UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

ANANIAS TESSARO

# Utilização do Snort para a avaliação da eficácia de firewalls

Trabalho de Graduação.

Prof. Dr. Raul Fernando Weber
Orientador

Porto Alegre, dezembro de 2012.

# AGRADECIMENTOS

Agradeço primeiramente a meus pais que apoiaram e ajudaram de todas as formas nessa longa caminhada. Agradeço a minha noiva pelo apoio e compreensão nos momentos mais difíceis, principalmente durante a realização desse trabalho. Agradeço à UFRGS por me dar a chance de concluir um curso superior de extrema qualidade. Agradeço ao professor Weber por ter me aceitado como orientando e dado apoio para chegar ao fim dessa jornada. Agradeço a todos os mestres que ao longo do curso me ajudaram a chegar até aqui, sem sombra de duvidas, todos, de uma maneira ou outra, foram muito importantes isso.

Agradeço a todos que de uma forma ou outra me ajudaram para chegar aonde cheguei, do fundo do meu coração o meu muito obrigado.

# SUMÁRIO

# LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| APT | Advanced Package Tool |
| BASE | Basic Analysis and Security Engine |
| BD | Banco de Dados |
| DNS | Domain Name System |
| DoS | Denial of Service |
| IDPS | Intrusion Detection and Prevention Systems |
| IDS | Intrusion Detection System |
| IP | Internet Protocol |
| IPS | Intrusion Prevention System |
| MAC | Media Access Control |
| MNDP | MikroTik Neighbor Discovery Protocol |
| NIDS | Network Intrusion Detection System |
| SGBD | Sistema de gerenciamento de banco de dados |

# LISTA DE FIGURAS

# RESUMO

A informação, independente do seu formato, é um dos maiores patrimônios de uma organização moderna, sendo vital para quaisquer níveis hierárquicos e dentro de qualquer instituição que deseja manter-se competitiva no mercado. Ciente dessa importância, este trabalho pretende focar na segurança da informação. Tecnologias como Firewall, antivírus e Sistemas de Detecção de Intrusão têm a finalidade de ajudar a proteger as empresas de ataques e vazamento de dados pela rede, funcionando, de certa forma, como um cão de guarda virtual que assiste a rede e hosts específicos em busca de comportamento suspeito.

Dentre uma série de mecanismos relacionados à segurança das informações está o firewall que é responsável por filtrar a entrada e saída de pacotes das redes, peça fundamental que funciona como porta de entrada para muitos problemas.

As câmeras de segurança vigiam quem entra e quem sai de um local. Por que não usar o mesmo artifício para verificar o que está entrando e saindo de uma rede com a finalidade de avaliar a segurança dessa e descobrir possíveis vulnerabilidades dos mecanismos que deveriam impedir esse acesso?

A proposta desse trabalho foi fazer um estudo de conceitos básicos relacionados ao gerenciamento dos riscos relativos às falhas de dispositivos de segurança, um estudo básico de sistemas de firewall e sistemas de detecção de intrusão. Em seguida foram realizados testes com o IDS Snort e outras ferramentas de segurança para avaliar o tráfego de rede a fim de encontrar possíveis vulnerabilidades num sistema de firewall.

**Palavras-Chave:** Snort, detecção intrusão, firewall, vulnerabilidade.

# ABSTRACT

The information, regardless of format, is one of the greatest assets of a modern organization and is vital to any hierarchical level and within any institution that wishes to remain competitive in the market. Aware of its importance, this paper intends to focus on information security. Technologies such as Firewall, Antivirus and Intrusion Detection Systems are designed to help protect companies from attacks and data leakage through the network, functioning somewhat like a virtual guard dog that watches the network and specific hosts in search of suspicious behavior.

Among a number of mechanisms related to information security is the firewall that is responsible for filtering incoming and outgoing packet networks, a fundamental piece that acts as a gateway to many problems.

The security cameras watch over who enters and who exits a location. Why not use the same trick to check what is going in and out of a network in order to evaluate the safety of potential vulnerabilities and discovering the mechanisms that should prevent such access?

