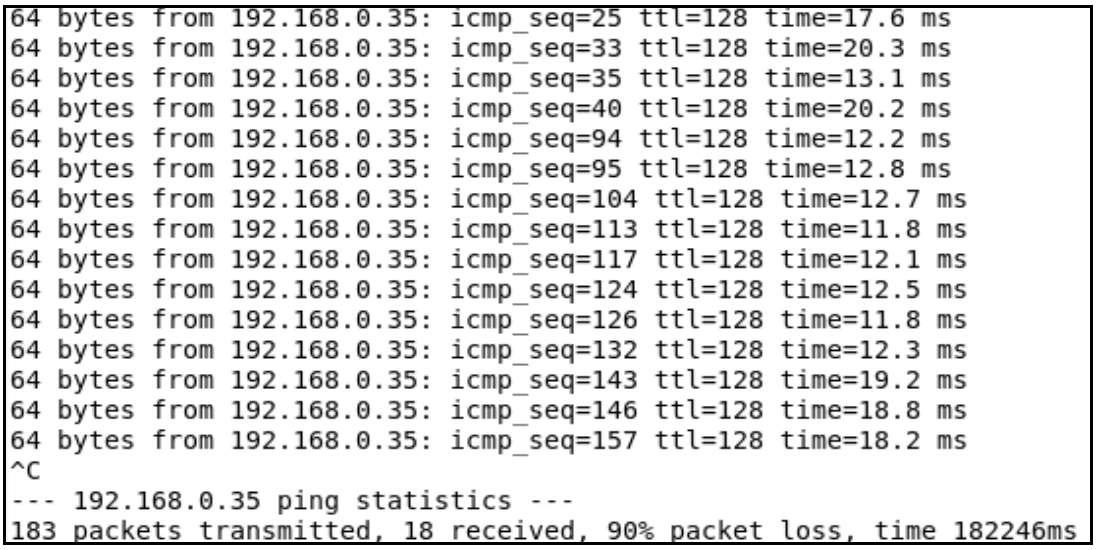
A Figura 10 apresenta 3% de pacotes perdidos, porém esse número pode ser considerado aceitável devido à circunstância.

Figura 12 – Resposta do Windows 2003 ao Ataque de ID 5



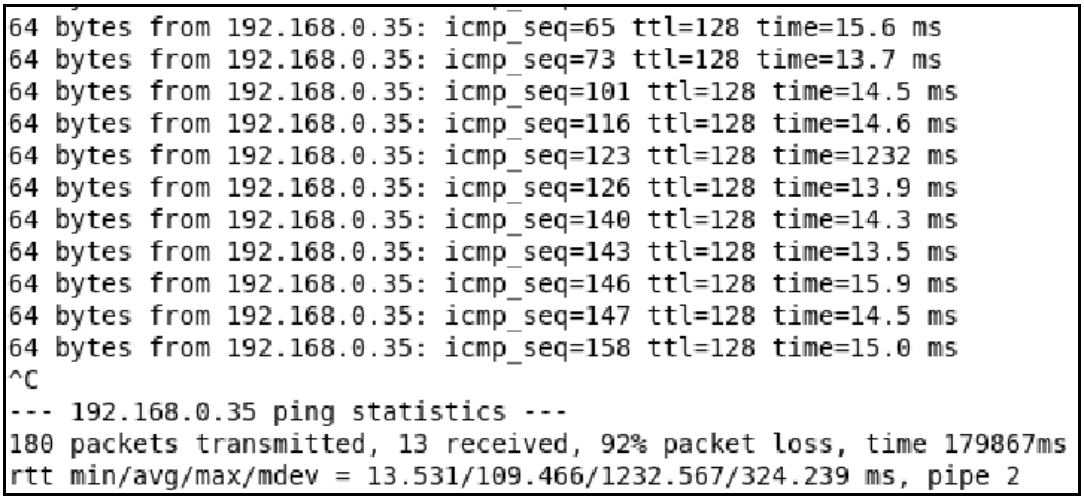
Ataque com ID 6: este ataque obteve sucesso sobre o sistema. Observou-se na Figura 11 que, além da perda de 90% dos pacotes, o tempo de resposta estava alto. O sistema negou serviço à maioria das requisiçoes, atendendo apenas a 18 dentre as 183 solicitações.

Figura 13 – Resposta do Windows 2003 ao Ataque de ID 6



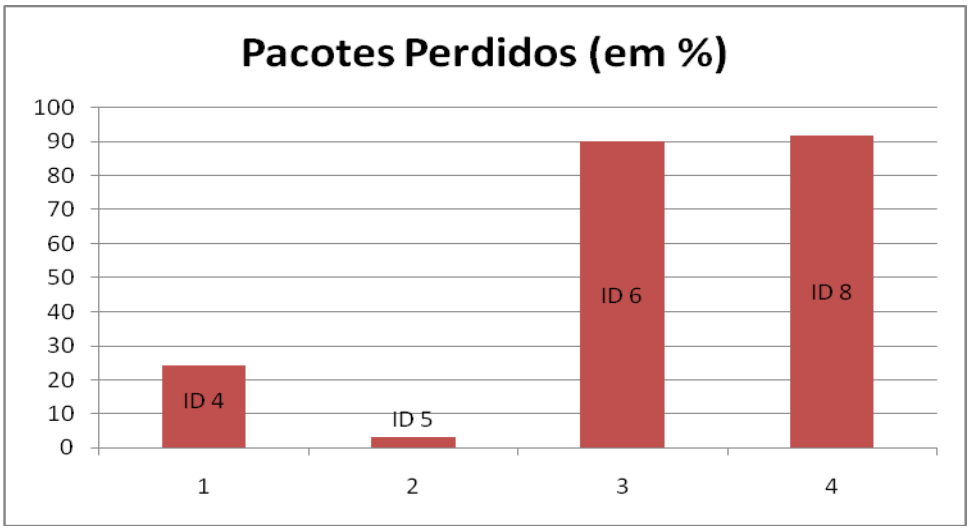
Ataque com ID 8: neste último ataque o desempenho da máquina teve uma queda logo após o primeiro minuto de ataque. De acordo com a Figura 12, 92% dos pacotes foram perdidos e o tempo de resposta ficou alto. Observou-se também que o sistema negou serviço à maioria das requisições.

Figura 14 – Resposta do Windows 2003 ao Ataque de ID 8



O Gráfico 4 apresenta a quantidade de solicitações negadas do sistema em relação a cada ataque. Como pode ser visto os ataques com ID 6 e 8 foram os que obtiveram maior sucesso, levando o sistema a uma queda brusca de desempenho rapidamente e causando posteriormente uma negação de serviço. Por outro lado o ataque com ID 5 foi o que obteve o menor sucesso, não afetando em nada a utilização dos serviços pelos usuários.

Gráfico 4 – Tipo de Ataque x Pacotes Perdidos Windows 2003



4.3.2 – Ataque contra Windows Server 2008

O segundo teste foi realizado com o Sistema Operacional da Microsoft Windows Server 2008.

Após a bateria de testes, os resultados obtidos foram os seguintes:

Ataque com ID 4: neste ataque o sistema apresentou sensível queda de desempenho. Nota-se pela Figura 13 que cerca de 30% dos pacotes foram perdidos e com o tempo de resposta elevado.

Figura 15 – Resposta do Windows 2008 ao Ataque de ID 4

```
64 bytes from 192.168.0.9: icmp_seq=159 ttl=64 time=8.70 ms
64 bytes from 192.168.0.9: icmp_seq=160 ttl=64 time=8.32 ms
64 bytes from 192.168.0.9: icmp_seq=161 ttl=64 time=8.46 ms
64 bytes from 192.168.0.9: icmp_seq=162 ttl=64 time=8.58 ms
64 bytes from 192.168.0.9: icmp_seq=163 ttl=64 time=6.95 ms
64 bytes from 192.168.0.9: icmp_seq=168 ttl=64 time=8.85 ms
64 bytes from 192.168.0.9: icmp_seq=173 ttl=64 time=7.49 ms
64 bytes from 192.168.0.9: icmp_seq=174 ttl=64 time=8.62 ms
64 bytes from 192.168.0.9: icmp_seq=175 ttl=64 time=8.49 ms
64 bytes from 192.168.0.9: icmp_seq=176 ttl=64 time=8.37 ms
64 bytes from 192.168.0.9: icmp_seq=177 ttl=64 time=8.50 ms
64 bytes from 192.168.0.9: icmp_seq=178 ttl=64 time=8.62 ms
64 bytes from 192.168.0.9: icmp_seq=179 ttl=64 time=8.51 ms
64 bytes from 192.168.0.9: icmp_seq=180 ttl=64 time=8.39 ms
^C
--- 192.168.0.9 ping statistics ---
180 packets transmitted, 126 received, 30% packet loss, time 180056ms
```

Ataque com ID 5: ataque sem sucesso, esse não conseguiu fazer com que o servidor negasse serviço. Observa-se na Figura 14 que o tempo de resposta de algumas solicitações teve um leve aumento, porém quase imperceptível ao usuário.

