CENTRO PAULA SOUZA
GOVERNO DO ESTADO DE SÃO PAULO

Faculdade de Tecnologia de Americana
Curso de Análise de Sistemas e Tecnologia da Informação

ABORDAGEM DE SISTEMAS DE DETECÇÃO DE INTRUSÃO UTILIZANDO A FERRAMENTA SNORT

Renato Tetsuo Yoshizawa

Monografia apresentada á Faculdade de Tecnologia de Americana, para graduação no Curso Superior de Tecnologia em Análise de Sistemas e Tecnologia da Informação.

Americana – SP
2010

CENTRO PAULA SOUZA
GOVERNO DO ESTADO DE SÃO PAULO

Faculdade de Tecnologia de Americana
Curso de Análise de Sistemas e Tecnologia da Informação

ABORDAGEM DE SISTEMAS DE DETECÇÃO DE INTRUSÃO UTILIZANDO A FERRAMENTA SNORT

Renato Tetsuo Yoshizawa

renatorens@gmail.com

Monografia apresentada á Faculdade de Tecnologia de Americana, para graduação no Curso Superior de Tecnologia em Análise de Sistemas e Tecnologia da Informação.

Área de Concentração: Segurança da informação

Orientador: José Luis Zem

Americana – SP
2010

CENTRO PAULA SOUZA
GOVERNO DO ESTADO DE SÃO PAULO

Faculdade de Tecnologia de Americana
Curso de Análise de Sistemas e Tecnologia da Informação

ABORDAGEM DE SISTEMAS DE DETECÇÃO DE INTRUSÃO UTILIZANDO A FERRAMENTA SNORT

Renato Tetsuo Yoshizawa

Monografia aprovada em 08/12/2010 para obtenção do título de Tecnólogo em Análise de Sistemas e Tecnologia da Informação

Banca Examinadora

Professor Orientador: José Luis Zem

Professor Co-Orientador: Lincon Peretto

Professor Convidado: Rogério Nunes de Freitas

DEDICATÓRIA

A toda a minha família e colegas que sempre estiveram e estarão comigo.

Principalmente aos meus pais: Luiz e Maria.

Que sempre apoiaram, ajudaram

e me incentivaram

nos momentos em

que realmente

precisei.

AGRADECIMENTOS

A todos que auxiliaram, acreditaram, incentivaram e me guiaram ao longo deste trabalho.

Aos meus pais e minha família que sempre estiveram comigo.

Aos professores que passaram seu conhecimento ao longo da graduação, principalmente aos professores Lincon e Rogério e ao meu orientador José Luis Zem.

A Faculdade de Tecnologia de Americana e toda a sua equipe que possibilitou a minha graduação no Ensino Superior público de qualidade e gratuito.

Aos meus colegas, colegas de classe, colegas de trabalho que demonstraram companheirismo, incentivo e proporcionaram momentos de muita alegria: Agnaldo, Alexandre, André, Arley, Bruno, Caio, Caroline Calixto, Danilo, David, Diogo, Elvis, Genérico, Guilherme, Gustavo, Henrique, Lucas, Mariana, Robson, Rodrigo e Rone.

EPÍGRAFE

“A persistência é o caminho do êxito”
Charles Chaplin

YOSHIZAWA, R. T. **Abordagem de Sistemas de Detecção de Intrusão utilizando a ferramenta Snort**. 2010. Americana. 51p. Monografia, Faculdade de Tecnologia de Americana

Resumo

Com o grande valor que a informação possui atualmente, a segurança em torno dela necessita de certos cuidados adicionais, além do convencional *firewall* e *softwares* antivírus. Os Sistemas de Detecção de Intrusão (IDS) vão além destas ferramentas de segurança, identificando tentativas de ataques e facilitando a análise do tráfego malicioso na rede ou em um *host* específico, além da possibilidade de bloqueio em tempo real. Este trabalho tem como principal objetivo verificar o comportamento de um Sistema de Detecção de Intrusão durante simulações de tentativas de ataques. A ferramenta implantada é o IDS Snort, por ser uma ferramenta *Open Source*, além da ampla aceitação e utilização em ambientes corporativos. Serão abordados ao longo do trabalho informações sobre os Sistemas de Detecção de Intrusão, ataques, estatísticas dos ataques no Brasil, *firewall* e alguns IDS disponíveis no mercado. Um cenário será elaborado para a realização dos testes, possuindo um *host* com o Sistema de Detecção de Intrusão implantado, recebendo diferentes tipos de ataques, para observar seu desempenho e verificar os resultados obtidos.

Palavras-Chave: Sistema de Detecção de Intrusão, Ataques, Segurança

YOSHIZAWA, R. T. **Approach of Intrusion Detection Systems using the Snort tool**. 2010. Americana. 51p. Monograph, Technology's university of Americana

Abstract

With the value that information has currently, the security around information requires some additional care, beyond the conventional firewall and antivirus software. The Intrusion Detection Systems (IDS) go beyond these security tools, identifying attack attempts and facilitating the analysis of malicious traffic on the network or on a particular host, adding the ability to block in real time. This work has as main objective to verify the behavior of an Intrusion Detection System during simulations of attempted attacks. The tool deployed is Snort, being an Open Source tool, in addition to broad acceptance and use in corporate environments. Will be addressed throughout the paper information of Intrusion Detection Systems, attacks, attacks statistics in Brazil, firewall and some IDS available in the market. A scenario will be prepared for the tests, having a host with the Intrusion Detection System deployed, receiving different types of attacks, to observe the performance and check the results.

Keywords: Intrusion Detection System, Attack, Security.

LISTA DE ILUSTRAÇÕES

LISTA DE ABREVIATURAS E SIGLAS

AIX *Advanced Interactive executive*

BSD *Berkeley Software Distribution*

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

DDoS *Distributed Denial of Service*

DIDS *Distributed Intrusion Detection System*

DNS *Domain Name System*

DoS *Denial of Service*

EMS *Enterprise Management Server*

FTP *File Transfer Protocol*

Gbps *Gigabytes* por segundo

HIDS *Host-based Intrusion Detection System*

HTML *HyperText Markup Language*

HTTP *Hypertext Transfer Protocol*

HP-UX *Hewlett Packard UNIX*

IDS *Intrusion Detection System*

IP *Internet Protocol*

IPS *Intrusion Prevention System*

IRIX *Silicon Graphics UNIX-like Operating System*

LAN *Local Area Network*

MAC *Media Access Control*

MAC OS *Macintosh Operation System*

Mbps *Megabytes* por segundo

MSDNAA *Microsoft Developer Network Academic Alliance*

NIDS *Network-based Intrusion Detection System*

OISF	*Open Information Security Foundation*
PDF	*Portable Document Format*
SSH	*Secure Shell*
SunOS	*Sun Operating System*
SYN	*Synchronize*
TCP	*Transmission Control Protocol*
TI	Tecnologia da Informação
UDP	*User Datagram Protocol*
UNIX	*Uniplexed Information and Computing System*
VLAN	*Virtual Local Area Network*

SUMÁRIO

INTRODUÇÃO

A informação, um bem muito valioso, necessita de grandes cuidados na questão de segurança, pois o conteúdo destas informações que trafegam na rede é de extrema importância, como dados confidenciais e transações bancárias. Com o crescimento do uso de recursos tecnológicos e a *Internet*, os incidentes em Tecnologia da Informação (TI) acompanharam este ritmo, o que é algo preocupante, pois a TI em ambientes corporativos é cada vez mais significativa e falhas de segurança nestas redes, que podem ser ocasionados por descuido de usuários, administradores de redes ou até mesmo no código dos sistemas, são utilizados em ataques, que de alguma forma causam danos e perdas, como a obtenção, alteração e destruição de informações confidenciais, interrupção de serviços ou simplesmente fazer propaganda, entre outros diversos motivos.

Visto a grande importância da informação, são necessários certos cuidados, como configurar corretamente roteadores e *switches*, manter o sistema operacional atualizado e principalmente, ter um *firewall* com regras bem definidas. Um Sistema de Detecção de Intrusão (*IDS*) vai além destes fatores, caso exista alguma falha na rede e uma invasão seja iniciada, tanto externa quanto internamente, este sistema irá detectar e alertar o administrador da rede ou até mesmo bloquear a invasão.

Os principais objetivos deste trabalho são observar o comportamento de uma ferramenta de Detecção de Intrusão durante tentativas de ataques e verificar o seu desempenho na detecção, analisando os registros e outras informações disponibilizadas pelo sistema após o ataque ser detectado.

Os Sistemas de Detecção de Intrusão serão abordados com o intuito de ampliar o conhecimento de ferramentas para segurança da informação, descrevendo os sistemas *IDS*, apresentando os diferentes tipos e seus métodos de detecção. Sendo necessário descrever alguns tipos de ataques conhecidos, para efetuá-los em uma rede com um sistema *IDS* e que o mesmo detecte estas tentativas. Para tornar possível os testes, foi elaborado um cenário com um sistema *IDS* implantado que receberá ataques, possibilitando verificar os

resultados obtidos pela ferramenta, que deverá detectar, gerar registros e alertas destas tentativas de invasão. Serão apresentadas algumas ferramentas *IDS* disponíveis no mercado, mas foi adotada, para fins de testes, a ferramenta *Snort*, por ser uma solução *Open source*, além de ser considerado um dos *IDS* mais utilizados.

Foram realizadas várias pesquisas sobre Sistemas de Detecção de Intrusão, *IDS* Snort, *firewall*, ataques e segurança da informação. Todo este levantamento teórico foi obtido através da leitura de livros e pesquisas na Internet, com o auxilio de professores e do orientador.

1 LEVANTAMENTO TEÓRICO

Serão abordados algumas ferramentas, estatísticas e conceitos relacionados á segurança de um *host* específico ou até abranger toda uma rede de computadores, servindo de base para o desenvolvimento e os testes que serão realizados ao longo deste trabalho.

1.1 Intrusão

A intrusão é basicamente uma atividade não autorizada em redes ou computadores, e SOUZA (2002) define que "*A intrusão ou ataque podem ser definidos como qualquer conjunto de ações que tentam comprometer a integridade, confidencialidade ou disponibilidade de um recurso computacional, independente do sucesso ou não destas ações*". A integridade refere-se a alterações não-autorizadas, a confidencialidade ao acesso não-autorizado e a disponibilidade à acessibilidade de recursos e/ou informações.

O objetivo principal de uma intrusão, geralmente é para a visualização, modificação e eliminação de informações ou ainda para a interrupção de recursos, como servidores (*Web* e banco de dados, por exemplo), explorando-se as vulnerabilidades dos sistemas. As principais causas para vulnerabilidades em redes e computadores estão vinculadas a falhas no processo de construção de programas (*bugs*), administração da rede de forma inadequada e usuários descuidados ou sem treinamento (RUBIN e CHESWICK, 2001).

1.2 Estatísticas

O CERT.br é o Grupo de Resposta a Incidentes de Segurança na Internet brasileira e atua como uma central de notificações, coordenando e apoiando no processo de respostas a incidentes e quando se fizer necessário, coloca as partes envolvidas em contato, em suma, consiste em um elemento intermediador.

Um incidente de segurança, segundo o grupo CERT.br consiste em "*qualquer evento adverso, confirmado ou sob suspeita, relacionado à segurança de sistemas de computação*

ou de redes de computadores". O grupo ainda exemplifica estes incidentes, que são "*tentativas de ganhar acesso não autorizado a sistemas ou dados; ataques de negação de serviço; uso ou acesso não autorizado a um sistema; modificações em um sistema, sem o conhecimento, instruções ou consentimento prévio do dono do sistema; desrespeito à política de segurança ou à política de uso aceitável de uma empresa ou provedor de acesso*."

Este grupo coleta estatísticas sobre incidentes relacionados à segurança no Brasil, como o gráfico representado pela Figura 1, que identifica o total de incidentes, ressaltando somente os incidentes reportados espontaneamente ao referido grupo.