The purpose of this work was to make a study of the basic concepts related to the management of risks relating to failures of safety devices, a basic study of firewall systems and intrusion detection systems. Then tests were performed with the Snort IDS and other security tools to assess network traffic in order to find potential vulnerabilities in a system firewall.

**Keywords:** Snort, intrusion detection, firewall, vulnerability.

# 1 INTRODUÇÃO

Com o crescimento da internet, o uso de firewalls tem-se tornado imprescindível, sendo utilizado para evitar que os perigos vindos da internet espalhem-se na rede interna. Mas como ter certeza de que um sistema de firewall está funcionando perfeitamente, e que está cumprindo todos os seus objetivos?

Os firewalls e filtros de pacotes possuem capacidades limitadas para examinar o trafego, normalmente não examinam o conteúdo da carga do pacote. Por outro lado os sensores IDS são projetados para examinar o conteúdo dos pacotes (NORTHCUTT, 2002).

Um bom sistema de detecção de intrusão pode ter um impacto positivo sobre a segurança geral da rede. O foco da detecção de intrusão é identificar ataques e incidentes, mas muito mais pode ser feito, além de sua tarefa principal ele pode ter a função de identificar pontos fracos e vulnerabilidades na segurança, complementando os outros componentes de defesa, pois se você não o fizer certamente os atacantes o farão (NORTHCUTT, 2002).

Um firewall poderia ter um sensor no segmento de rede externo, para identificar toda atividade suspeita e um sensor no segmento de rede interno que possa identificar toda atividade suspeita que passe pelo firewall vindo do exterior, avaliando a eficácia do firewall e também de dentro pra fora verificando se as políticas de segurança estão sendo cumpridas.

A motivação se deu baseada na curiosidade de infância de entender como os administradores de rede visualizavam e entendiam o trafego existente numa rede. Além disso, desde o inicio do curso, me chamava atenção como eram feitos os bloqueios automáticos das estações da UFRGS caso ferissem a política de uso. Tive acesso ao trabalho de Straub (2012) e resolvi estudar sobre sistemas de detecção de intrusão. A abrangência desse trabalho não chega a um IPS nem mesmo tem o mesmo foco, porém a inspiração e motivação partiram dele.

Sendo assim ficou decido fazer um estudo sobre alguns conceitos básicos relacionados à segurança e em seguida investigar a segurança de uma rede de pequeno porte tendo foco na investigação do trafego externo e interno a ela.

## 1.1 Organização

No próximo capítulo, serão abordados conceitos relacionados a gerencia de risco, conceitos de firewall e definição e classificação dos sistemas de detecção de intrusão.

No capítulo três, serão abordados conceitos do Snort, será abordado a sua arquitetura e detalhes sobre as assinaturas.

No capítulo quatro, são apresentados os detalhes do ambiente de testes, instalação do Snort, configuração e resultados obtidos. O capítulo cinco encerra o trabalho, mostrando algumas conclusões.

# 2 SISTEMAS DE DETECÇÃO DE INTRUSÃO

Antes de entender o que é um sistema de detecção de intrusão se faz necessário entender o que é uma propriamente uma intrusão. Uma intrusão consiste no ato de entrar em um lugar sem convite, direito, ou não ser bem vindo. Para nossos propósitos, uma intrusão é simplesmente uma atividade não autorizada em um sistema computacional ou rede, podendo assumir a identidade de um usuário legítimo do sistema e tentar aumentar seus privilégios e conseguir maior acesso ao sistema tentando comprometer um serviço em execução, criar uma conta em um sistema, ou outras atividades semelhantes (BEALE, 2004).

Intrusões podem vir de atividades intencionalmente maliciosas, ou de usuários finais muito ingênuos que clicam em todos os anexos de e-mail enviados a eles, apesar de repetidas advertências para não fazê-lo. Intrusões podem vir de um desconhecido total a três continentes de distância, a partir de um ex-empregado descontente do outro lado da cidade, ou a partir de sua própria equipe de confiança interna.