Figura 16 – Resposta do Windows 2008 ao Ataque de ID 5

```
64 bytes from 192.168.0.50: icmp_seq=166 ttl=128 time=0.456 ms
64 bytes from 192.168.0.50: icmp_seq=167 ttl=128 time=1.33 ms
64 bytes from 192.168.0.50: icmp_seq=168 ttl=128 time=36.4 ms
64 bytes from 192.168.0.50: icmp_seq=169 ttl=128 time=0.340 ms
64 bytes from 192.168.0.50: icmp_seq=171 ttl=128 time=1.34 ms
64 bytes from 192.168.0.50: icmp_seq=172 ttl=128 time=62.6 ms
64 bytes from 192.168.0.50: icmp_seq=173 ttl=128 time=26.8 ms
64 bytes from 192.168.0.50: icmp_seq=174 ttl=128 time=0.238 ms
64 bytes from 192.168.0.50: icmp_seq=175 ttl=128 time=0.355 ms
64 bytes from 192.168.0.50: icmp_seq=176 ttl=128 time=19.7 ms
64 bytes from 192.168.0.50: icmp_seq=177 ttl=128 time=40.3 ms
64 bytes from 192.168.0.50: icmp_seq=178 ttl=128 time=0.489 ms
64 bytes from 192.168.0.50: icmp_seq=179 ttl=128 time=0.365 ms
64 bytes from 192.168.0.50: icmp_seq=180 ttl=128 time=0.252 ms
64 bytes from 192.168.0.50: icmp_seq=181 ttl=128 time=19.3 ms
^C
--- 192.168.0.50 ping statistics ---
181 packets transmitted, 177 received, 2% packet loss, time 180750ms
```

Ataque com ID 6: Como pode ser observado na Figura 15 das 181 requisições solicitadas durante o ataque cerca de 95% foram perdidas. Sendo assim o sistema não conseguiu atender a todas as solicitações e passou a negar serviço.

Figura 17 – Resposta do Windows 2008 ao Ataque de ID 6

```
[root@no03 tfn2k]# ping 192.168.0.35
PING 192.168.0.35 (192.168.0.35) 56(84) bytes of data.
64 bytes from 192.168.0.35: icmp_seq=55 ttl=128 time=26064 ms
64 bytes from 192.168.0.35: icmp_seq=84 ttl=128 time=25.2 ms
64 bytes from 192.168.0.35: icmp_seq=112 ttl=128 time=26.5 ms
64 bytes from 192.168.0.35: icmp_seq=117 ttl=128 time=28.7 ms
64 bytes from 192.168.0.35: icmp_seq=118 ttl=128 time=27.8 ms
64 bytes from 192.168.0.35: icmp_seq=121 ttl=128 time=27.2 ms
64 bytes from 192.168.0.35: icmp_seq=136 ttl=128 time=42.5 ms
64 bytes from 192.168.0.35: icmp_seq=147 ttl=128 time=25.7 ms
64 bytes from 192.168.0.35: icmp_seq=149 ttl=128 time=22.8 ms
^C
--- 192.168.0.35 ping statistics ---
181 packets transmitted, 9 received, 95% packet loss, time 180717ms
rtt min/avg/max/mdev = 22.817/2921.293/26064.845/8182.482 ms, pipe 27
```

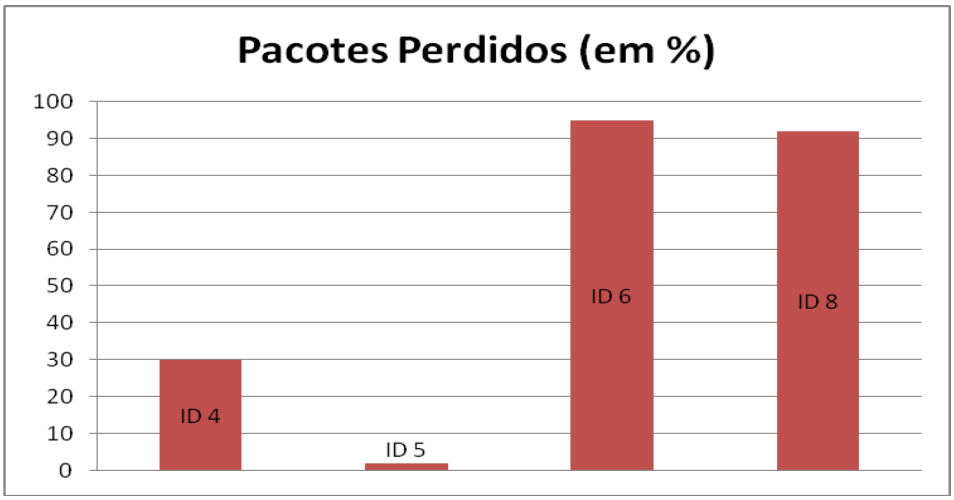
Ataque com ID 8: Como pode ser observado na Figura 16 das 180 requisições solicitadas durante o ataque cerca de 92% foram perdidas. Desta maneira pode considerar que o sistema não conseguiu atender a todas as solicitações negando serviço às proxima requisições.

Figura 18 – Resposta do Windows 2008 ao Ataque de ID 8

```
64 bytes from 192.168.0.9: icmp_seq=50 ttl=64 time=24.6 ms
64 bytes from 192.168.0.9: icmp_seq=51 ttl=64 time=23.7 ms
64 bytes from 192.168.0.9: icmp_seq=53 ttl=64 time=23.5 ms
64 bytes from 192.168.0.9: icmp_seq=55 ttl=64 time=24.0 ms
64 bytes from 192.168.0.9: icmp_seq=56 ttl=64 time=26.4 ms
64 bytes from 192.168.0.9: icmp_seq=57 ttl=64 time=26.0 ms
64 bytes from 192.168.0.9: icmp_seq=58 ttl=64 time=25.9 ms
64 bytes from 192.168.0.9: icmp_seq=60 ttl=64 time=25.1 ms
64 bytes from 192.168.0.9: icmp_seq=62 ttl=64 time=25.6 ms
64 bytes from 192.168.0.9: icmp_seq=67 ttl=64 time=26.3 ms
64 bytes from 192.168.0.9: icmp_seq=105 ttl=64 time=21.9 ms
64 bytes from 192.168.0.9: icmp_seq=179 ttl=64 time=42.9 ms
^C
--- 192.168.0.9 ping statistics ---
180 packets transmitted, 39 received, 78% packet loss, time 179510ms
```

O Gráfico 5 apresenta a quantidade de pacotes perdidos durante cada ataque. Como pode ser visto o ataque com ID 6 e 8 foram os que mostraram maior eficiência, levando o sistema a uma queda brusca de desempenho rapidamente e posteriormente a uma negação de serviço. Por outro lado o ataque com ID 5 foi o que obteve o menor sucesso, aumentando apenas o trafego na rede, mas não afetando em nada a utilização dos serviços pelos usuários.

Gráfico 5 – Tipo de Ataque x Pacotes Perdidos Windows 2008



4.3.3 – Ataque contra o Ubuntu 6

Esta seção apresenta os resultados obtidos com o ataque contra o Sistema Operacional Linux Ubuntu 6.