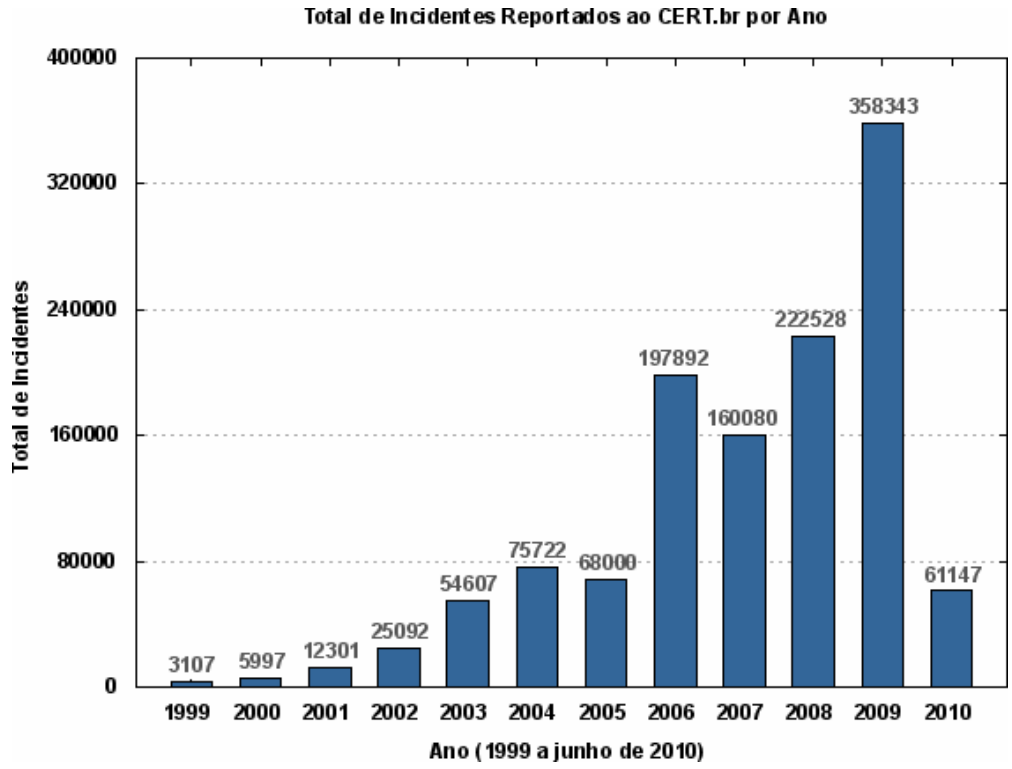

Figura 1 – Total de incidentes reportados ao CERT.br

(de 1999 a junho de 2010).

Como é possível observar na Figura 1, os incidentes cresceram muito desde o ano de 1999, que é algo alarmante, pois a tecnologia da informação nos ambientes corporativos está sendo usada como um diferencial estratégico, para maior produção e melhor controle,

em menor tempo e com menor custo, além do oferecimento de serviços *online* (desde instituições financeiras até supermercados) que movimentam parte da economia do país. O gráfico da Figura 2, obtido através do *site* CERT.br, apresenta os incidentes reportados, subdivididos por tipos de ataque.

Figura 2 - Incidentes reportados ao CERT.br (Tipos de ataques).

1.3 Tipos de ataque

Os sistemas estão sujeitos a falhas, principalmente se possuírem grande número de linhas de código, como o sistema operacional, mas também afeta os *softwares* aplicativos, como navegadores *Web* ou leitores de *PDF* e na opinião de MORIMOTO (2006) "*As brechas de segurança são como balas perdidas, ninguém pode dizer onde surgirá a próxima. Mesmo um sistema com um excelente histórico de segurança pode revelar um bug monstruoso a qualquer momento*".

Existem vários tipos de ataques que utilizam estas falhas para invadir ou afetar de alguma forma um sistema. A seguir serão apresentados alguns destes tipos de ataque.

1.3.1 *Exploits*

Consiste em um termo genérico para se referir a um código elaborado para explorar vulnerabilidades específicas, podendo ser usado diretamente ou incorporados a um vírus, ferramentas de detecção de vulnerabilidades e outros programas.

A ferramenta *Nessus*, um conhecido *scanner* que detecta vulnerabilidades, possui um conjunto de *exploits* para as brechas conhecidas, verificando se o serviço ou programa vulnerável está ativo, através de simulações de ataques com o *exploit* correspondente.

1.3.2 *DoS (Denial of Service)*

O objetivo do ataque de negação de serviços (*DoS*) é interromper um sistema e deixá-lo indisponível e não uma intrusão. Este ataque pode ser efetuado contra *hosts* conectados a *Internet.*

São utilizadas técnicas que sobrecarregam uma rede ou *host* de forma que os verdadeiros usuários não consigam acessá-la, realizando inúmeras requisições até que o sistema torne-se inacessível.

O *DoS* é efetuados a partir de algumas características do protocolo *TCP/IP*, o tipo mais conhecido é através do *SYN Flooding*, no qual é estabelecida uma conexão *TCP* com um servidor utilizando o sinal *SYN*. Sendo estabelecidas várias conexões, o servidor ficará sobrecarregado e não conseguirá atender todas as conexões solicitadas e então, recusa novos pedidos.

Um ataque mais eficaz é a negação de serviços distribuído (*DDoS*) que utiliza vários computadores para atacar um alvo. Para obter vários computadores que contribuirão com o ataque, normalmente são inseridos programas de ataque *DDoS* em vírus, que se espalham e contaminam milhares de computadores, tornando-se zumbis, que são controlados por máquinas mestres, e estes recebem ordem do atacante.

Figura 3 - Hierarquia de computadores em um ataque *DDoS*

(ALECRIM, 2004)

1.3.3 Engenharia Social

Tipo de ataque que não utiliza falhas de segurança em sistemas, mas que tenta convencer o usuário a ceder informações confidenciais e Alecrim afirma que *"A engenharia social é um dos meios mais utilizados de obtenção de informações sigilosas e importantes. Isso porque explora com muita sofisticação as 'falhas de segurança dos humanos'. As empresas investem fortunas em tecnologias de segurança de informações e protegem fisicamente seus sistemas, mas a maioria não possui métodos que protegem seus funcionários das armadilhas de engenharia social."*

O principal objetivo é obter informações, como o número e senha de uma conta-corrente. Como os sistemas dos bancos são bem protegidos, o criminoso faz uma página falsa do banco, praticamente idêntica, com endereço semelhante e envia para uma lista com milhares de *e-mails*, geralmente com uma mensagem de premiação ou atualização de dados, juntamente com o *link* da página falsa e quando o usuário digita seus dados, eles são enviados para o criminoso.

1.3.4 *Backdoors*

Um atacante, para garantir o retorno ao computador já invadido sem a necessidade de utilizar novamente os métodos de invasão, abre uma "porta dos fundos", podendo retornar com maior facilidade ou até mesmo que a falha inicial já esteja corrigida.

Normalmente um *backdoor* é inserido por um invasor, mas também pode ser inserido através de cavalos de tróia ou falhas na configuração de *softwares* de administração remota. O *backdoor* afeta tanto sistemas operacionais *Windows*, como os *Unix* e também o *Mac OS*, entre outros.

1.3.5 *Worms*

O *worm* é um programa que se propaga automaticamente através de redes e utiliza vulnerabilidades ou falhas na configuração de *softwares*, podendo se propagar sem mesmo a necessidade de ser executado explicitamente.

Diferentemente dos vírus, o *worm* não causa danos como infecção de programas e arquivos ou destruição de informações, eles consomem recursos do computador, podendo causar efeitos notáveis em sistemas ou em redes de computadores, pois se propagam rapidamente, gerando grande quantidade de cópias.

1.3.6 *Site Defacement*

São ataques realizados em páginas *Web*, que geralmente modificam seu conteúdo, com o objetivo de disseminar uma mensagem ou apenas como uma realização pessoal, por ter a capacidade de quebrar a segurança de um sistema.

Este ataque explora falhas nos serviços do sistema operacional (*SSH* ou *FTP*, por exemplo) no qual é feito *upload* de um documento *HTML* criado por ele, alterando a página original e finalmente, todas as pessoas que acessarem o site invadido, visualizarão a página modificada.

1.3.7 Forca Bruta

O processo de força bruta é utilizado para quebrar senhas de usuários ou programas, utilizados por invasores para descobrir senhas e obter acesso ao sistema. A força bruta utiliza um dicionário de senhas possíveis ou tenta combinações de letras, números e símbolos.

Este processo é muito lento e a eficiência esta ligada ao grau de dificuldade da senha dos usuários. São consideradas senhas fortes as senhas que possuem letras minúsculas e maiúsculas, números e símbolos.

1.3.8 *Port scanning*

O *Port scanning* ou varredura de portas é o processo de se conectar ás portas *TCP* e *UDP* de um sistema, para verificar quais portas estão ativas, sendo possível também verificar o sistema operacional e aplicações em uso, entre outras informações.

A varredura de portas pode ser utilizada para realizar auditoria em um sistema, verificando os serviços que estão ativos sem necessidade e eliminar estes serviços, diminuindo as vulnerabilidades que possam existir no sistema. Mas também é utilizada para efetuar invasões, sendo essencial para determinar as portas ativas e as possíveis vulnerabilidades dos serviços. Uma forma de detectar estas varreduras é implantar um Sistema de Detecção de Intrusão, como o *Open source Snort.*

1.4 *Firewall*

É uma barreira entre a rede interna e a rede externa, que analisa o tráfego em tempo real, permitindo ou bloqueando de acordo com as regras definidas. MORIMOTO (2006) faz uma analogia *“Imagine o firewall como a muralha que cercava muitas cidades na idade média. Mesmo que as casas não sejam muito seguras, uma muralha forte em torno da cidade garante a segurança. Se ninguém consegue passar pela muralha, não é possível chegar até as casas vulneráveis. Se, por acaso, as casas já são seguras, então a muralha*

aumenta ainda mais a segurança.”. O *firewall* pode ser um software ou uma combinação de *software* com *hardware*, centralizando a configuração e administração do tráfego, sem a necessidade da instalação de *softwares* nos *hosts*. Conforme MORIMOTO (2006) afirma “*é sempre melhor prevenir do que remediar, e a melhor forma de se proteger contra brechas é manter um firewall ativo, permitindo apenas acesso aos serviços que você realmente deseja disponibilizar. Reduzindo os pontos vulneráveis, fica mais fácil cuidar da atualização dos serviços expostos e, assim, manter seu servidor seguro”*.

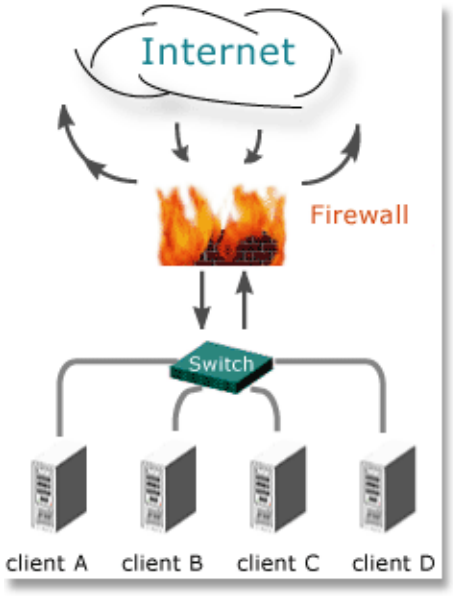

Figura 4 - *Firewall* em uma rede

(MORIMOTO, 2006)

1.4.1 Funções do *firewall*

O *firewall* é uma das principais ferramentas para garantir segurança de uma rede através da filtragem de pacotes entre a rede interna e a externa. As principais funções do f*irewall* são filtrar ou bloquear serviços, controlar acesso à *hosts* ou serviços e registrar o tráfego de uma rede.

1.4.1.1 Filtrar e bloquear serviços

Serviços que não são considerados seguros (*TELNET*, por exemplo) ou que forneçam informações utilizadas em intrusões, podem ser bloqueados no *firewall*, aumentando a segurança da rede interna. Ele ainda pode rejeitar pacotes de determinada origem.

1.4.1.2 Controle de acesso a *hosts*

O firewall permite tornar os *hosts* da rede interna inacessíveis e aplicar regras de acesso externo para determinados *hosts* ou serviços (como servidores *Web*) de uma rede interna, possibilitando assim, o controle de acesso.

1.4.1.3 Registro do tráfego

Através dos *logs* do *firewall* é possível obter informações dos acessos, permitindo identificar possíveis tentativas de ataques á rede.

1.4.2 Firewalls e os Sistemas de Detecção de Intrusão

Em um Sistema de Detecção de Intrusão é possível adicionar *plugins* de saída, como o *Guardian* e o *Snortsam* (específicos para o *IDS Snort*), que trabalham em conjunto com os sistemas *IDS,* fazendo atualizações automáticas das regras do firewall baseadas nos alertas gerados pelo sistema. Estes *plugins* trabalham com os *firewalls* mais conhecidos (como o *Checkpoint Firewall* e o *iptables*, por exemplo), bloqueando as origens dos ataques em tempo real. Esta combinação se assemelha com uma das funções de um IPS (Intrusion Prevention System), ou seja, além da detecção, o sistema proporciona prevenção á ataques, efetuando o bloqueio automático do tráfego malicioso.