Detectar atividades maliciosas quando se trata de seus próprios funcionários ou usuários é um dos objetivos mais importantes para um IDS em diversos ambientes. Na verdade, um IDS corretamente configurado, analisado por alguém diferente dos próprios administradores de sistema pode ser um dos poucos métodos capazes de pegar um administrador do sistema praticando algo ilícito. Esta é uma das principais razões pelas quais você deve ter o pessoal de segurança de rede analisando eventos IDS e administradores de sistemas que gerenciando sistemas.

## 2.1 Conceitos básicos de gerencia de riscos

Estar vulnerável às ameaças é um risco, uma tentativa de intrusão numa rede é uma ameaça, portanto, antes de entrar em conceitos mais específicos relacionados à detecção de intrusão se faz necessário esclarecer alguns conceitos básicos que levam os gerentes de TI a tomar as decisões acerca de aumento de nível de segurança de suas instalações.

Entender e diferenciar algumas definições de gerencia de risco está intimamente ligado à decisão da instalação de um sistema de prevenção ou detecção de intrusão.

Em diferentes contextos são tomadas diferentes decisões, roubo de informações de uma grande empresa, em alguns casos, podem significar a falência dela, por outro lado a invasão de um servidor web que hospeda apenas uma pagina web simples com apenas informações de contato de uma pequena empresa pode ter uma impacto insignificante caso possua um backup dessas informações. Portanto contextualizar a diferença do risco

de intrusão em diferentes situações é um fator chave na decisão da necessidade de um sistema IDS.

### 2.1.1 Ativos

Um ativo é algo que tem valor para a organização e que, portanto, requer proteção. Para a identificação dos ativos convém que se tenha em mente que um sistema de informação compreende mais do que hardware e software, englobando pessoas, instalações, etc. (Bezerra, 2011).

Por que precisamos prevenir as intrusões? Porque possuímos algo de valor para ser protegido, possuímos elementos que possuem valor para a organização. Eles elementos são chamados de ativos.

### 2.1.2 Vulnerabilidade

É qualquer fraqueza de um ou mais ativos que possa ser explorada para comprometer a segurança de sistemas de informações. Fragilidade de um ativo ou grupo de ativos que pode se explorada por uma ou mais ameaças (Bezerra, 2011).

No contexto da proposta desse trabalho um firewall mal configurado é um ótimo exemplo de vulnerabilidade.

As vulnerabilidades por si só não causam incidentes, elas são apenas brechas possíveis de exploração por parte das ameaças.

### 2.1.3 Ameaça

Ameaça é qualquer evento que possa prejudicar, total ou parcialmente, as chances de sucesso do projeto, isto é, as chances do projeto realizar o que foi proposto dentro do prazo e fluxo de caixa que foram estabelecidos (ALENCAR, 2005). Por outro lado, Bezerra (2011) define como sendo agentes ou condições que exploram as vulnerabilidades, causando incidentes aos ativos e comprometendo e gerando danos ao negócio da organização.

Exemplos:

- Erros humanos;
- Falhas de hardware;
- Falhas de software;
- Ações da natureza;
- Terrorismo;
- Vandalismo; entre outras.

Uma tentativa de usuários mal intencionados causarem danos a ativos valiosos, normalmente tentando explorar uma ou mais vulnerabilidades é um bom exemplo de uma ameaça no contexto de segurança de redes. O dano pode incluir roubo de informações, sabotagem, destruição de informações, espionagem, ou adulteração.

É comum que os especialistas em segurança acabem se preocupando apenas com ameaças virtuais, entretanto é necessário que se considere todos os tipos de ameaças que possam vir a compreender o negocio de uma empresa.

### 2.1.4 Risco

Combinação da probabilidade de um evento indesejado ocorrer e de suas consequências para a organização. Risco é a possibilidade de uma determinada ameaça explorar vulnerabilidade de um ativo ou de um conjunto de ativos, assim prejudicando a organização (BEZERRA, 2011).