Os resultados foram os seguintes:

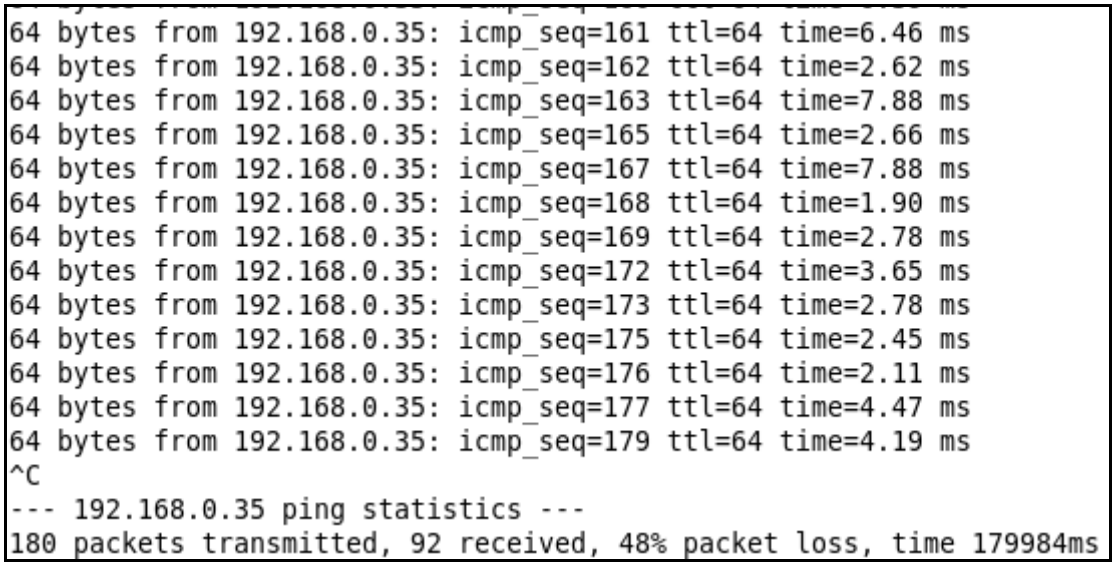
Ataque com ID 4: Observa-se na Figura 17 que o sistema negou serviço a cerca de 83% das requisições e que o tempo de resposta foi alto.

Figura 19 – Resposta do Ubuntu 6 ao Ataque de ID 4

```
64 bytes from 192.168.0.35: icmp_seq=123 ttl=64 time=10.6 ms
64 bytes from 192.168.0.35: icmp_seq=124 ttl=64 time=10.6 ms
64 bytes from 192.168.0.35: icmp_seq=144 ttl=64 time=1011 ms
64 bytes from 192.168.0.35: icmp_seq=151 ttl=64 time=10.8 ms
64 bytes from 192.168.0.35: icmp_seq=152 ttl=64 time=10.9 ms
64 bytes from 192.168.0.35: icmp_seq=156 ttl=64 time=10.8 ms
64 bytes from 192.168.0.35: icmp_seq=160 ttl=64 time=10.0 ms
64 bytes from 192.168.0.35: icmp_seq=161 ttl=64 time=10.2 ms
64 bytes from 192.168.0.35: icmp_seq=164 ttl=64 time=10.8 ms
64 bytes from 192.168.0.35: icmp_seq=167 ttl=64 time=9.98 ms
64 bytes from 192.168.0.35: icmp_seq=171 ttl=64 time=10.2 ms
64 bytes from 192.168.0.35: icmp_seq=178 ttl=64 time=10.6 ms
^P^P^C
--- 192.168.0.35 ping statistics ---
184 packets transmitted, 30 received, 83% packet loss, time 183275ms
```

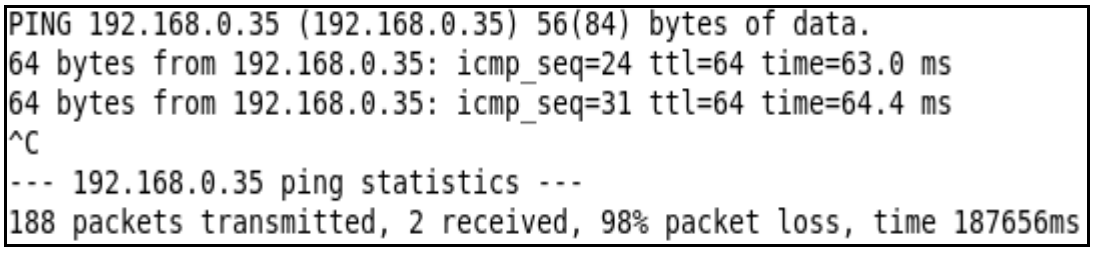
Ataque com ID 5: esse ataque mesmo não sendo tão eficiente conseguiu fazer com que o sistema negasse serviço e perdesse 48% dos pacotes. O tempo de resposta teve uma sensível elevação, porem quase que imperceptível para o usuário.

Figura 20 – Resposta do Ubuntu 6 ao Ataque de ID 5



Ataque com ID 6: esse tipo de ataque levou o sistema a negar cerca de 98% das requisições, alem de ter o tempo de resposta elevadíssimo.

Figura 21 – Resposta do Ubuntu 6 ao Ataque de ID 6



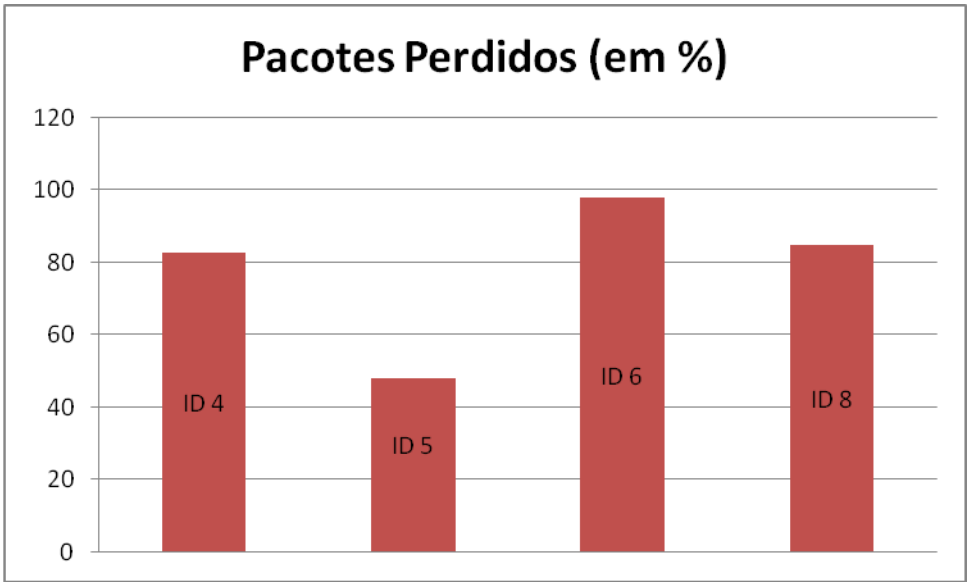
Ataque com ID 8: nesse ataque o desempenho do sistema sofreu uma queda logo após o primeiro minuto de ataque. As respostas a novas solicitações foram ficando lenta. A Figura 20 apresenta que 85% dos pacotes foram perdidos, demonstrando a negação de serviço à maioria das requisições.

Figura 22 – Resposta do Ubuntu 6 ao Ataque de ID 8

```
64 bytes from 192.168.0.35: icmp_seq=43 ttl=64 time=14.7 ms
64 bytes from 192.168.0.35: icmp_seq=54 ttl=64 time=13.6 ms
64 bytes from 192.168.0.35: icmp_seq=58 ttl=64 time=31.4 ms
64 bytes from 192.168.0.35: icmp_seq=67 ttl=64 time=31.5 ms
64 bytes from 192.168.0.35: icmp_seq=71 ttl=64 time=23.8 ms
64 bytes from 192.168.0.35: icmp_seq=73 ttl=64 time=33.3 ms
64 bytes from 192.168.0.35: icmp_seq=78 ttl=64 time=37.7 ms
64 bytes from 192.168.0.35: icmp_seq=79 ttl=64 time=31.3 ms
64 bytes from 192.168.0.35: icmp_seq=80 ttl=64 time=39.9 ms
64 bytes from 192.168.0.35: icmp_seq=88 ttl=64 time=26.9 ms
64 bytes from 192.168.0.35: icmp_seq=90 ttl=64 time=26.2 ms
64 bytes from 192.168.0.35: icmp_seq=97 ttl=64 time=72.6 ms
64 bytes from 192.168.0.35: icmp_seq=100 ttl=64 time=33.2 ms
64 bytes from 192.168.0.35: icmp_seq=113 ttl=64 time=30.9 ms
64 bytes from 192.168.0.35: icmp_seq=120 ttl=64 time=29.8 ms
64 bytes from 192.168.0.35: icmp_seq=129 ttl=64 time=28.9 ms
64 bytes from 192.168.0.35: icmp_seq=131 ttl=64 time=24.0 ms
64 bytes from 192.168.0.35: icmp_seq=179 ttl=64 time=1070 ms
64 bytes from 192.168.0.35: icmp_seq=180 ttl=64 time=70.5 ms
^C
--- 192.168.0.35 ping statistics ---
180 packets transmitted, 27 received, 85% packet loss, time 179461ms
```

O Gráfico 6 apresenta o desempenho o sistema em relação a cada ataque. Como pode ser visto o ataque com ID 6 foi o que obteve maior sucesso, levando o sistema a uma queda brusca de desempenho rapidamente e posteriormente a uma negação de serviço. Por outro lado mesmo o ataque com ID 5 sendo o que obteve o menor sucesso, a utilização dos serviços pelos usuários ficou insatisfatória.