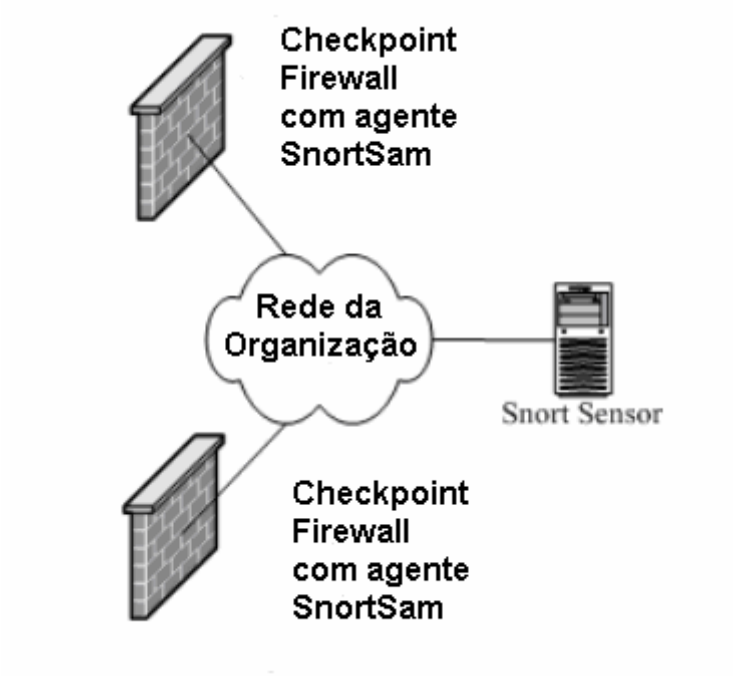

Figura 5 - *Snotsam* trabalhando em conjunto com o *Checkpoint firewall*

(REHMAN, 2003)

1.5 Conceitos e definições

Para melhor entendimento dos sistemas *IDS,* será feita uma breve apresentação de alguns termos que serão utilizados neste trabalho e terão grande importância para o entendimento. Tais termos foram propostos e definidos por REHMAN (2003).

1.5.1 Assinaturas

É um padrão de conteúdo de um pacote. Ela é utilizada para detectar tipos de ataques, que pode ser localizada em diferentes partes dos pacotes, dependendo do tipo ataque.

1.5.2 Alertas

Os alertas são avisos de intrusão, que podem estar no formato de janelas, mensagens no console ou por *e-mail*. Estes alertas são armazenados nos *logs* ou em banco de dados, para que possam ser examinados posteriormente.

1.5.3 *Logs*

Logs são registros de eventos relevantes, para que os administradores possam observar o comportamento dos sistemas computacionais. Nas ferramentas *IDS* tem o objetivo de descrever uma intrusão ou outras atividades que possam causar algum dano.

1.5.4 Alarmes falsos

São alarmes gerados pelo sistema, porém, não se trata de atividades de intrusão. Estes alarmes falsos são causados por um tráfego que não seja de atividades maliciosas, mas que por algum motivo (algum *host* mal configurado, por exemplo) podem ter comportamento típico de um ataque.

1.5.5 Sensor

É um computador ou outro dispositivo localizado em uma rede no qual esteja implantado um Sistema *de* Detecção de Intrusão.

1.6 Sistemas de Detecção de Intrusão *(IDS – Intrusion Detection System)*

Um Sistema de Detecção de Intrusão (*IDS*) é todo *software*, *hardware* ou uma combinação de ambos com o objetivo de detectar atividades de intrusos e quando atividades suspeitas são detectadas, são gerados registros e alertas possibilitando o administrador da rede identificar o tráfego malicioso e tomar as providencias necessárias.

Os sistemas *IDS* são implantados em locais estratégicos, variando de acordo com o seu tipo, podendo ser aplicado em hosts específicos, em segmentos de rede ou até uma combinação de vários sensores em *hosts* e redes com um gerenciamento centralizado.

Há dois métodos que um sistema *IDS* pode utilizar para detectar atividades maliciosas, a baseada em assinaturas, que pode ser comparada com as ferramentas de antivírus, e a baseada em anomalias, que verifica atividades padrões e quando há um desvio, o alerta é gerado.

1.6.1 Tipos de *IDS*

Existem diferentes tipos de *IDS e* são classificados por funcionalidade, nas seguintes categorias:

- *Network-Based Intrusion Detection System (NIDS)*
- *Host-Based Intrusion Detection System (HIDS)*
- *Distributed Intrusion Detection System (DIDS)*

1.6.1.1 *NIDS (Network-based Intrusion Detection System)*

É chamado *IDS* de rede, pois monitora todo o segmento da rede ou sub-rede. Normalmente um adaptador de rede opera em modo não-promiscuo, ou seja, recebe apenas pacotes destinados para seu *MAC Address*, fazendo com que os outros pacotes sejam ignorados. Para monitorar todo o tráfego da rede, a interface de rede do *NIDS* deve aceitar todos os pacotes, para isto ser possível, deve trabalhar em modo promíscuo, além disso, o dispositivo de rede necessita estar configurado para enviar todo o tráfego para o *NIDS* (se o dispositivo for um *hub*, os pacotes são enviados automaticamente, mas, se for um *switch*, será necessário configurar para trabalhar como *mirroring port)*.

A implantação de um *NIDS* não provoca impacto nos sistemas ou redes que estão sendo monitoradas, além de não contribuir com nenhum tráfego adicional para os computadores da rede.

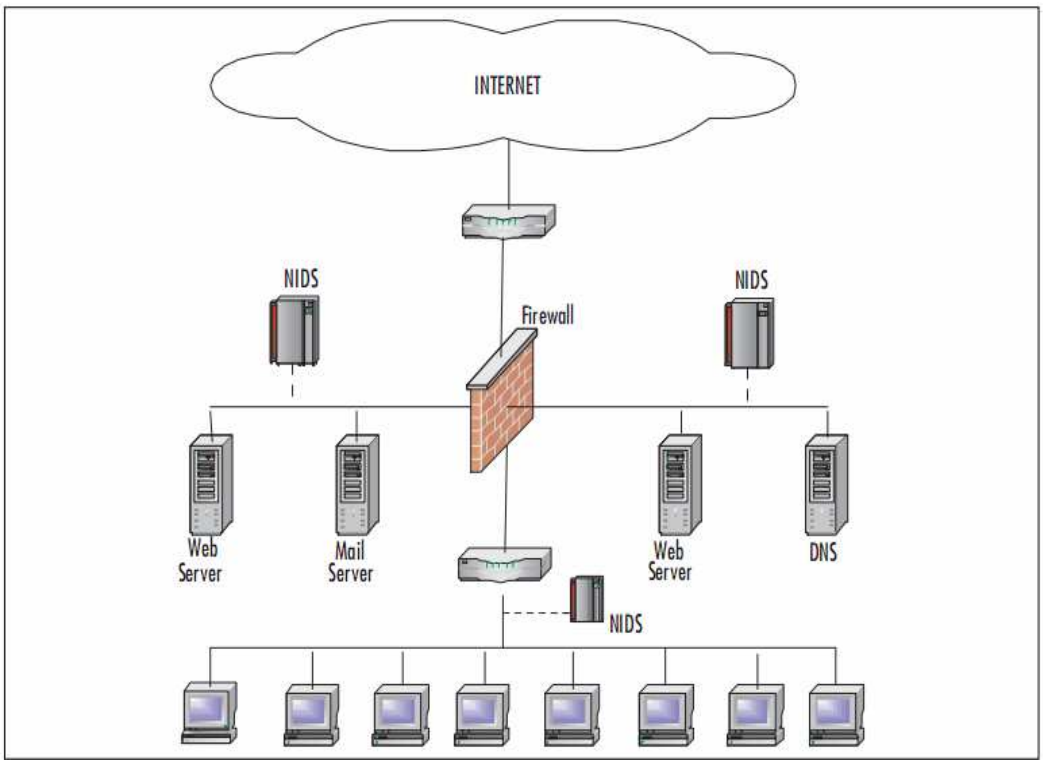

Figura 6 – Rede utilizando *NIDS*

(REHMAN, 2003)

A Figura 6 apresenta uma rede com três *NIDS*, distribuídos em segmentos estratégicos, possibilitando o monitoramento de toda a rede, no qual os servidores estão protegidos por dois *NIDS* e os computadores da rede interna possuem um *NIDS* adicional, para prevenir contra incidentes internos.

1.6.1.2 *HIDS (Host-based Intrusion Detection System)*

Os *HIDS* trabalham diferentemente dos *NIDS*, eles monitoram somente o sistema no qual está instalado, e conseqüentemente, não trabalham em modo promiscuo. As regras são especificas para cada *host* e não para todo o segmento de rede, diminuindo muito o número de alertas falsos positivos. Uma grande vantagem é a detecção de mudanças nos arquivos e no sistema operacional, fazendo *checksum* e monitorando o tamanho dos arquivos, para verificar se foram modificados de forma maliciosa. Além disso, um *HIDS* pode detectar informações que não trafeguem pela rede e que não seria notado por um *NIDS.*

Há pontos que devem ser levados em consideração, como o sistema operacional em que se deseja implantar os *HIDS*, pois nem todas as ferramentas *HIDS* são compatíveis com os sistemas operacionais do mercado, podendo ser necessário utilizar *HIDS* diferentes para sistemas operacionais distintos, além de adicionar certo tráfego no host em que está instalado, consumindo recursos computacionais, que podem comprometer a estabilidade de servidores. Dependendo do número de *HIDS* instalados na rede, a manutenção pode ser trabalhosa, necessitando implantar uma solução de gerenciamento centralizado.

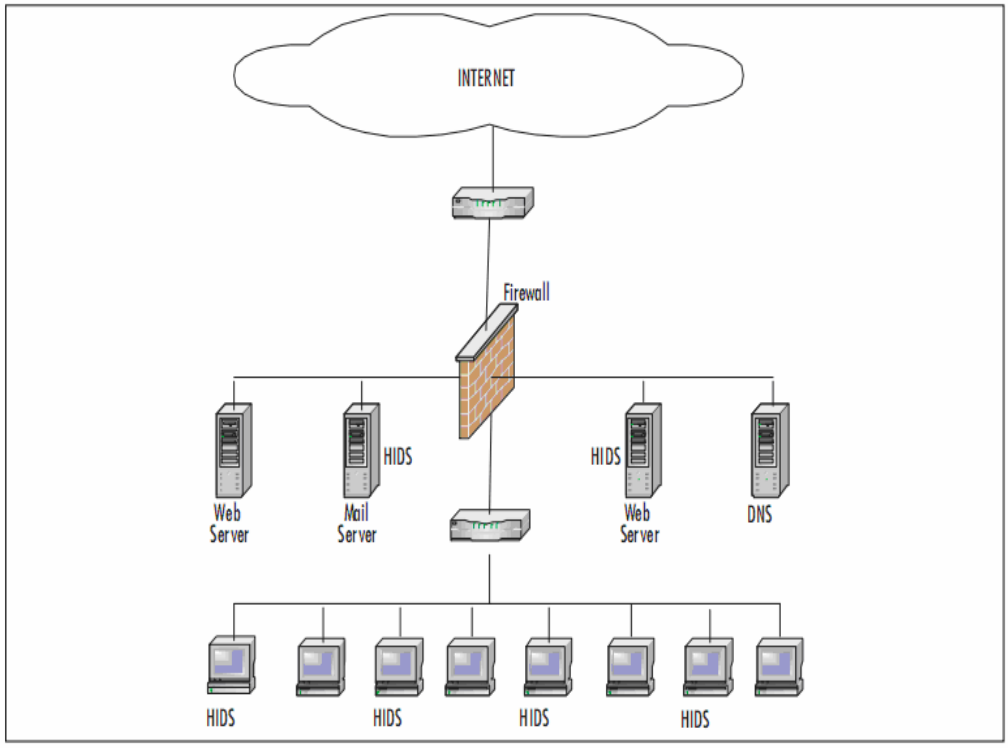

Figura 7 - Rede utilizando *HIDS*

(REHMAN, 2003)

A Figura 7 apresenta HIDS instalados em hosts e servidores, os quais necessitam de configuração personalizada, pois *hosts* e servidores devem ser monitorados de acordo com suas respectivas funções, além do fato que servidores diferentes fornecem serviços distintos.

1.6.1.3 *DIDS (Distributed Intrusion Detection System)*

Um Sistema de Detecção de Intrusão distribuído é uma combinação de sensores *NIDS* e/ou *HIDS* distribuídos por toda a rede. Todos os *logs* de atividades maliciosas que os sensores da rede são enviados para o servidor central e armazenados no banco de dados. Cada sensor da rede deve ser configurado conforme suas necessidades individuais, levando em conta o host ou segmento da rede que o sensor irá monitorar.