"Risco é a probabilidade de que um fator de risco venha a assumir um valor que possa prejudicar, total ou parcialmente, as chances de sucesso de um projeto" (ALENCAR, 2005, p. 18).

### 2.1.5 Impacto

"Mudança adversa no nível obtido dos objetivos de negócios. Consequência avaliada dos resultados com a ocorrência de um evento em particular, em que determinada vulnerabilidade foi explorada, uma ameaça ocorreu e o risco se concretizou" (BEZERRA, 2011, p. 4).

Impacto é a consequência do evento de risco que se concretizou.

Risco é o produto da probabilidade de ocorrência de consequências negativas com o impacto dessas consequências. Portanto em alguns casos a probabilidade de ocorrência de um terremoto, por exemplo, é bem pequena, mas o impacto pode ser gigantesco.

## 2.2 A importância de um sistema de detecção de intrusão

Sistemas de detecção de intrusão proporcionam um componente de auditoria integrante de um projeto de segurança robusto. Eles permitem que se saiba quando uma rede está sendo escaneada e quando está ocorrendo um ataque, além de fornecer muitas informações que o administrador não conseguiria obter apenas verificando os logs dos servidores e do firewall. É possível ver os ataques que falharam e os ataques que foram bem sucedidos, e obter notificações em tempo real de ataques e tentativas de ataques. Um IDS permite que o administrador assista o tráfego da rede e fique consciente dos erros de configuração, bem como ataques maliciosos muito antes do que seria possível perceber sem a ajuda de um IDS. Eles não são a solução de todos os problemas, e sim, uma ferramenta valiosa nas mãos de um administrador de segurança, oferecendo uma visão sem precedentes sobre o que realmente está acontecendo em sua rede (BEALE, 2004).

Segundo Scarfone (2007) existem outras funções importantes para os sistemas de detecção de intrusão:

- **Identificação de problemas na política de segurança**. Um IDPS pode fornecer algum controle do grau de qualidade para a implementação de políticas de segurança, como a duplicação do conjunto de regras do firewall e alertar quando existir tráfego de rede que deveria ter sido bloqueado pelo firewall, mas não foi por causa de um erro de configuração de firewall.
- **Documentação da existência de ameaças para uma organização**. Um IDPS armazena informações a respeito das ameaças que ele detecta. Entender a frequência e as características dos ataques contra os recursos computacionais de uma organização de computação é útil na identificação de medidas de segurança adequadas para proteger os recursos. A informação

também pode ser usada para mostrar as ameaças que a organização enfrenta à gerência.

- **Dissuadir as pessoas de violar políticas de segurança**. Se as pessoas estão cientes de que suas ações estão sendo monitoradas pelas tecnologias de IDPS a fim de evitar violações das políticas de segurança, elas podem ser menos propensas a cometer tais violações por causa do risco de descobertas.

## 2.3 Falsos positivos

Os falsos positivos são alertas gerados por um IDS achando ter detectado um ataque válido contra um sistema monitorado, mas o ataque na verdade não é válido. Os falsos positivos podem gerar sérios problemas, porque eles geram muitos alertas que podem esconder um ataque real, e pode fazer com que o administrador da rede perca muito tempo analisando uma ameaça que na verdade nunca ocorreu (SCOTT, 2004). Um falso positivo ocorre quando um IDS gera um alerta em ambos os casos:

O trafego de rede que aparenta ser um ataque, mas não é.

Um ataque real, porém contra um serviço que não existe na rede monitorada.

## 2.4 Falsos negativos

Um falso negativo é um ataque real que foi perdido pelo IDS, e, portanto, não gerou alerta. Um IDS pode perder um ataque, porque o ataque não é reconhecido por ele, porque o IDS está sobrecarregado, ou porque o atacante teve sucesso utilizando um método de para enganá-lo. As implicações disso são óbvias: um ataque que você não está ciente (SCOTT, 2004).