Gráfico 6 – Tipo de Ataque x Pacotes Perdidos Ubuntu 06



4.3.4 – Ataque contra o Fedora 5

Esta seção apresenta os resultados obtidos com o ataque contra o Sistema Operacional Linux Fedora 5.

Os resultados foram os seguintes:

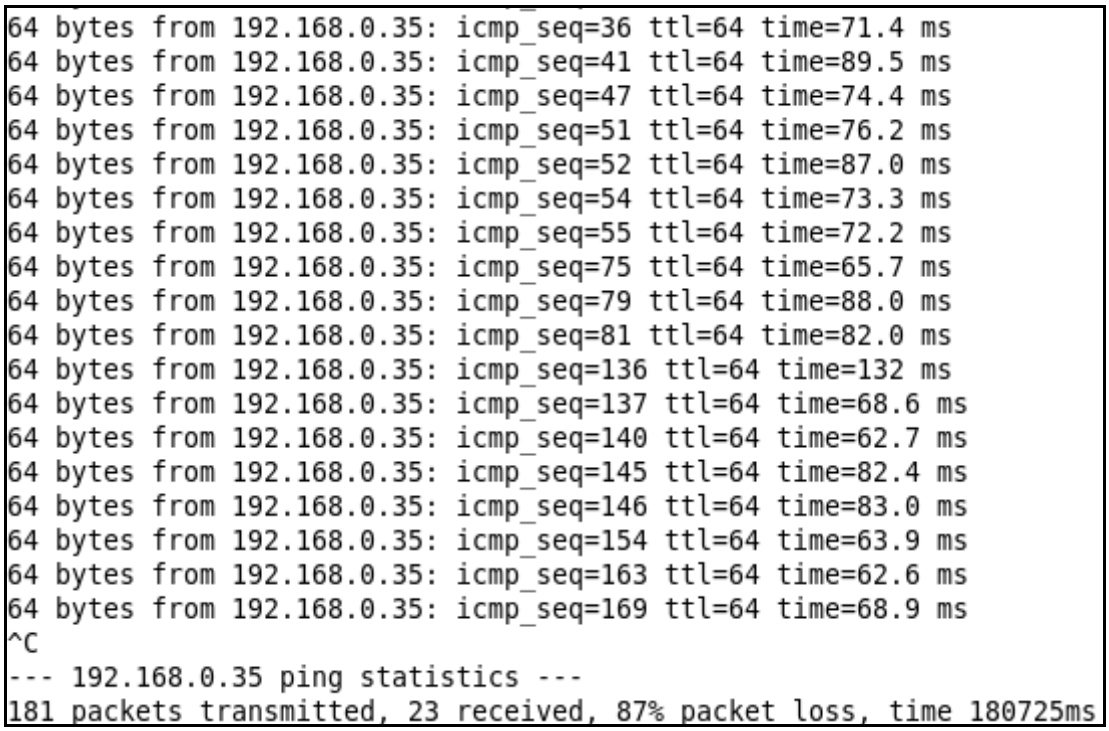
Ataque com ID 4: nesse ataque o sistema sofreu uma queda de desempenho. Como pode ser visto na Figura 21 o sistema negou serviço a 72% das solicitações e, além disso, teve o seu tempo de resposta as requisições elevado.

Figura 23 – Resposta do Fedora 05 ao Ataque de ID 4

```
64 bytes from 192.168.0.35: icmp_seq=126 ttl=64 time=10.5 ms
64 bytes from 192.168.0.35: icmp_seq=131 ttl=64 time=10.9 ms
64 bytes from 192.168.0.35: icmp_seq=133 ttl=64 time=11.4 ms
64 bytes from 192.168.0.35: icmp_seq=136 ttl=64 time=10.5 ms
64 bytes from 192.168.0.35: icmp_seq=138 ttl=64 time=10.6 ms
64 bytes from 192.168.0.35: icmp_seq=141 ttl=64 time=11.9 ms
64 bytes from 192.168.0.35: icmp_seq=145 ttl=64 time=10.4 ms
64 bytes from 192.168.0.35: icmp_seq=147 ttl=64 time=10.7 ms
64 bytes from 192.168.0.35: icmp_seq=149 ttl=64 time=10.5 ms
64 bytes from 192.168.0.35: icmp_seq=154 ttl=64 time=10.6 ms
64 bytes from 192.168.0.35: icmp_seq=156 ttl=64 time=11.1 ms
64 bytes from 192.168.0.35: icmp_seq=157 ttl=64 time=10.7 ms
64 bytes from 192.168.0.35: icmp_seq=165 ttl=64 time=11.0 ms
64 bytes from 192.168.0.35: icmp_seq=176 ttl=64 time=11.2 ms
^C
--- 192.168.0.35 ping statistics ---
181 packets transmitted, 49 received, 72% packet loss, time 180295ms
```

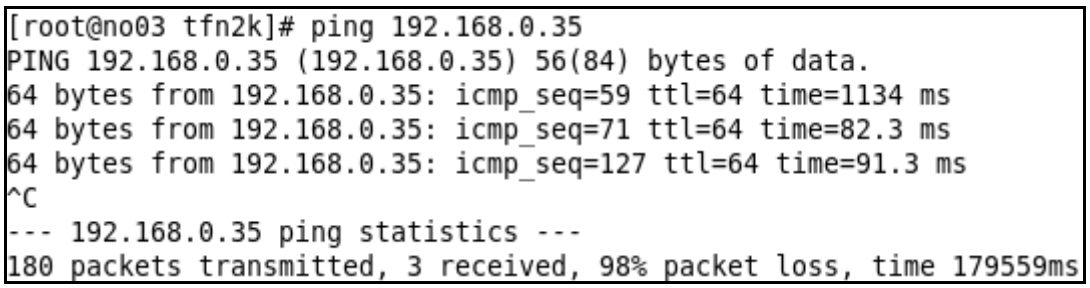
Ataque com ID 5: esse ataque se mostrou eficiente contra essa versão de sistema operacional. Como pode ser observado na Figura 22, o sistema negou serviço a cerca de 87% das requisições e o tempo de resposta foi elevadíssimo.

Figura 24 – Resposta do Fedora 05 ao Ataque de ID 5



Ataque com ID 6: nesse teste o sistema teve uma elevada queda de desempenho. Observa-se na Figura 23 que o sistema negou serviço 98% das requisições, atendendo apenas 3 e ainda com um tempo de resposta muito elevado.

Figura 25 – Resposta do Fedora 05 ao Ataque de ID 6



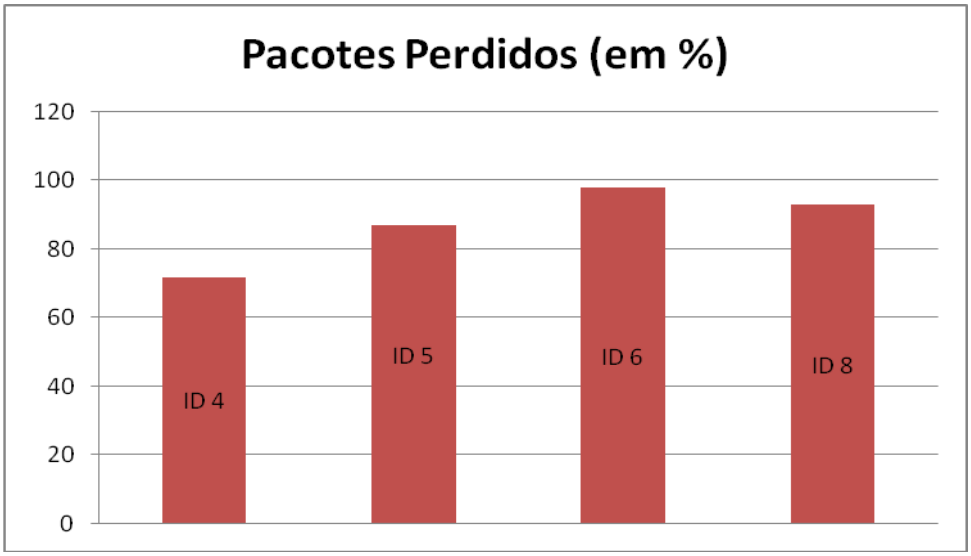
Ataque com ID 8: nesse ultimo ataque o desempenho da máquina teve uma queda logo após o primeiro minuto de ataque. As respostas a novas solicitações foram ficando lenta. Como pode ser observado na Figura 24 aproximadamente 93% dos pacotes foram perdidos e teve-se um tempo de resposta aumentado.