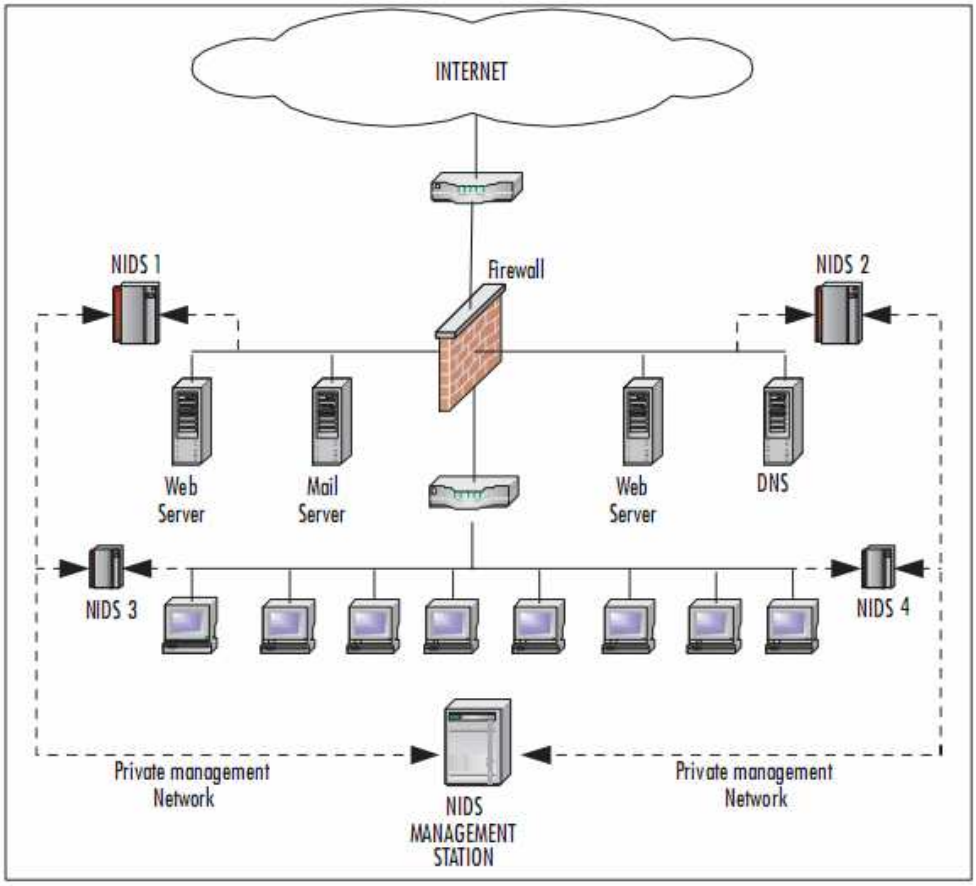

Figura 8 - Rede utilizando DIDS

(REHMAN, 2003)

Na Figura 8, é apresentado um sistema *DIDS*, com quatro sensores *NIDS* e um servidor de gerenciamento central. Os sensores *NIDS* 1 e *NIDS* 2 estão monitorando os servidores de *e-mail*, *DNS* e *Web* e os sensores *NIDS* 3 e *NIDS* 4 monitoram todos os *hosts* da rede.

1.6.2 Métodos de detecção

Os sistemas *IDS* podem utilizar dois métodos para identificar intrusões em uma rede ou host, o método que se baseia em assinaturas e o baseado em anomalia.

1.6.2.1 Baseado em assinaturas

Este método de detecção procura por atividades maliciosas comparando com padrões pré-definidos, ou seja, assinaturas que correspondem a determinados tipos de ataque, semelhante á maneira que os antivírus verificam os arquivos á procura de *malwares.* O funcionamento é eficiente contra ataques conhecidos, o que não se pode afirmar em relação aos ataques que não estão no banco de assinaturas, por este motivo, deve-se atualizar periodicamente o conjunto de assinaturas.

1.6.2.2 Baseada em anomalia

Utiliza regras ou conceitos predefinidos sobre atividades padrões e quando há um desvio, é enviado um alerta. São elaborados perfis de comportamento normal de usuários, *hosts*, rede ou aplicações, através de monitoramento das atividades por um período. É importante destacar que *IDS* baseados em anomalias são efetivos contra atividades maliciosas desconhecidas, por outro lado, causam grande número de falsos positivos, devido à possibilidade dos ambientes serem diversificados e dinâmicos, além da dificuldade para identificar a origem do ocorrido e relatar se ele é realmente uma ameaça ou um falso positivo.

1.7 Exemplos de IDS

Existem vários sistemas *IDS* disponíveis no mercado, tanto soluções *Open Source* como o *Snort* e o *Suricata*, quanto soluções proprietárias, como os sistemas da *Enterasys* e da *Cisco*. A maioria dos sistemas IDS atuais possuem a função de IPS, possibilitando também a prevenção de ataques.

1.7.1 Snort

O Snort é um *IDS/IPS* de rede (*NIDS*) *Open Source* muito popular, possui constantes atualizações nas regras de detecção, é leve e pequeno. Ele faz análises em tempo real dos protocolos e possui grande número de opções para tratamento dos alertas.

Pode ser utilizado em plataformas *Linux, OpenBSD, FreeBSD, NetBSD, Solaris, SunOS, HP-UX, AIX, IRIX, Tru64, MacOS X Server* e *Windows.*

O Snort possui três módulos, entre eles o *sniffer* que captura pacotes, o *packet logger* que registra os pacotes capturados e a sua principal função, o *NIDS,* que faz a análise do tráfego de rede á procura de tentativas de ataques.

Suas principais características são a flexibilidade, uso de algoritmos de inspeção baseados em regras, baixa quantidade de falsos positivos inerentes, assinaturas do ataque, anomalias no protocolo, imensa adoção (comunidade *Snort*), milhares de contribuidores fazendo regras para novas vulnerabilidades, suporte da Comunidade *Open Source* e rápida resposta às ameaças.

1.7.2 *Suricata*

O *Suricata* é o mais recente entre os sistemas *IDS/IPS*, sua versão *beta* foi lançada em 31 de dezembro de 2009. O sistema foi desenvolvido pelo *Open Information Security Foundation (OISF)*, um grupo multinacional de desenvolvedores de software na área de segurança.

Suas principais características são o Suporte a *Multi-threading* e a detecção automática de protocolos, além de ter opções de *logs* em *HTTP* e *PostgreSQL*. Ele é um sistema desenvolvido recentemente, possuindo projetos para ampliação de suas funcionalidades.

É compatível com o conjunto de regras do *Snort*, facilitando muito a migração para o *Suricata*. Pode ser implantado em diversos sistemas operacionais, como o *FreeBSD*, *Linux*, *UNIX, Mac OS X e Microsoft Windows*

1.7.3 *Enterasys Instrusion Prevention/Detection System*

O *Enterasys Instrusion Prevention/Detection System*, também conhecido por *Dragon IPS*, é um *software/hardware* proprietário desenvolvido pela *Enterasys*, empresa muito reconhecida na área de redes, com foco em grandes empresas e seus principais produtos são *switches*, roteadores e *softwares* de segurança de redes, como o *IDS/IPS Dragon*.

Possui dois tipos de sensores, o sensor de rede e o sensor baseado em *host*. Os sensores de rede (*NIDS*) da *Enterasys* possuem várias velocidades disponíveis, que vão de 100 mbps até 1 gbps, também suportando *multi gigabit*. As principais características são ferramentas forenses como captura de pacotes flexíveis e sessão de reconstrução completa, suporte *multi-thread* e detecção de múltiplos algoritmos simultâneos, sensores virtuais associando com Virtual LAN (*VLAN).*

Os sensores baseados em *host* detectam ataques no *host* em tempo real, monitora atributos de arquivos específicos do sistema, checa a integridade de arquivos críticos, analisa os *logs* do sistema, analisa registros do *Windows*, além de monitorar o *kernel* e serviços *TCP/UDP*. O *Enterasys Host Based Sensors* oferece suporte as plataformas *Windows, Solaris, Red Hat Enterprise Linux, HP-UX, Fedora Core, SUSE* e *AIX.*

O sistema da *Enterasys* possui o *EMS (Enterprise Management Server)* com arquitetura cliente-servidor, oferecendo um gerenciamento eficiente e centralizado para todos os sensores *Enterasys* da rede.

1.7.4 *Cisco Intrusion Prevention/Detection System*

Sistema de Detecção e Prevenção de Intrusão proprietário da Cisco, considerada a maior empresa em soluções para redes de computadores e telecomunicações, oferecendo equipamentos, softwares e prestação de serviços. É um sistema baseado em redes (*NIDS*) que identifica, classifica e bloqueia invasões conhecidas e não conhecidas.

Recomendado para se obter o máximo de segurança em redes que utilizam equipamentos *Cisco*, possui atualizações constantes e adiciona proteção contra mais de 30.000 ameaças conhecidas.

2 DESENVOLVIMENTO

Para efetuar os testes em uma ferramenta de Detecção de Intrusão, foi elaborado um cenário, contendo um *host* com o sistema *IDS Snort*, que receberá simulações de ataques.

2.1 Cenário

Os testes foram realizados a partir de um único computador (máquina real), utilizando-se máquinas virtuais.

2.1.1 Máquina real

O sistema operacional utilizado foi o *Windows XP Professional*, com licença *MSDNAA (Microsoft Developer Network Academic Alliance*). Esta máquina irá efetuar simulações de ataques utilizando a ferramenta *Nessus* 4.2.2 (Figura 9), com licença *Home Feed* (gratuito).

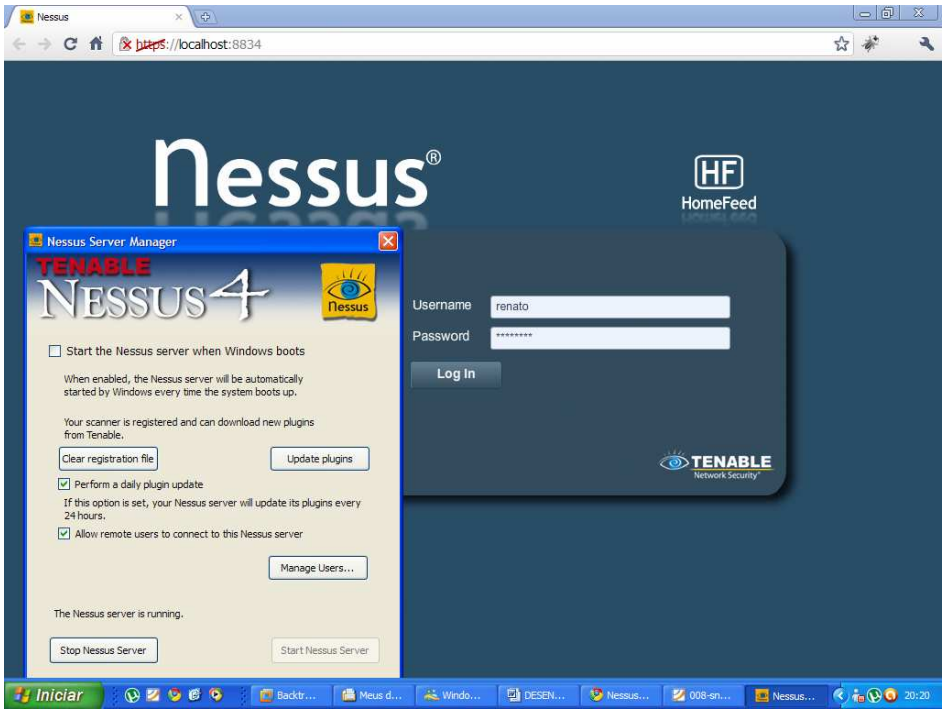

Figura 9 - Máquina real com *Windows XP Pro Edition* e a Ferramenta *Nessus*

2.2 Máquinas virtuais

A virtualização foi feita pela ferramenta gratuita *VMware Player* 3.1.2. Foram virtualizados dois *hosts*, ambos com sistema operacional *Linux*, mas com distribuições diferentes, uma delas utilizando o *Ubuntu* e a outra com a distribuição *Backtrack*.

2.2.1 *Ubuntu*

Foi instalado o sistema operacional *Ubuntu 10.10* (Figura 10), juntamente com a ferramenta de Detecção de Intrusão *Snort* 2.9.0, utilizando o servidor de banco de dados *MySQL* e com o módulo adicional *Snort Report*, que atribui uma interface *Web para o Snort*, apresentando em tempo real os alertas registrados e armazenadaos na base de dados.