Falsos positivos são informações falsas e inúteis produzidas por um sensor, atrapalham bastante, porém um falso negativo é uma situação bem pior, preocupando bastante, porque geram uma falsa sensação de segurança por darem a impressão de que já estamos vendo todos os potenciais ataques.

Falsos negativos é um problema muito grave em relação à eficácia de um IDS/IPS, é uma grandeza inversamente proporcional à utilidade dele, pois de nada adianta ter um sistema que deveria avisar a ocorrência de um ataque na rede e não avisa. Pior que isso, apenas nos sistemas de prevenção de intrusão que reagem de maneira ativa, bloqueando uma conexão legítima e gerando transtornos tanto para o administrador de rede quanto para o usuário.

Normalmente o preço que se paga por tentar diminuir os falsos negativos é aumentar os falsos positivos, ou seja, na tentativa de detectar todos os reais ataques acabam-se considerando eventos normais como intrusões. Essas grandezas inversamente proporcionais são um fator importantíssimo nas decisões acerca de sistemas de prevenção de intrusão.

## 2.5 Por que um firewall não serve como IDS?

Enquanto alguns equipamentos disponíveis comercialmente afirmam ser um firewall e um IDS, e esta é inclusive uma tendência crescente, a função de um firewall é a filtragem de pacotes, não para alertar sobre o tráfego potencialmente malicioso. Firewalls são projetados principalmente para negar ou permitir o tráfego que acessa a

rede, e não para alertar os administradores de atividade maléfica. Muitos firewalls são apenas filtros de pacotes em nível de rede, permitindo ou negando o tráfego com base exclusivamente no endereço IP de origem, endereço IP de destino, e nas portas. Isso não é nem o começo em relação à complexidade de analise de trafego que um IDS atinge (BEALE, 2004). Uma analogia simples que pode ser usada para diferenciar firewalls de IDS é o fato de que as pessoas não confiam as fechaduras de suas portas para atuar também como câmeras, então por que suas fechaduras em sua rede, os firewalls, poderiam atuar como câmeras, ou seja, IDS?

## 2.6 Segurança em redes de computadores

O que é considerado uma rede segura? Redes não podem ser classificadas simplesmente como seguras e inseguras, pois o termo não é absoluto, cada organização que vai definir o nível de acesso. Se uma organização guarda segredos valiosos precisa impedir que pessoas de foras acessem os computadores da organização, no entanto, outras se concentram em manter a comunicação confidencial, ou seja, definem como segura uma rede em que ninguém diferente do emissor e do receptor pretendido pode interceptar e ler a mensagem.

Segundo Nakamura e Geus (2007) o risco é definido em segurança da informação como a probabilidade de um agente de ameaça explorar uma vulnerabilidade, comprometendo dados da confidencialidade, integridade e disponibilidade. Por outro lado Teixeira (1999) considera que segurança de rede consiste em uma pessoa ou organização tentando contra a rede.

Existe uma serie de considerações que devem ser feitas quanto aos riscos existentes, dentre elas:

- as informações que trafegam estão sujeitas à captura;

- os e-mails podem ser lidos, modificados ou alterados;

- a internet deve ser considerada um ambiente hostil e, portanto, não confidenciável;

- a interação de diferentes ambientes resulta na multiplicação de pontos vulneráveis;

- novas tecnologias significam novas vulnerabilidades;

- a segurança das informações é algo mais complexo do que parece.

Tendo em vista a iminência dos problemas decorrentes dessas vulnerabilidades é preciso que se tenha a disposição varias ferramentas objetivando mitigar essas ameaças.

Segundo Bhardwaj, (2007), é de extrema importância à identificação das ameaças que fazem uso das vulnerabilidades para atingir sistemas de segurança. As ameaças e riscos de segurança estão fortemente interligados, pois ao identificarem-se os riscos, estamos identificando as ameaças.

### 2.6.1 Política de segurança

"A política de segurança é a base para todas as questões relacionadas à proteção da informação, desempenhando um papel importante em todas as organizações" (Nakamura e Geus, 2007 p. 188).