Figura 26 – Resposta do Fedora 05 ao Ataque de ID 8

```
[root@no03 tfn2k]# ping 192.168.0.35
PING 192.168.0.35 (192.168.0.35) 56(84) bytes of data.
64 bytes from 192.168.0.35: icmp_seq=91 ttl=64 time=2075 ms
64 bytes from 192.168.0.35: icmp_seq=100 ttl=64 time=59.1 ms
64 bytes from 192.168.0.35: icmp_seq=101 ttl=64 time=56.5 ms
64 bytes from 192.168.0.35: icmp_seq=102 ttl=64 time=45.9 ms
64 bytes from 192.168.0.35: icmp_seq=109 ttl=64 time=46.0 ms
64 bytes from 192.168.0.35: icmp_seq=114 ttl=64 time=75.1 ms
64 bytes from 192.168.0.35: icmp_seq=141 ttl=64 time=2087 ms
64 bytes from 192.168.0.35: icmp_seq=158 ttl=64 time=52.0 ms
64 bytes from 192.168.0.35: icmp_seq=163 ttl=64 time=53.7 ms
64 bytes from 192.168.0.35: icmp_seq=174 ttl=64 time=46.3 ms
64 bytes from 192.168.0.35: icmp_seq=176 ttl=64 time=46.3 ms
^C
--- 192.168.0.35 ping statistics ---
180 packets transmitted, 11 received, 93% packet loss, time 179707ms
```

O Gráfico 7 apresenta o desempenho o sistema em relação a cada ataque. Como pode ser visto o ataque com ID 6 foi o que obteve maior sucesso, levando o sistema a uma queda brusca de desempenho rapidamente e posteriormente a uma negação de serviço. Por outro lado o ataque com ID 4 foi o que obteve o menor sucesso. Com base nos dados deste gráfico pode-se concluir que os quatro tipos de ataque afetaram consideravelmente o desempenho do sistema.

Gráfico 7 – Tipo de Ataque x Pacotes Perdidos Fedora 05



4.3.5 – Ataque contra o Fedora 6

Esta seção apresenta os resultados obtidos com o ataque contra o Sistema Operacional Linux Fedora 6.

Os resultados foram os seguintes:

Ataque com ID 4: nesse ataque o sistema sofreu uma alta queda de desempenho. Observa-se na Figura 25 que 95% dos pacotes foram perdidos e o tempo de resposta ficou altíssimo.

Figura 27 – Resposta do Fedora 06 ao ataque ID 04

```
[root@no03 tfn2k]# ping 192.168.0.35
PING 192.168.0.35 (192.168.0.35) 56(84) bytes of data.
64 bytes from 192.168.0.35: icmp_seq=38 ttl=64 time=1189 ms
64 bytes from 192.168.0.35: icmp_seq=58 ttl=64 time=140 ms
64 bytes from 192.168.0.35: icmp_seq=77 ttl=64 time=133 ms
64 bytes from 192.168.0.35: icmp_seq=80 ttl=64 time=114 ms
64 bytes from 192.168.0.35: icmp_seq=94 ttl=64 time=2256 ms
64 bytes from 192.168.0.35: icmp_seq=140 ttl=64 time=96.7 ms
^C
--- 192.168.0.35 ping statistics ---
180 packets transmitted, 6 received, 96% packet loss, time 179826ms
rtt min/avg/max/mdev = 96.748/655.430/2256.933/815.672 ms, pipe 3
```

Ataque com ID 5: nesse tipo de ataque a queda de desempenho do sistema foi menor, porém como pode ser observado na Figura 26 o número de pacotes perdidos foi de 53%,uma quantidade considerável.O tempo de resposta teve um sensível aumento.

Figura 28 – Resposta do Fedora 06 ao ataque ID 05

```
64 bytes from 192.168.0.35: icmp_seq=142 ttl=64 time=2.40 ms
64 bytes from 192.168.0.35: icmp_seq=143 ttl=64 time=3.53 ms
64 bytes from 192.168.0.35: icmp_seq=144 ttl=64 time=4.65 ms
64 bytes from 192.168.0.35: icmp_seq=148 ttl=64 time=4.42 ms
64 bytes from 192.168.0.35: icmp_seq=149 ttl=64 time=4.80 ms
64 bytes from 192.168.0.35: icmp_seq=151 ttl=64 time=3.55 ms
64 bytes from 192.168.0.35: icmp_seq=152 ttl=64 time=3.93 ms
64 bytes from 192.168.0.35: icmp_seq=155 ttl=64 time=8.31 ms
64 bytes from 192.168.0.35: icmp_seq=156 ttl=64 time=8.94 ms
64 bytes from 192.168.0.35: icmp_seq=160 ttl=64 time=7.70 ms
64 bytes from 192.168.0.35: icmp_seq=162 ttl=64 time=3.46 ms
64 bytes from 192.168.0.35: icmp_seq=164 ttl=64 time=3.47 ms
64 bytes from 192.168.0.35: icmp_seq=165 ttl=64 time=4.10 ms
64 bytes from 192.168.0.35: icmp_seq=166 ttl=64 time=5.47 ms
64 bytes from 192.168.0.35: icmp_seq=167 ttl=64 time=4.60 ms
64 bytes from 192.168.0.35: icmp_seq=169 ttl=64 time=3.61 ms
64 bytes from 192.168.0.35: icmp_seq=172 ttl=64 time=3.49 ms
64 bytes from 192.168.0.35: icmp_seq=174 ttl=64 time=3.00 ms
64 bytes from 192.168.0.35: icmp_seq=177 ttl=64 time=2.88 ms
^C
--- 192.168.0.35 ping statistics ---
181 packets transmitted, 85 received, 53% packet loss, time 180559ms
```

Ataque com ID 6: o sistema se mostou completamente vulnerável a esse tipo de ataque negando serviço a todas as solicitações, como pode ser visto na Figura 27.

Figura 29 – Resposta do Fedora 06 ao ataque ID 06

```
PING 192.168.0.35 (192.168.0.35) 56(84) bytes of data.
^C
--- 192.168.0.35 ping statistics ---
180 packets transmitted, 0 received, 100% packet loss, time 179701ms
```

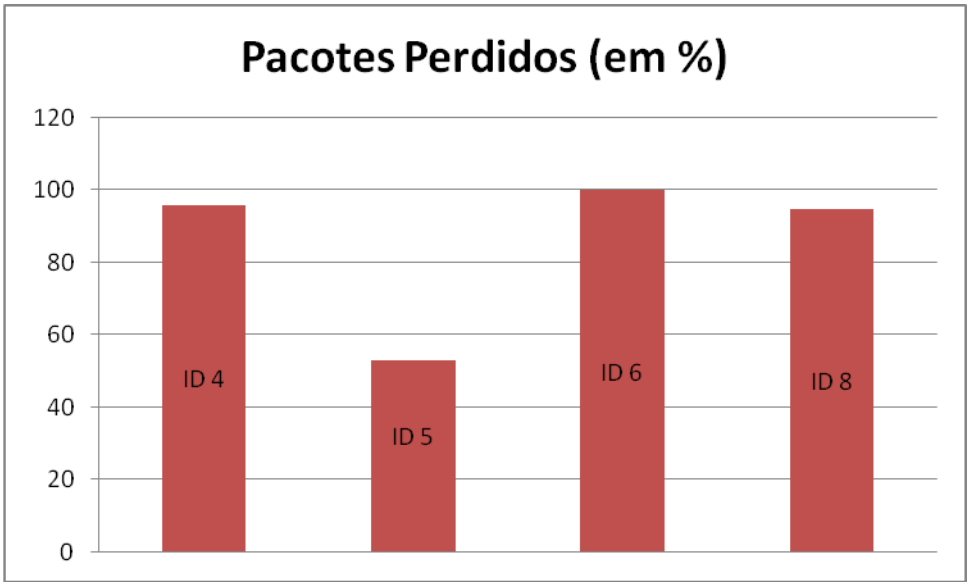
Ataque com ID 8: esse tipo de ataque também se mostrou muito eficiente contra essa versão de SO. Observa-se na Figura 28 que o sistema negou serviço a 95% dos pedidos e que o tempo de resposta ficou elevadíssimo.