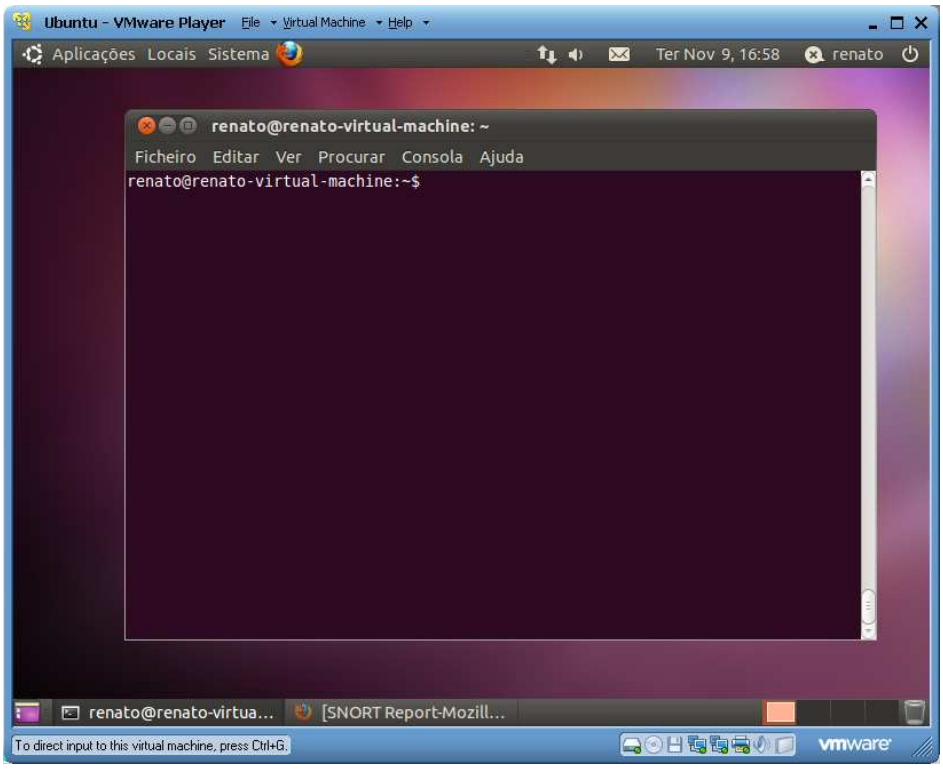

Figura 10 - Maquina Virtual com o sistema operacional *Ubuntu 10.10*

2.2.2 *Backtrack*

Nesta máquina virtual foi instalada o sistema operacional *Backtrack* 4 R1 (Figura 11), uma distribuição Linux focada em *pen test* (teste de penetração), que irá simular ataques na rede e na máquina virtual *Ubuntu*.

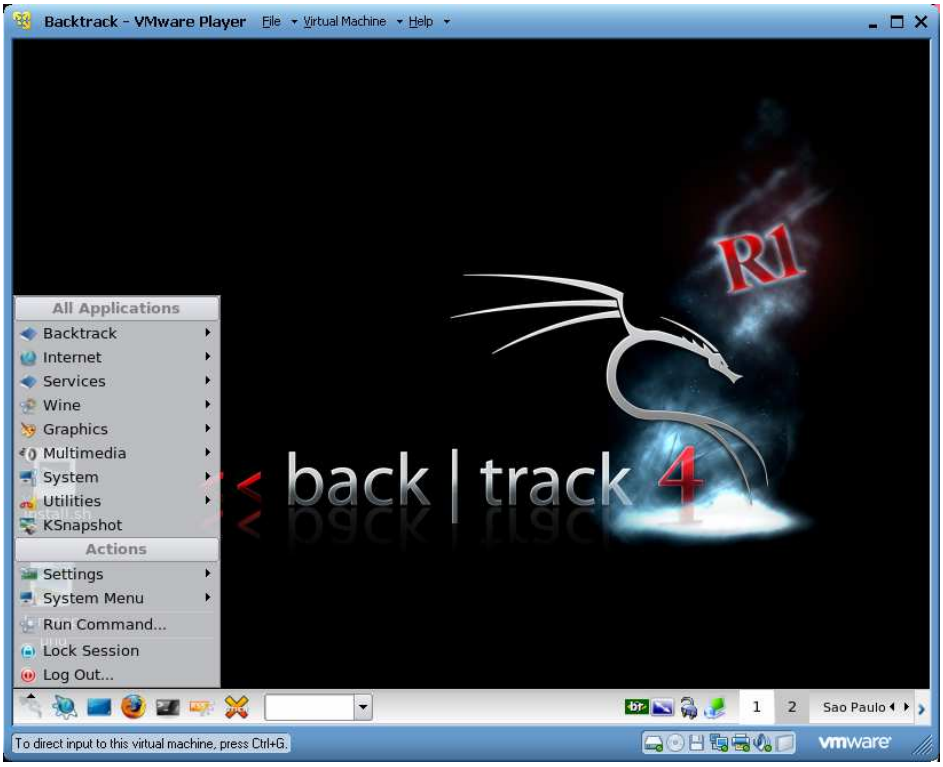

Figura 11 - Máquina Virtual com o sistema operacional *Backtrack*

3 TESTES E RESULTADOS

Os resultados foram obtidos a partir de testes da ferramenta de Detecção de Intrusão Snort, através de simulações de ataques. Serão apresentados os resultados de diferentes tipos de ataques e os respectivos alertas gerados pelo sistema.

3.1 *Nessus*

O *Nessus* é uma ferramenta de detecção de falhas e vulnerabilidades de segurança em um sistema ou rede, fazendo varreduras de portas e simulando invasões para detectar as vulnerabilidades, além de apresentar as possíveis soluções para as respectivas falhas. O *Nessus* será utilizado para efetuar simulações de ataques na rede e no *host* com o sistema *IDS* Snort implantado.

3.1.1 Varredura completa

Esta varredura irá verificar todos os *IP's* da rede 192.168.246.0, utilizando os 42 tipos de simulações de ataques (totalizando 39128 *plugins* de vulnerabilidades) disponíveis nesta versão do *Nessus (Figura 12)*.

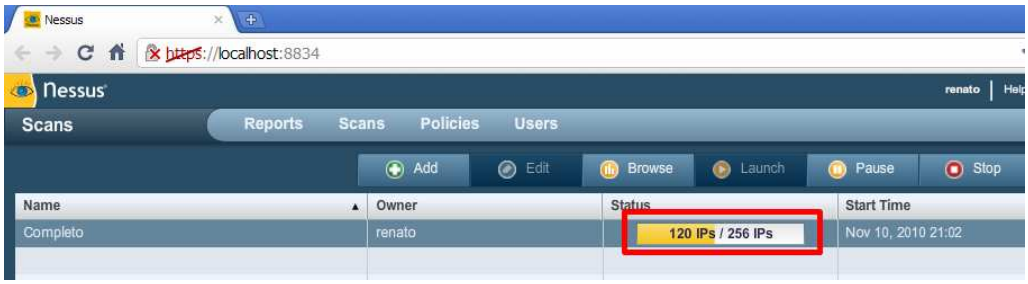

Figura 12 - Varredura completa da rede utilizando o *Nessus*

Após o termino da varredura completa utilizando a ferramenta *Nessus*, foram obtidos 167 alertas com 38 assinaturas diferentes no sistema *IDS Snort* (Figura 13).

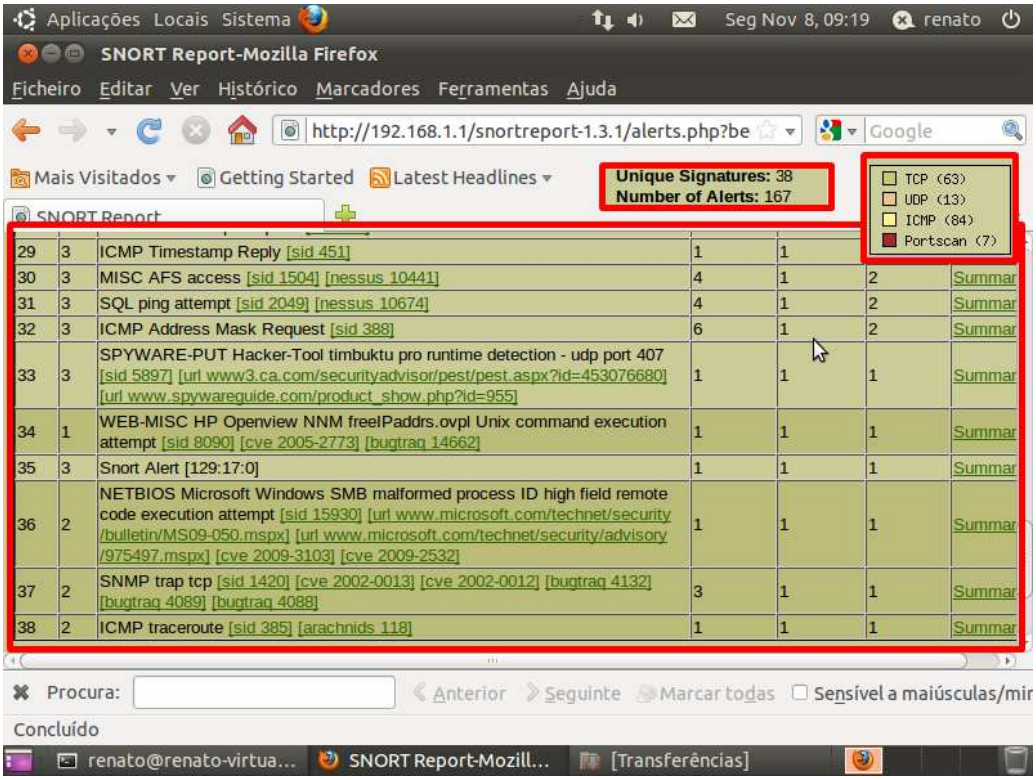

Figura 13 - Alertas gerados pelo *Snort* após a varredura do *Nessus*

Os alertas gerados pelo *IDS Snort* detectaram como *IP* de origem o endereço 192.168.246.1 (*IP* da máquina *Windows*), como mostra a Figura 14.

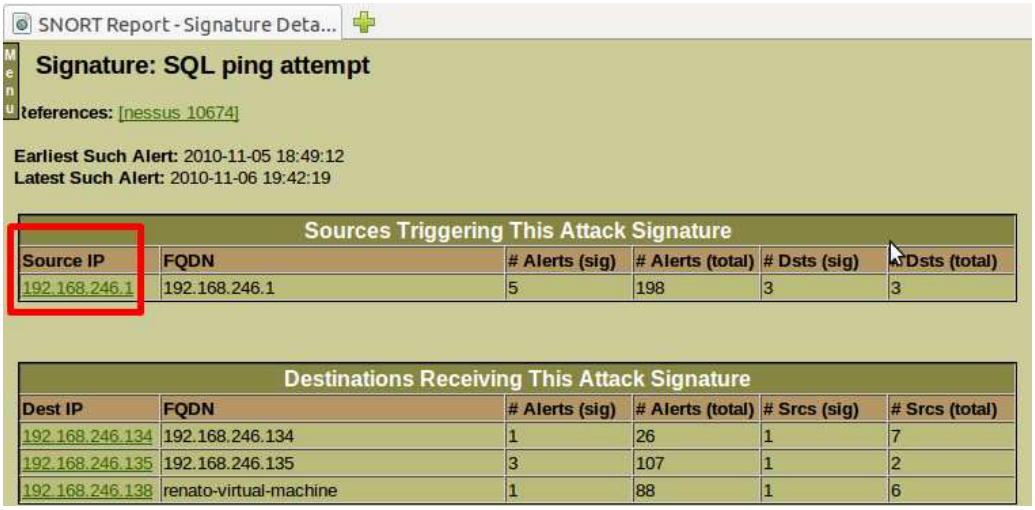

Figura 14 – *IP's* de origem e destino identificados pelo *Snort*

3.1.2 *Denial of Service (DoS)*

Nesta etapa foram utilizados apenas os plugins que simulam ataques *DoS*, com o intuito de verificar alertas específicos gerados pelo sistema *Snort* sofrendo este tipo de ataque. Estão disponíveis 92 *plugins* que simulam diversos ataques DoS (Figura 15).

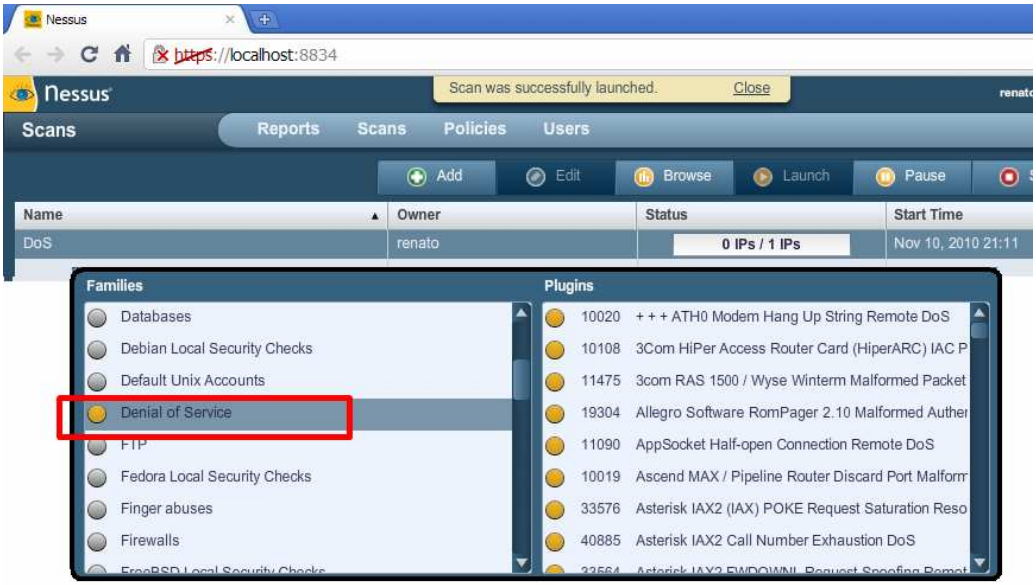

Figura 15 - *Plugins DoS* utilizados na varredura.