Dentre uma serie de questões que são discutidas nas políticas de segurança existe a política para o firewall, é ela que define as regras de filtragem, que, por sua vez,

definem os serviços a serem fornecidos para os usuários externos e internos. Por exemplo, uma empresa tem um sistema de recursos humanos que só pode ser acessado pela rede interna, se por acaso ele for acessado de outro lugar estará ferindo a política de segurança dessa empresa. Portanto, antes de qualquer análise em uma ferramenta de IDS é importante ficar claro o que pode e o que não pode ser feito em uma rede.

A partir do momento que se determina essa política é possível determinar o que se pode e não se pode fazer numa rede de computadores para posteriormente, coma ajuda de um IDS, verificar se ela esta sendo cumprida.

## 2.7 Firewall

A capacidade de conectar qualquer computador em qualquer lugar a qualquer outro computador no mundo trás muitos benefícios, porém tem seus perigos. É muito divertido para as pessoas navegarem pela Internet quando estão em casa. Para os gerentes de segurança das empresas, trata-se de um pesadelo. Muitas empresas têm grandes quantidades de informações confidenciais on-line — segredos comerciais, planos de desenvolvimento de produtos, estratégias de marketing, análises financeiras etc.

Entre esse mundo externo de muitas informações e o usuário de uma rede local existe o que se chama de firewall. Uma maneira bem interessante de entender do que se trata um firewall é fazer uma analogia com os castelos medievais onde se cavava um fosso profundo em torno do castelo. Esse recurso forçava todos aqueles que quisessem entrar ou sair do castelo a passar por uma única ponte levadiça, onde poderiam ser revistados por guardas. Nas redes, é possível usar o mesmo artifício: uma empresa pode ter muitas LANs conectadas de forma arbitrária, mas todo o tráfego de saída ou de entrada da empresa é feito através de uma única passagem que chamamos de firewall (TANENBAUM, 2003).

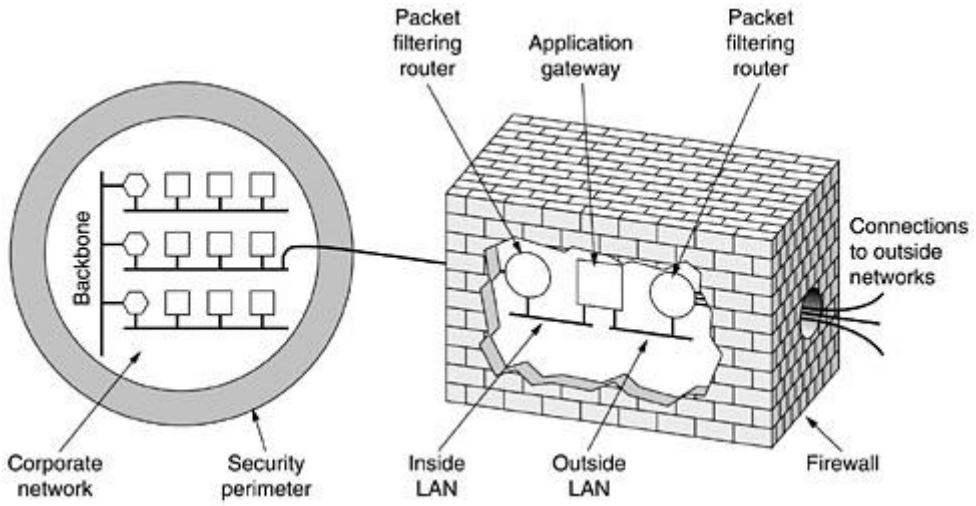


Figura 2.1: Firewall (TANENBAUM, 2003).

Dessa forma, percebe-se a importância que esse portão de entrada tem nas redes de computadores, é através dele que tudo acontece. Levando tudo isso em consideração percebe-se a importância de tal mecanismo de proteção, o qual deve ser muito bem configurado e testado.