Figura 30 – Resposta do Fedora 06 ao ataque ID 08

```
PING 192.168.0.35 (192.168.0.35) 56(84) bytes of data.
64 bytes from 192.168.0.35: icmp_seq=3 ttl=64 time=99.5 ms
64 bytes from 192.168.0.35: icmp_seq=10 ttl=64 time=92.7 ms
64 bytes from 192.168.0.35: icmp_seq=13 ttl=64 time=107 ms
64 bytes from 192.168.0.35: icmp_seq=52 ttl=64 time=179 ms
64 bytes from 192.168.0.35: icmp_seq=67 ttl=64 time=96.1 ms
64 bytes from 192.168.0.35: icmp_seq=68 ttl=64 time=103 ms
64 bytes from 192.168.0.35: icmp_seq=80 ttl=64 time=106 ms
64 bytes from 192.168.0.35: icmp_seq=95 ttl=64 time=92.7 ms
^C
--- 192.168.0.35 ping statistics ---
181 packets transmitted, 8 received, 95% packet loss, time 180993ms
```

O Gráfico 8 apresenta o desempenho o sistema em relação a cada ataque. Como pode ser visto o ataque com ID 6 foi o que obteve maior sucesso, levando o sistema a uma queda brusca de desempenho rapidamente e a negação de serviço a todas as solicitações. Os ataques do tipo ID 8 e 4 também se mostraram bastante eficiente contra essa versão de SO. O ID 5 mesmo sendo o de menor eficiência nota-se um número considerável de serviço negado.

Gráfico 8 – Tipo de Ataque x Pacotes Perdidos Fedora 06



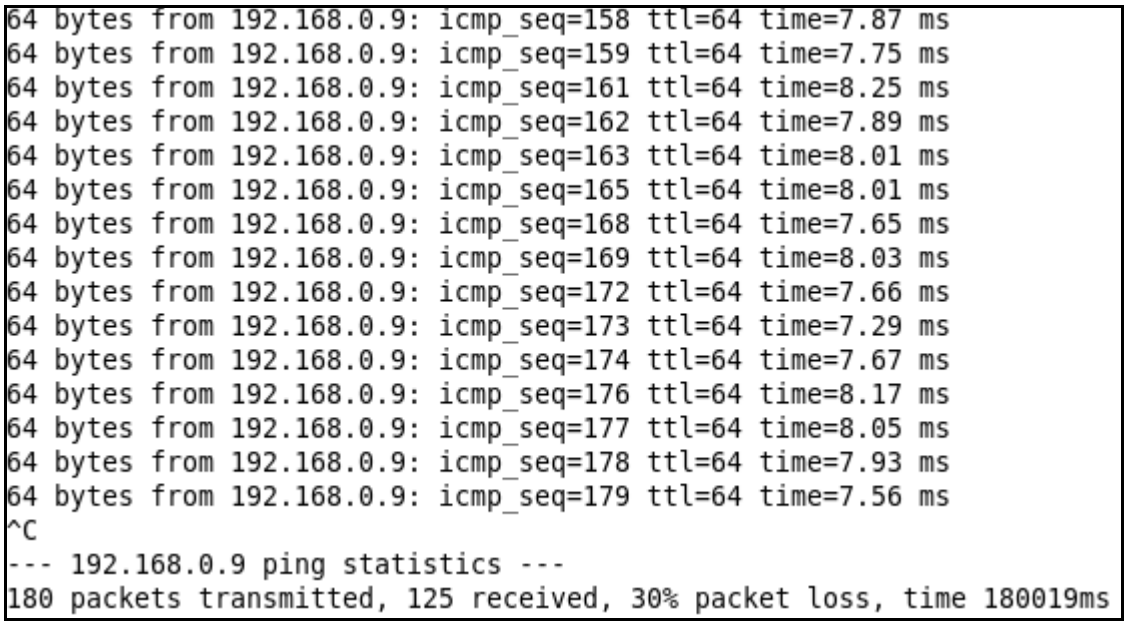
4.3.6 – Ataque contra o Fedora 9

Esta seção apresenta os resultados obtidos com o ataque contra o Sistema Operacional Linux Fedora 9.

Os resultados foram os seguintes:

Ataque com ID 4: nesse ataque o sistema sofreu uma queda de desempenho considerável. Nota-se pela Figura 29 uma sensível elevação no tempo de resposta e a perda de 30% das requisições.

Figura 31 – Resposta do Fedora 09 ao ataque ID 04



Ataque com ID 5: esse ataque não afetou o desempenho do sistema. Como pode ser observado na Figura 30 o sistema negou apenas 1% das requisições,um número aceitável.

Figura 32 – Resposta do Fedora 09 ao ataque ID 05

```
64 bytes from 192.168.0.9: icmp_seq=163 ttl=64 time=4.30 ms
64 bytes from 192.168.0.9: icmp_seq=164 ttl=64 time=1.19 ms
64 bytes from 192.168.0.9: icmp_seq=165 ttl=64 time=0.820 ms
64 bytes from 192.168.0.9: icmp_seq=166 ttl=64 time=2.44 ms
64 bytes from 192.168.0.9: icmp_seq=167 ttl=64 time=1.07 ms
64 bytes from 192.168.0.9: icmp_seq=168 ttl=64 time=4.70 ms
64 bytes from 192.168.0.9: icmp_seq=169 ttl=64 time=5.56 ms
64 bytes from 192.168.0.9: icmp_seq=170 ttl=64 time=5.95 ms
64 bytes from 192.168.0.9: icmp_seq=171 ttl=64 time=1.58 ms
64 bytes from 192.168.0.9: icmp_seq=172 ttl=64 time=1.20 ms
64 bytes from 192.168.0.9: icmp_seq=173 ttl=64 time=3.59 ms
64 bytes from 192.168.0.9: icmp_seq=174 ttl=64 time=0.719 ms
64 bytes from 192.168.0.9: icmp_seq=175 ttl=64 time=3.08 ms
64 bytes from 192.168.0.9: icmp_seq=176 ttl=64 time=3.72 ms
64 bytes from 192.168.0.9: icmp_seq=177 ttl=64 time=2.10 ms
64 bytes from 192.168.0.9: icmp_seq=178 ttl=64 time=0.470 ms
64 bytes from 192.168.0.9: icmp_seq=179 ttl=64 time=0.358 ms
64 bytes from 192.168.0.9: icmp_seq=180 ttl=64 time=5.73 ms
^C
--- 192.168.0.9 ping statistics ---
180 packets transmitted, 178 received, 1% packet loss, time 179678ms
```

Ataque com ID 6: nesse ataque o sistema sofreu uma queda brusca de desempenho. Observe na Figura 31 que o sistema negou serviço a 98% das solicitações e que o tempo de resposta estava muito alto.

Figura 33 – Resposta do Fedora 09 ao ataque ID 06

```
[root@no03 tfn2k]# ping 192.168.0.35
PING 192.168.0.35 (192.168.0.35) 56(84) bytes of data.
64 bytes from 192.168.0.35: icmp_seq=59 ttl=64 time=1134 ms
64 bytes from 192.168.0.35: icmp_seq=71 ttl=64 time=82.3 ms
64 bytes from 192.168.0.35: icmp_seq=127 ttl=64 time=91.3 ms
^C
--- 192.168.0.35 ping statistics ---
180 packets transmitted, 3 received, 98% packet loss, time 179559ms
```