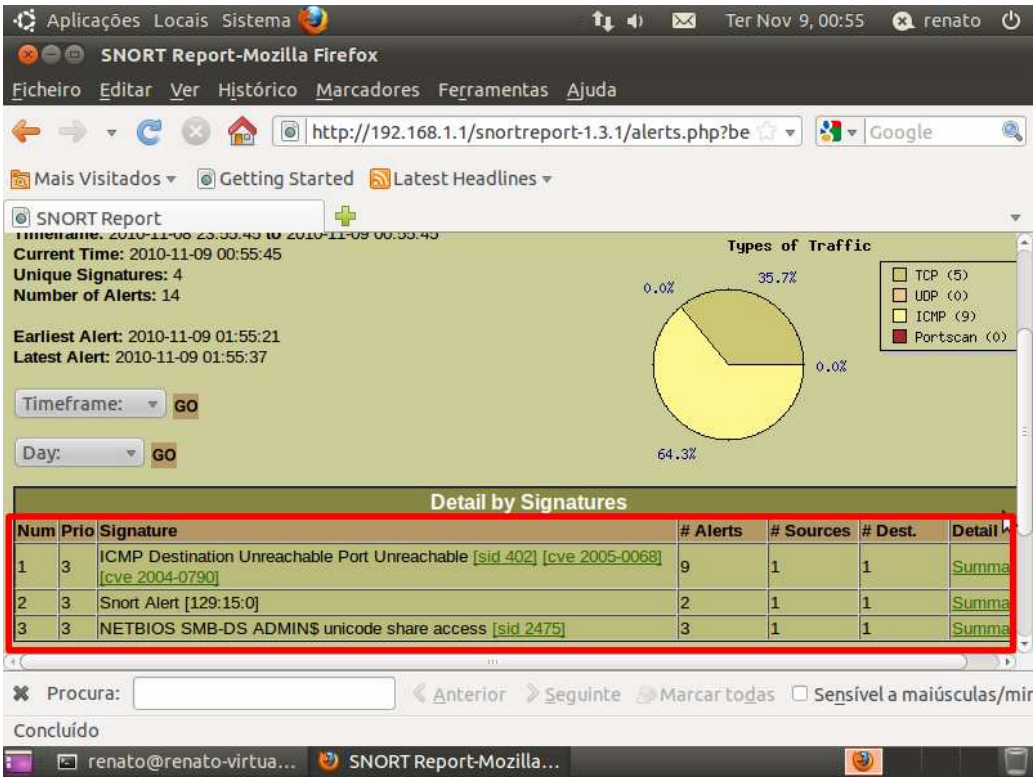

Figura 16 - Alertas gerados pelo *IDS Snort* após as simulações de ataque *DoS*.

O sistema *IDS Snort* gerou os alertas apresentados na Figura 16, como é possível observar, foram encontradas poucas tentativas de ataque, pois foram utilizados somente os *plugins* de ataques *DoS*.

3.1.3 *Backdoors*

O *Nessus* possui *plugins* que simulam *backdoors* e estes serão utilizados na máquina com o *IDS* implantado, para verificar as reações geradas pelo sistema. Serão utilizados 87 *plugins* para este teste (Figura 17).

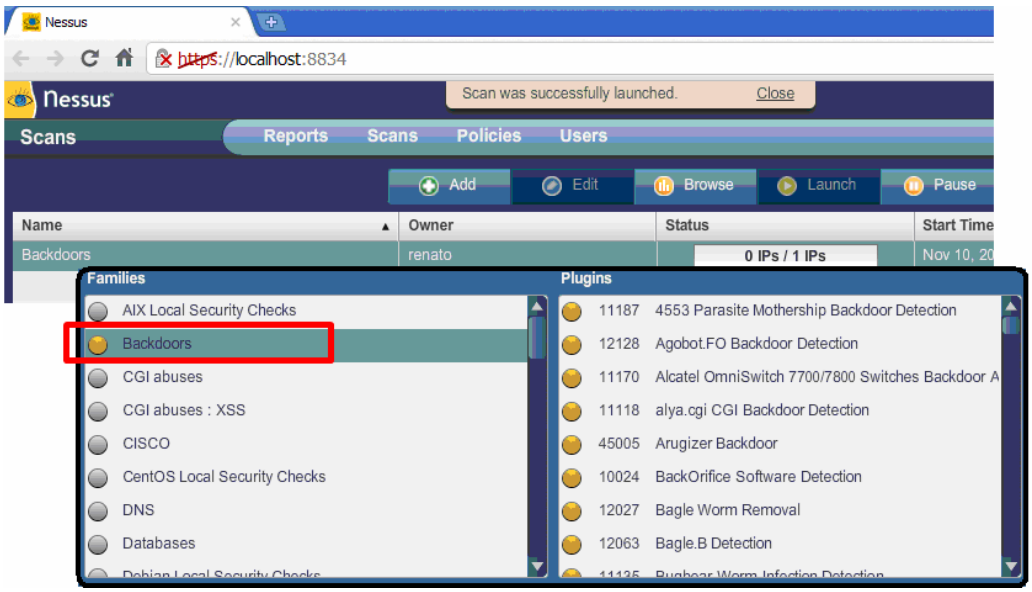

Figura 17 - Simulações de ataque utilizando *Backdoors*.

O sistema *IDS* gerou 27 alertas relacionados a simulação deste ataque. Nesta simulação podemos observar novos tipos de tráfego em relação aos ataques *DoS*. O tráfego *portscan* e *UDP* são utilizado por estas simulações, afirmando as particularidade dos diferentes tipos de ataque. O tipo de tráfego é apresentado no gráfico, localizando-se na parte superior direita da Figura 18.

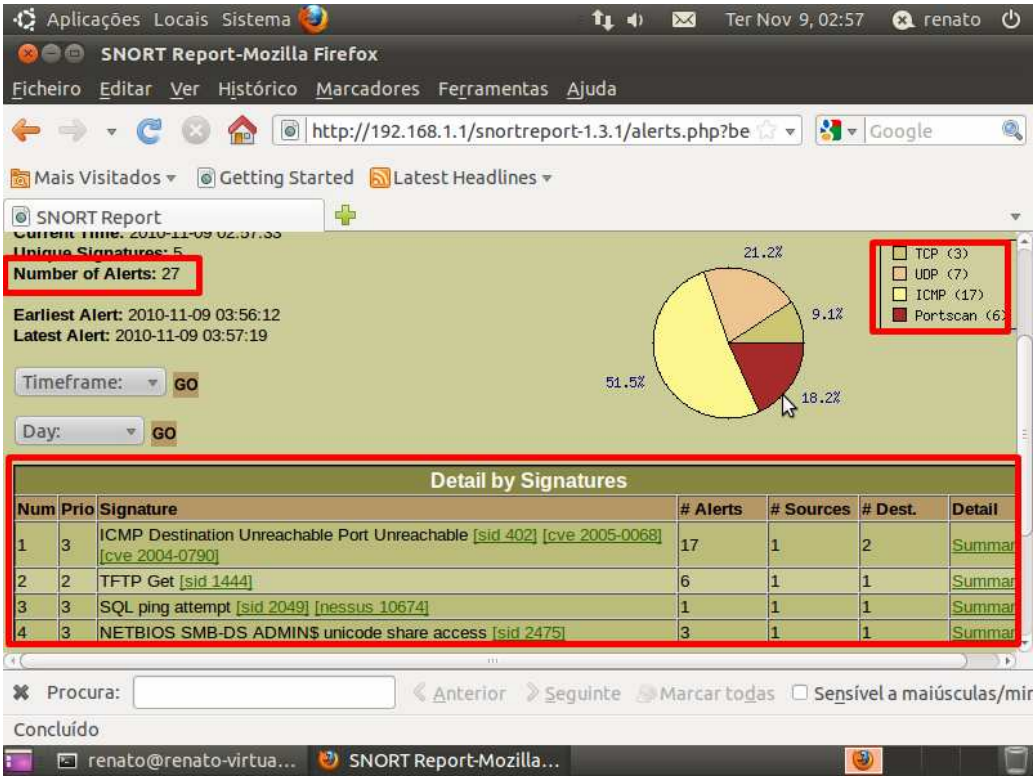

Figura 18 - Alertas gerados pelo *Snort* após as simulações de *backdoors*.

3.2 *Backtrack*

Foram utilizadas algumas ferramentas disponíveis na distribuição *Backtrack* 4 R1, para realizar simulações de ataques no *host* com o *IDS Snort* implantado.

3.2.1 *BruteSSH*

O *BruteSSH* é uma ferramenta simples que utiliza força bruta para quebrar senhas e obter acesso a um sistema via *SSH* (Figura 19). A ferramenta foi executada com *IP* de destino 192.168.246.138 (máquina Ubuntu com o sistema *IDS Snort*), para tentar quebrar a senha de *root* a partir do arquivo *pass.txt* (arquivo com uma lista de possíveis senhas).

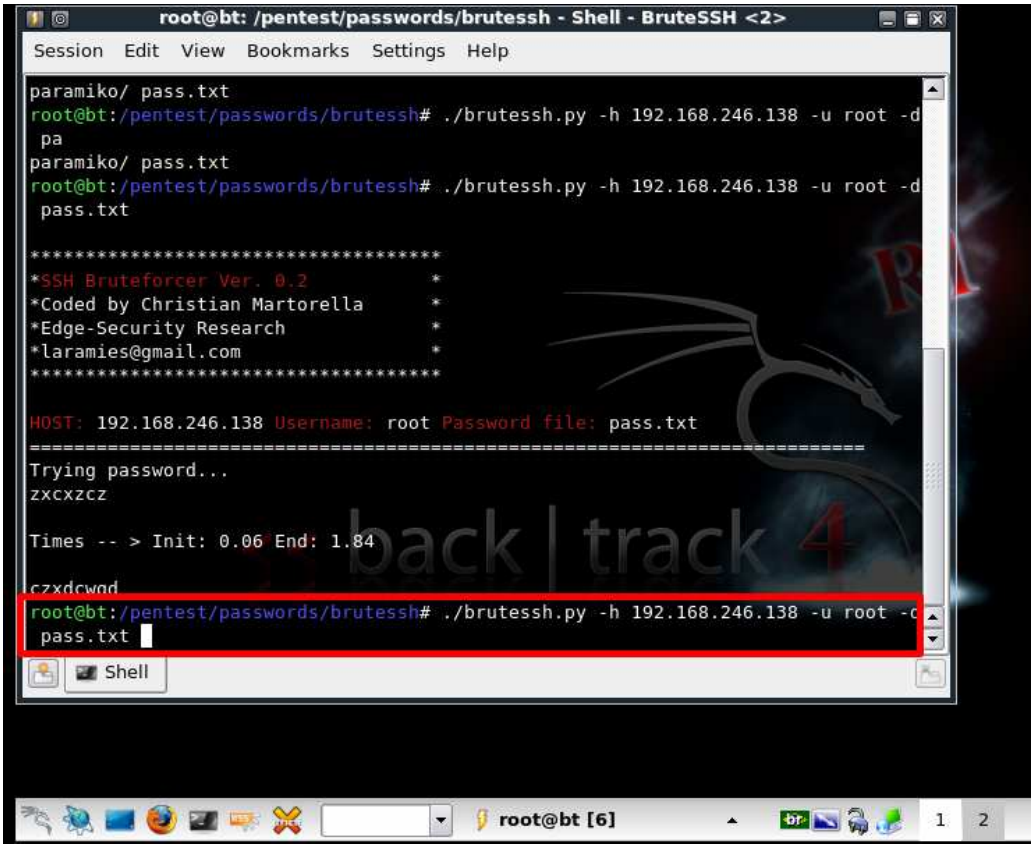

Figura 19 - Utilizando o *BruteSSH* para quebrar a senha de *root*.