Por outro lado Nakamura e Geus (2007) define firewall como uma ponte entre duas ou mais redes, no qual circula todo o trafego. A partir desse único ponto, é possível controlar e autenticar o tráfego alem de registrar, por meio de logs, todo o trafego da rede, facilitando sua auditoria. Portanto, pode-se dizer que o firewall é um ponto entre duas ou mais redes, que pode ser um componente ou um conjunto de componentes, por onde passa todo o trafego, permitindo que o controle, a autenticação e os registros de todo o tráfego sejam realizados.

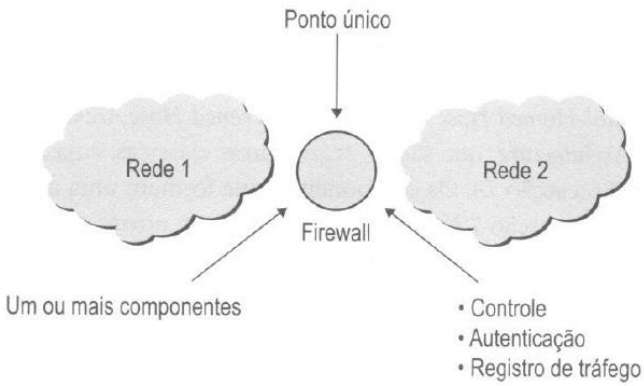


Figura 2.2: Firewall (NAKAMURA e GEUS, 2007).

O firewall é um dos principais, mais conhecido e antigo componente de um sistema de segurança. Sua fama, de certa forma, acaba contribuindo para a criação de uma falsa sensação de segurança, além de causar uma banalização quando à sua definição (NAKAMURA e GEUS, 2007).

Segundo Nakamura e Geus (2007) os firewalls podem ser: filtro de pacotes (static packet filter), proxy (application-level gateway e circuit-level gateway) ou filtro de pacotes baseados em estados (dynamic packet filter, stateful packet filter). Por outro lado, Scrimger (2002) divide nos seguintes tipos: firewalls de filtragem de pacotes, firewall de inspeção de pacote com informações de estado e aplicativos de firewall e de Proxy.

### 2.7.1 O que um firewall não pode fazer

Segundo Komar (2003) um firewall é incapaz de realizar a seguintes tarefas:

Ataques internos – usuários de uma rede local já passaram pelo firewall. O firewall não tem o que fazer quanto ao trafego interno, outras medidas de segurança como configurar restrições de segurança em estações de trabalho e servidores e habilitar auditoria do acesso à internet deve ser implementado para proteção contra esse tipo de ataque.

Engenharia social – é um tipo de ataque em que hackers obtêm informações através dos funcionários, fingindo ser um colega na recepção do hotel, um membro da equipe de segurança, ou apenas alguém da empresa fazendo verificações de rotina. Esta pessoa pede informações privilegiadas, tais como nomes de servidor, endereços IP, ou senhas. Os funcionários devem estar cientes dessas táticas e saber que determinadas informações nunca deve ser passadas.

Vírus e cavalos de troia – Devem ser tomadas precauções para evitar a propagação do vírus e minimizar o dano que ele pode causar. Os cavalos de Tróia são talvez ainda mais difíceis de serem detectados porque não tentam se espalhar para outros arquivos ou

computadores como os vírus. Um bom exemplo do tipo de dano que estes programas podem fazer é um programa cavalo de tróia que envia semanalmente as teclas digitadas em campos de senhas. Um firewall não pode proteger contra essas ameaças.

Administradores de rede mal treinados - o firewall não sabe o que é aceitável e o que não é, a menos que um administrador diga isso a ele. Um administrador de firewall competente deve especificar corretamente qual o tráfego de rede deve ser bloqueado.

O firewall por ser um dos principais, mais conhecidos e antigos componentes de um sistema de segurança acaba contribuindo para a criação de uma falsa sensação expectativa quanto à segurança total da organização. Portanto o firewall por si só não garante a segurança de uma organização (NAKAMURA e GEUS, 2007).