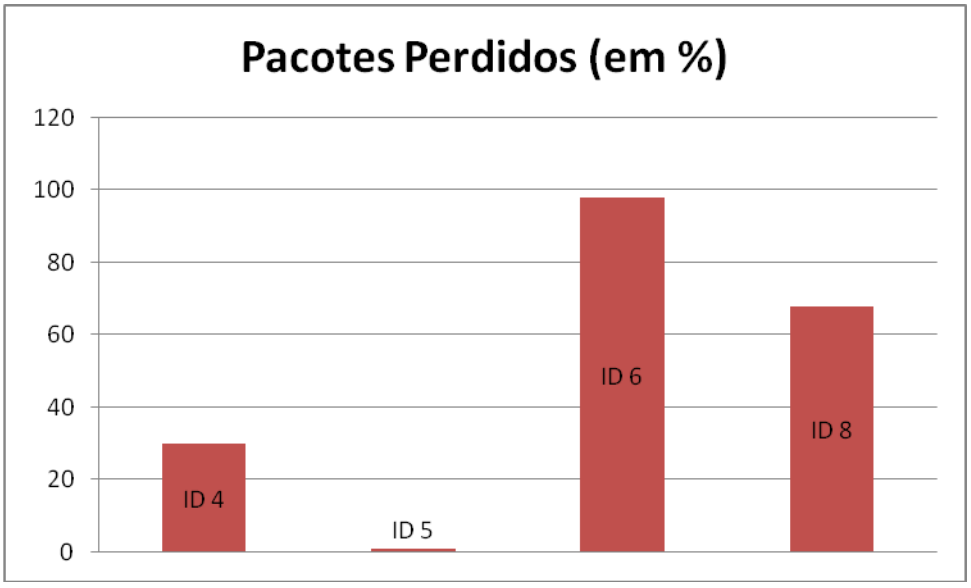
Ataque com ID 8: nesse ataque o sistema sofreu uma queda considerável. Mesmo conseguindo responder algumas solicitações, apesar do tempo de resposta, o sistema com o decorrer do ataque passou a ter seu desempenho afetado e passou a negar serviço. No final, como pode ser observado na Figura 32 cerca de 68% dos pacotes foram perdidos.

Figura 34 – Resposta do Fedora 09 ao ataque ID 08

```
64 bytes from 192.168.0.9: icmp_seq=136 ttl=64 time=2040 ms
64 bytes from 192.168.0.9: icmp_seq=139 ttl=64 time=19.8 ms
64 bytes from 192.168.0.9: icmp_seq=143 ttl=64 time=19.3 ms
64 bytes from 192.168.0.9: icmp_seq=145 ttl=64 time=19.1 ms
64 bytes from 192.168.0.9: icmp_seq=146 ttl=64 time=19.7 ms
64 bytes from 192.168.0.9: icmp_seq=151 ttl=64 time=19.6 ms
64 bytes from 192.168.0.9: icmp_seq=152 ttl=64 time=20.5 ms
64 bytes from 192.168.0.9: icmp_seq=154 ttl=64 time=20.7 ms
64 bytes from 192.168.0.9: icmp_seq=157 ttl=64 time=19.4 ms
64 bytes from 192.168.0.9: icmp_seq=158 ttl=64 time=20.3 ms
64 bytes from 192.168.0.9: icmp_seq=164 ttl=64 time=20.0 ms
64 bytes from 192.168.0.9: icmp_seq=165 ttl=64 time=19.7 ms
64 bytes from 192.168.0.9: icmp_seq=166 ttl=64 time=19.8 ms
64 bytes from 192.168.0.9: icmp_seq=167 ttl=64 time=21.7 ms
64 bytes from 192.168.0.9: icmp_seq=169 ttl=64 time=19.9 ms
64 bytes from 192.168.0.9: icmp_seq=171 ttl=64 time=20.2 ms
64 bytes from 192.168.0.9: icmp_seq=172 ttl=64 time=18.1 ms
^C
--- 192.168.0.9 ping statistics ---
181 packets transmitted, 57 received, 68% packet loss, time 180426ms
```

O Gráfico 9 apresenta o desempenho o sistema em relação a cada ataque. Como pode ser visto o ataque com ID 6 foi o que obteve maior sucesso, levando o sistema a uma queda brusca de desempenho rapidamente e posteriormente a uma negação de serviço. Por outro lado o ataque com ID 5 foi o que obteve o menor sucesso, não afetando em nada a utilização dos serviços pelos usuários.

Gráfico 9 – Tipo de Ataque x Pacotes Perdidos Fedora 09



4.3.7 – Análise dos testes efetuados

Com base nos resultados obtidos pode-se observar que os Sistemas Operacionais para servidores com suas configurações *default* são vulveráveis. Apesar de em alguns casos, como o observado no Fedora, a versão mais atual consegue tratar alguns tipos de ataques com melhor eficiência, não se pode confiar plenamente nas configurações *default* destes Sistemas Operacionais.

O tipo de ataque que não afetou em muito o desempenho dos sistemas foi o SYN flood, o que pode demonstrar que os SOs junto com o Firewall consegue tratar as requisições SYN sem afetar o desempenho do sistema.

Por outro lado os ataques do tipo UDP flood, ICMP echo reply e o MIX (UDP, SYN e ICMP) conseguiram afetar o desempenho do sistema, sendo que o ataque do tipo ICMP foi o que apresentou maior eficiência, fazendo os sistemas negarem serviços a maioria das requisições.

Sendo assim observa-se que é recomendável a utilização de ferramentas e software de segurança específicos para minimizar essas vulnerabilidade e consequentemente impedir esses tipos de ataques. Tornando dessa maneira o sistema mais confiável e sempre disponível.

CONSIDERAÇÕES FINAIS

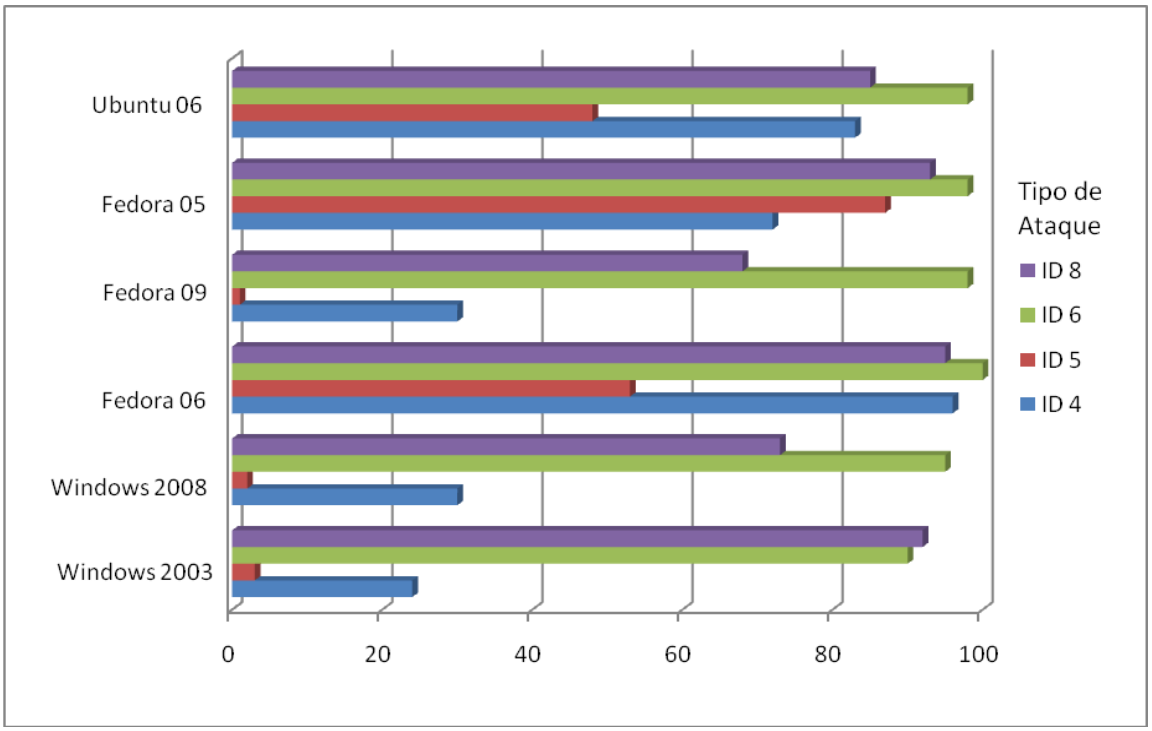
Estudos e pesquisas acerca de testes de vulnerabilidades em sistemas operacionais para servidores são complexos e não se esgotam com esta investigação bibliográfica e empírica.

Desde o início o presente trabalho teve como objetivo verificar as vulnerabilidades existentes nos Sistemas Operacionais para Servidores e em momento algum se pensou na pretensão de avaliar e/ou comparar qual desses SOs é o melhor. Estas questões podem ser tema de um trabalho diferenciado e com a utilização de outra metodologia.

Sendo assim, com base nos resultados obtidos pôde-se confirmar a premissa inicial do presente trabalho, constatar que os Sistemas Operacionais testados são vulneráveis e que, apesar dos ataques efetuados serem bastante conhecidos, as configurações *default* desses Sistemas Operacionais não os previne.