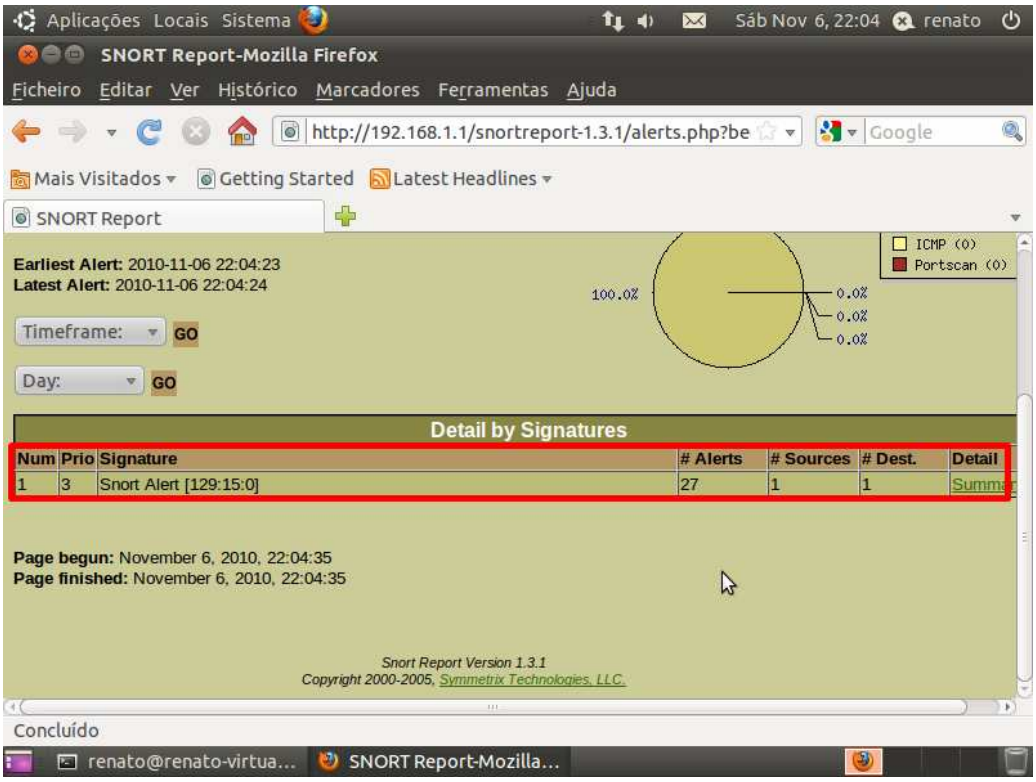

Figura 20 - Alertas gerados após a execução do *BruteSSH*.

O sistema Snort detectou estas tentativas de quebra de senha utilizando força bruta, gerando 27 alertas, mas o sistema não identificou a assinatura deste tipo de ataque, apenas apresentou como *Snort Alert*, como mostra a Figura 20.

3.2.2. *Httprint*

O *httprint* é uma ferramenta que obtêm informações (como o sistema operacional e a versão do servidor *Web*) de uma máquina que possua um servidor *Web* disponível. Foi utilizada a versão 0.301 com interface gráfica para efetuar o teste no servidor *Web* da máquina *Ubuntu (Figura 21).*

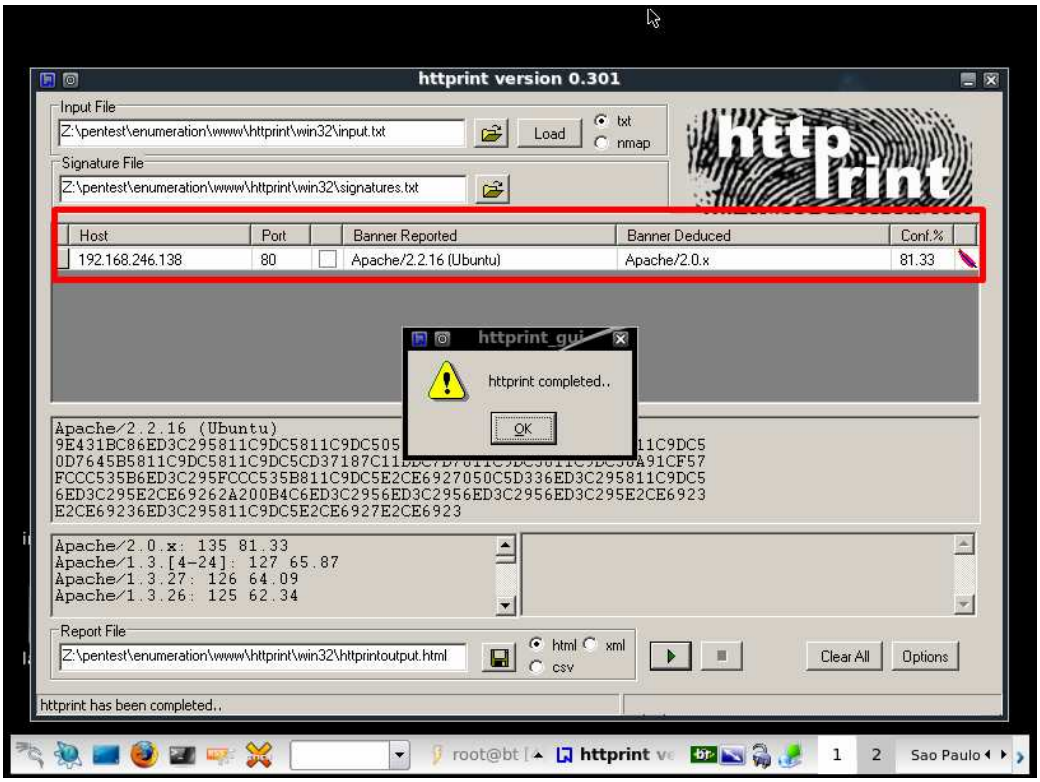

Figura 21 - Obtendo informações do servidor *Web* da máquina *Ubuntu*.

Foram obtidos 3 alertas pelo sistema *IDS*, entre eles o *ATTACK-RESPONSES 403 Forbidden*, que é uma assinatura ligada diretamente a páginas inacessíveis ou que o usuário não possui acesso no servidor *Web (Figura 22)*.

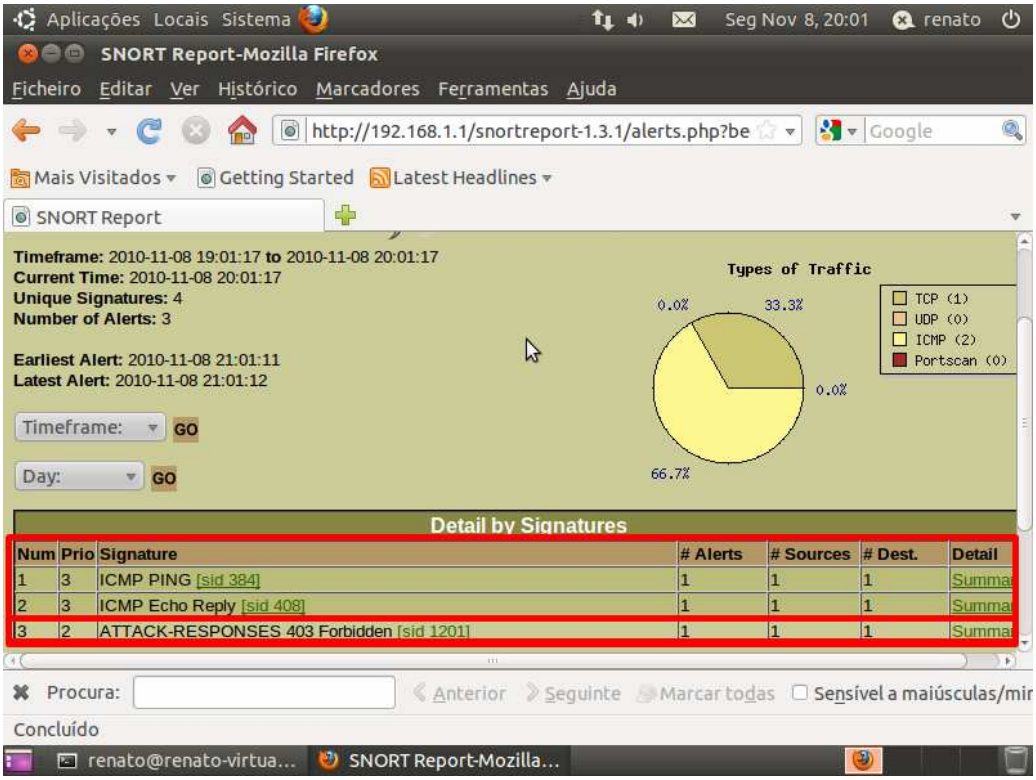

Figura 22 - Alertas gerados pelo *Snort* após utilizar a ferramenta *httprint.*

3.2.3. NMAP

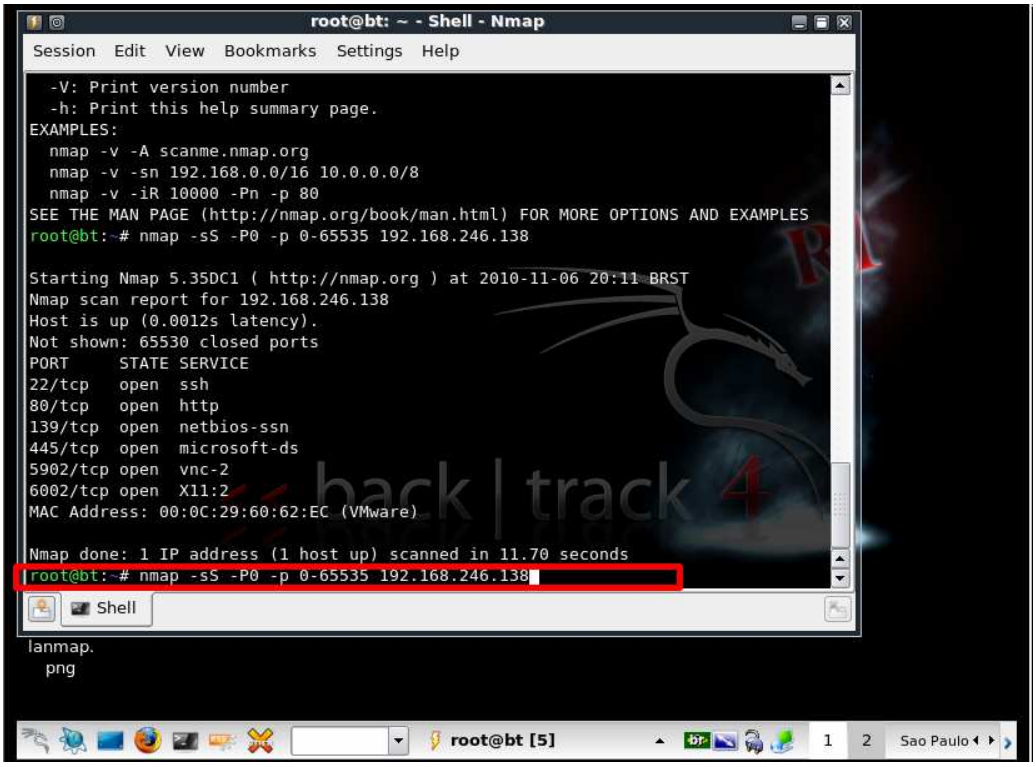

Figura 23 - Buscando informações da máquina *Ubuntu* utilizado a ferramenta *NMAP*

O *nmap* é um *port scan* que verifica portas abertas e obtêm também outras informações (o sistema operacional utilizado na máquina, por exemplo) de um *host* ou de uma rede. Foi feito uma varredura na máquina *Ubuntu* com diversos parâmetros, como o parâmetro –sS, que procura por portas escondidas (Figura 23).

Foram gerados vários alertas para esta varredura, confirmando a identificação de ferramentas de *port scanner* em uma rede pelos sistemas *IDS* (Figura 24).