Desta forma, baseado nos resultados obtidos neste trabalho observou-se a importância de uma configuração correta dos sistemas operacionais e da utilização de ferramentas específicas (Firewall, IDS, antivírus, etc.) buscando aumentar a segurança e confiabilidade dos sistemas computacionais. No Gráfico 10 é possível verificar as diferenças entre as vulnerabilidades e desta maneira se prevenir de acordo o sistema operacional utilizado.

Gráfico 10 – Resultados obtidos



Comprovou-se a facilidade em se conseguir acesso às ferramentas de ataques DDoS, pois durante a pesquisa foi possível a constatação de que estas ferramentas são amplamente disponibilizadas pela Internet e que após consegui-las, os processos de instalações e utilizações são extremamente simples.

Esta constatação demonstra e pleiteia a importância em investir na prevenção aos ataques, pois um usuário mesmo sem muito conhecimento específico pode fazer uso dessas ferramentas, causar ataques eficientes de acordo com seus objetivos e provocar danos desastrosos às possíveis vítimas.

Propostas de trabalhos futuros

No decorrer da realização deste trabalho surgiram novas idéias a respeito do tema estudado, e que por não se enquadrarem no objetivo inicial nem no cronograma predeterminado, não foi possível implementá-las. Assim, tais idéias apresentam-se como propostas para possíveis trabalhos futuros.

Entre essas idéias surgidas durante o desenvolvimento do trabalho duas delas merecem ser sugeridas:

- Uma proposta de configuração segura para servidores utilizando ferramentas específicas e verificando sua eficiência.
- Verificação do desempenho de uma máquina física quando a máquina virtual sofre o ataque e vice-versa.

Em síntese, estudos e pesquisas envolvendo esta área tão abrangente dentro da segurança computacional podem suscitar novas idéias que possivelmente culminarão em trabalhos significativos para o meio acadêmico.

REFERÊNCIAS

ALBUQUERQUE, Ricardo; RIBEIRO, Bruno. **Segurança no desenvolvimento de software: como garantir a segurança do sistema para seu cliente usando a ISSO/IEC.** Rio de Janeiro: Campus, 2002.

ASSUNÇÃO, Marcos F. A. **Guia do Hacker Brasileiro.** Florianópolis:Visual Books, 2002

CANEDO, Daniel Rosa. **Um Ambiente Experimental para Análise de Ataques de Negação de Serviço.** 85f. Dissertação (Mestrado em Engenharia Elétrica). Universidade De Brasília, Faculdade de Tecnologia. Departamento de Engenharia Elétrica. Distrito Federal, 2006.

CARUSO, Carlos A. A.; STEFFEN, Flávio D. **Segurança em informática e de informação.** 2. ed. São Paulo: SENAC, 1999.

CERT, 2006. **Cartilha de segurança para Internet,versão 3.1**. Disponível em: http://cartilha.cert.br/. Acesso em: 03 de maio de 2009.

CERT-07,1999. **CERT® Incident Note IN-99-07.** Disponível em: http://www.cert.org/incident_notes/IN-99-07.html. Acesso em: 12 de agosto de 2009.

CERT-17, 1999, **CERT® Advisory CA-1999-17 Denial-of-Service Tools**. Disponível em: http://www.cert.org/advisories/CA-1999-17.html. Acesso em: 12 de agosto de 2009.

CHESWICK, Willian R.; BELOVIN Steven M.; RUBIN Aviel D.. **Firewalls e segurança na Internet**: repelindo o hacker ardiloso. Trad. Edson Furmankiewicz – 2. ed. – Porto Alegre: Bookman, 2005.

ÉPOCA, **Hacker regenerado (2003)**. Entrevista de Kevin Mitnick à revista Época. Disponível em: http://revistaepoca.globo.com/Epoca/0,6993,EPT600936-1666,00.html. Acesso em: 20 de julho de 2009.

GOMES, Carlos F. S.; RIBEIRO, Priscilla C. C. **Gestão da cadeia de suprimentos integrada à tecnologia da informação.** São Paulo: Pioneira Thomson Learning, 2004.

GUIMARÃES, Alexandre G.; LINS, Rafael D.; Oliveira, Raimundo. **Segurança com VPNs.** Rio de Janeiro: Brasport, 2006.

HEADY, R.; LUGER, G.; MACCABE, A., SERVILA, M.. **The Architecture of a Network Level Intrusion Detection System**, Technical Report - Department of Computer Science, University of New Mexico, USA , 1990.

JUNIOR, Jorge L. P. **Protótipo de Software para a monitoraçãode pacotes em uma rede TCP/IP em ambiente Linux** 48f. Dissertação (Trabalho de Conclusão de Curso). Universidade Regional de Blumenau, Blumenau,2002.

KONRATH, Marlon A. **Estudo das Vulnerabilidades da Arquitetura TCP/IP e Desenvolvimento de uma Ferramenta para Detecção de Intrusão** 55f. Monografia

(Trabalho de Conclusão de Curso). Universidade do Vale do Rio dos Sinos – Centro de Ciências Exatas e Tecnológicas – Curso de Informática, São Leopoldo, novembro de 2001.

KUROSE, J. F.; ROSS, K. W. **Redes de computadores e a Internet:** uma abordagem top-down. 3. ed. São Paulo: Pearson-Addison Wesley, 2005.

MATTOS, Diogo M. F. **Virtualização:** VMWare e XEN. Disponível em: http://www.gta.ufrj.br/grad/08_1/virtual/artigo.pdf. Acesso em: 12 de maio de 2009.

McCLURE, Stuart; SCAMBRAY, Joel; KURTZ, George. **Hackers exposto: segredos e soluções para a segurança de redes.** Tradução de Daniel Vieira. Rio de Janeiro: Elsevier, 2003.

MICROSOFT: **Proteja-se contra o worm de computador Conficker**. Disponível em: http://www.microsoft.com/brasil/protect/computer/viruses/worms/conficker.mspx. Acesso em: 10 de julho de 2009.

MOURA, J. A. B.; et al. **Redes de computadores:** serviços, administração e segurança. São Paulo: Makron Books, 1999.

NORTHCUTT, Stephe et. al. **Desvendando segurança em redes:** o guia definitivo para fortificação de perímetro de rede usando Firewall, VPNs, roteadores e sistema de detecção de invasores. Tradução de Daniel Vieira. Rio de Janeiro: Campus, 2002.

INSECURE.ORG. NMAP. **Ferramenta de Port Scaning.** Disponível em: http://www.insecure.org/nmap/. Acesso em: 15 de julho de 2009.

SANS. ***Twenty Most Critical Internet Security Vulnerabilities***. Disponível em: www.sans.org. Acesso em: 21 de agosto de 2009.

SCRIMGER, Rob... [et al.]; **TCP/IP: a bíblia.** Tradução de Edson Furmankievics, DocWare Traduções Técnicas. Rio de Janeiro: Elsevier, 2002.

SPANCESKI, Francini Reitz. **Política De Segurança Da Informação - Desenvolvimento De Um Modelo Voltado Para Instituições De Ensino.** Monografia (Trabalho de Conclusão de Curso) – Instituto Superior Tupy, Joinville, 2004.

STAFF, 1999. **The "stacheldraht" distributed denial of service attack tool**. Disponível em: http://staff.washington.edu/dittrich/misc/stacheldraht.analysis. Acesso em: 20 de julho de 2009.

TANENBAUM, Andrew S. **Redes de Computadores**. 7. ed. Tradução: Vanderberg D. Souza. Rio de Janeiro: Elsevier, 2003.

TERRA, 2009. **Vírus Conficker já infecta mais de 15 milhões de PC.** Disponivel em: **s**http://tecnologia.terra.com.br/interna/0,,OI3473025-EI4805,00-Virus+Conficker+ja+infecta+mais+de+milhoes+de+PCs.html. Acesso em: 10 de julho de 2009.

TITTEL, Ed. **Redes de Computadores.** 3. ed. Porto Alegre: Ed.BookMan, 2003.

TUFANO, Douglas. Guia Prático da Nova Ortografia.(2008). Disponível em: http://www.livrariamelhoramentos.com.br/Guia_Reforma_Ortografica_Melhoramentos.pdf. Acesso em: 05 de maio de 2009.

ULBRICH, Henrique C.; VALLE, James D.**Universidade Hacker.** São Paulo: Digerati Books, 2006.