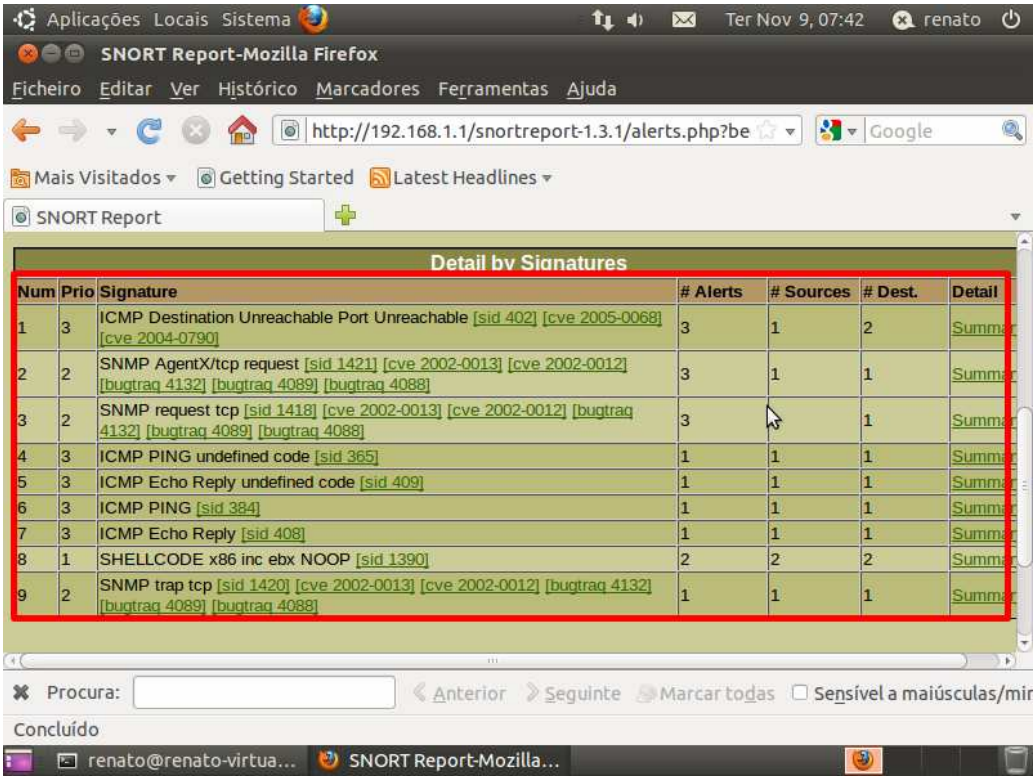

Figura 24 - Alertas após a varredura utilizando *NMAP*

3.2.4. FALSOS POSITIVOS

Foram utilizadas regras padrão (*Snortrules-snapshot*-2.9) nos testes, o que pode gerar vários alarmes falsos, como apresenta a Figura 25, que apenas acessando uma conta de *webmail* e utilizando o comando *ping* da máquina *Windows*, o Snort gerou vários alarmes, que na realidade não são referentes a ataques.

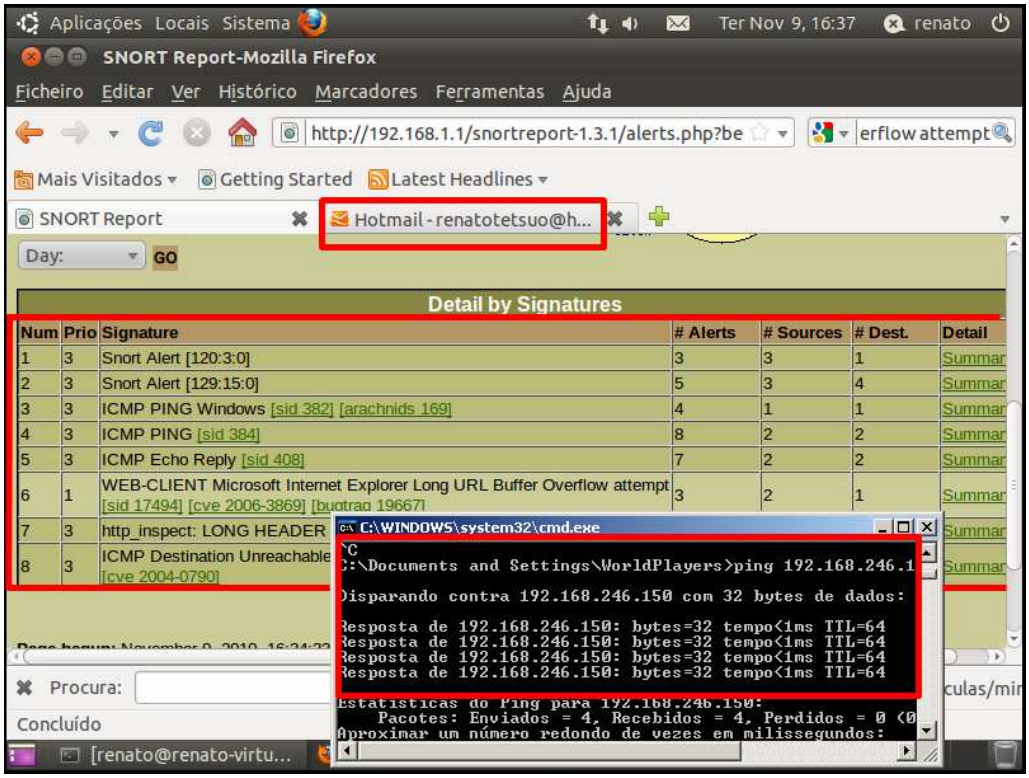

Figura 25 - Alarmes falsos gerados pelo *IDS.*

CONCLUSÃO

A segurança da informação é de grande importância, como já foi dito anteriormente, devendo ser verificada constantemente, pois novas vulnerabilidades são descobertas a qualquer instante e as informações confidenciais necessitam permanecer protegidas contra acessos indevidos, sendo altamente aconselhável a implantação de ferramentas de segurança, para acrescentar maior confiabilidade no tráfego das informações e uma das soluções viáveis é a implantação de um Sistema de Detecção de Intrusão.

O principal objetivo de um Sistema de Detecção de Intrusão é gerar alertas quando houver tentativas de invasão em um *host* ou em uma rede, então, com o intuito de verificar o funcionamento de um IDS, foram efetuados testes nesta ferramenta, observando os alertas gerados pelo sistema. Verificando os resultados destes testes em um cenário, com o sistema *IDS Snort* implantado, foi possível observar que o sistema realmente detectou as tentativas de ataque, tanto das simulações da ferramenta *Nessus, quanto* das ferramentas de *pen test* do *Backtrack.* Algumas simulações foram detectadas como *Snort Alert*, ou seja, significa que o sistema verificou o tráfego malicioso, mas apenas não identificou sua assinatura.

Há também a questão de falsos positivos verificados durante os testes, cujos alertas não são referentes a verdadeiros ataques, mas possuem características semelhantes, fazendo com que o *IDS* apresente alertas sobre este tráfego. Foram detectados vários falsos positivos durante o período de testes devido à utilização de regras padrão do Snort, sendo aconselhável a implantação de regras personalizadas para cada ambiente específico, verificando as necessidades e o comportamento da rede ou hosts em questão.

Com a ferramenta *Nessus* foram utilizados 39128 *plugins* para a simulação de 42 diferentes tipos de ataque e o sistema IDS gerou apenas 167 alertas e 38 assinaturas diferentes de ataques (omitindo vários alertas de ataques simulados pelos plugins). Neste caso, há diversos fatores que podem interferir nas detecções, entre elas, a inexistência de

regras do Snort que detectem estas assinaturas de ataques, ou as alternativas mais possíveis, como a porta estar bloqueada no *firewall* ou que o sistema operacional já está protegido contra estas falhas e bloqueou antes mesmo de chegar ao sistema *IDS*, pois o sistema operacional está atualizado (Ubuntu 10.10, com todas as atualizações instaladas), já possuindo correções de segurança relacionadas aos referentes ataques.

Os ataques identificados pelo sistema *IDS Snort* podem ser bloqueados automaticamente utilizando ferramentas como o *Guardian* e o *Snortsam*, que possibilita o trabalho em conjunto do sistema *IDS Snor*t e o *firewall*. Eles atualizam as regras do *firewall* de acordo com os alertas gerados, podendo definir várias configurações, como o bloqueio automático e o tempo que determinado *IP* permanecerá bloqueado.

O Sistema de Detecção de Intrusão é uma importante ferramenta para prevenir e identificar ataques em uma rede, alertando sobre as tentativas de ataque e possibilitando que administrador da rede analise e tome as decisões necessárias sobre este tráfego malicioso. A segurança é aprimorada com o trabalho em conjunto do sistema *IDS* e o *firewall*, pois o bloqueio será automático e em tempo real, mas é válido ressaltar a necessidade de todo um planejamento e configuração adequada para cada ambiente, a fim de evitar alarmes falsos e o bloqueio indevido de endereços IPs.

REFERÊNCIAS BIBLIOGRÁFICAS

ALECRIM, Emerson. Ataques de engenharia social na Internet, 2004. Disponível em: <http://www.infowester.com/col120904.php>. Acesso em: 11 set. 2010.

ALECRIM, Emerson. Ataques *DoS* (*Denial of Service*) *e DDoS (Distributed DoS*). 2004. Disponível em: <http://www.infowester.com/col091004.php>. Acesso em: 29 set. 2010

Centro de Estudos Resposta e Tratamento de Incidentes de Segurança no Brasil. Estatísticas dos Incidentes Reportados ao CERT.br. Disponível em: <http://www.cert.br/stats/incidentes/>. Acesso em: 01 out. 2010.

Centro de Estudos Resposta e Tratamento de Incidentes de Segurança no Brasil. Cartilha de Segurança para Internet. Disponível em: <http://cartilha.cert.br/>. Acesso em 02 out. 2010.

MORIMOTO, Carlos E. Redes e Servidores Linux 2ed. *GdH Press* (Versão digital): São Paulo. 2006.

REHMAN, Rafeeq U. *Intrusion Detection with Snort – Advanced Techniques using Snort, Apache, MySQL, PHP, and Acid. Pearson Education, Inc.: New Jersey*. 2003.

RUBIN, Aviel D.; CHESWICK, William R. *White-hat security arsenal: tackling the threats. Boston: Addison-Wesley*, 2001.

SOUZA, Marcelo. Readaptação do modelo ACME para detecção de novas técnicas de intrusão. Monografia de Graduação. UNESP – Departamento de Ciência da Computação e Estatística, São José do Rio Preto - SP, 2002.

BIBLIOGRAFIA CONSULTADA

Backtrack Penetration Testing Distribution. Disponível em:<http://www.backtrack-linux.org/>. Acesso em: 10 out. 2010

BEALE, Jay; CASWELL, Brian. Snort 2.1 Intrusion Detection 2ed. United States o f America: Singress Publishing, Inc. 2004.

BRADLEY, Tony. *Introduction to Port Scanning*, [entre 2000 e 2010]. Disponível em: <http://netsecurity.about.com/cs/hackertools/a/aa121303.htm>. Acesso em: 30 set. 2010.

Cisco Systems, inc. Cisco Intrusion Prevention/Detection System. Disponível em: <http://www.cisco.com/en/US/products/sw/secursw/ps2113/index.html>. Acesso em: 22 out. 2010.

Enterasys Secure Network. Enterasys Intrusion Prevention/Detection. Disponível em: <http://www.enterasys.com/products/advanced-security-apps/dragon-intrusion-detection-protection.aspx>. Acesso em: 22 out. 2010

GULLETT, David, *Snort 2.9.0 and Snort Report 1.3.1 on Ubuntu 10.04 LTS Installation Guide*. 2010. Disponível em: <http://www.symmetrixtech.com/articles/004-snortinstallguide286.html>. Acesso em: 10 out. 2010.

KRZIZANOWSKI, David. Sistema de Controle de Acesso de *Notebooks, Desktops* e Ativos de Rede em uma *LAN*. Trabalho de Conclusão de Curso. Universidade Regional de Blumenau. Bacharelado - Sistemas de Informação. Blumenau – SC. 2006.

Nessus, The Network Vulnerability Scanner. Disponível em: <http://www.nessus.org/nessus/>. Acesso em: 10 out. 2010.

NORTHCUTT, Stephen; NOVAK, Judy. *Network Intrusion Detection 3ed. Indiana: New Riders Publishing*. 2002

OLIVEIRA, Anderson K.; HIRANO, Leandro T. Análise de Segurança – Sistemas de Detecção de Intrusão. Universidade Luterana do Brasil, ULBRA. Curso de Redes de Computadores. [entre 2008 e 2010]

OLIVEIRA, Rodrigo A. Desenvolvimento de uma Estrutura de Resposta Ativa Utilizando o *IDS Snort*. Universidade do Oeste Paulista - Curso de Ciência da Computação. Presidente Prudente - SP. 2004

PATRÍCIO, Daiane C., et al. Detecção de Intrusão. Artigo – UNESC – Ciência da Computação. Criciúma – SC. [entre 2005 e 2010]

SANTOS, Bruno R. Detecção de Intrusos Utilizando o *Snort* - Monografia. Universidade Federal de Lavras - Lavras - MG. 2005

The Open Information Security Foundation (OISF). *Suricata IDS/IPS.* Disponível em: <http://www.openinfosecfoundation.org/index.php>. Acesso em: 22 out. 2010.

Ubuntu Official Site. Disponível em: <http://www.ubuntu.com/>. Acesso em: 10 out. 2010.

VELOSO, Rene R. *Defacements*. [entre 2005 e 2010]. Disponível em: <http://www.comp.pucpcaldas.br/~al550069715/defaced.htm>. Acesso em: 22 out. 2010.

VMware Virtualization Software. Disponível em: <http://www.vmware.com/products/player/>. Acesso em: 10 out. 2010