Um firewall não pode proteger uma rede interna de ataques que não passem por ele. A presença de um modem interno à rede é uma grande ameaça, pois não adianta ter todas as regras do firewall funcionando bem se temos outras maneiras de acessar a internet e ser acessado de fora por meio de uma 3G, por exemplo.

### 2.7.2 Filtro de pacotes

Os filtros de pacotes realizam roteamento de maneira seletiva, ou seja, aceitam ou descartam pacotes por meio da analise das informações de seus cabeçalhos. Pouco ou nenhum contexto é mantido, as decisões são tomadas única, e exclusivamente com base no pacote atual. Essa decisão é tomada de acordo com as regras de filtragem definidas na política de segurança da organização (NAKAMURA e GEUS, 2007).

Os firewalls de filtragens de pacotes decidem quando negar ou permitir um pacote com base nas seguintes informações:

- Endereço de origem;
- Endereço de destino;
- Porta de origem e de destino.


Figura 2.3: Campos do cabeçalho IP usados pelo firewall (NAKAMURA e GEUS, 2007).

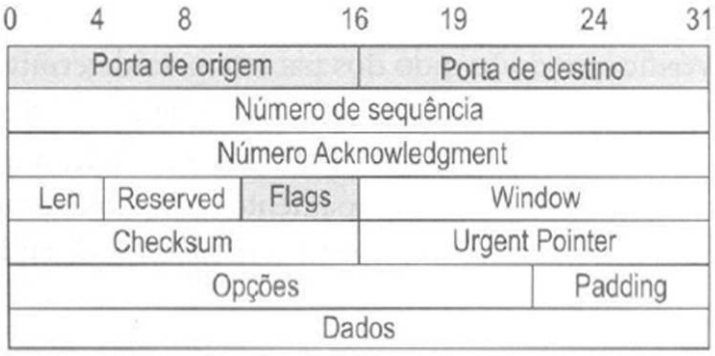


Figura 2.4: Campos do cabeçalho TCP usados pelo firewall (NAKAMURA e GEUS, 2007).

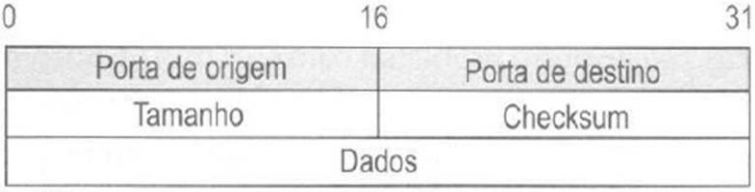


Figura 2.5: Campos do cabeçalho UDP usados pelo firewall (NAKAMURA e GEUS, 2007).

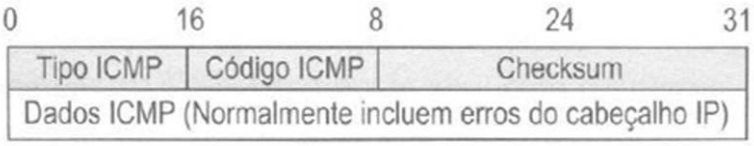


Figura 2.6: Campos do cabeçalho ICMP usados pelo firewall (NAKAMURA e GEUS, 2007).

Um roteador que opere com essas informações pertence à categoria de firewall de filtragem de pacotes tradicional. O recurso que distingue os firewalls de filtragem de pacotes é que eles roteiam o trafego diretamente por eles. Portanto, um pacote precisa pertencer ao bloco de endereços IP autorizado para o qual o firewall está configurado ou ele deve utilizar o endereço IP da rede privada que o firewall esta protegendo (SCRIMGER et al, 2002).

O filtro de pacotes não se encarrega de examinar nenhum protocolo de nível superior ao nível de transporte, como por exemplo, o nível de aplicação que fica como tarefa dos gateways de aplicação. Portanto, qualquer falha de segurança em nível de aplicação não pode ser evitada utilizando somente com um filtro de pacotes (OLIVEIRA, 2003).

Segundo Nakamura e Geus (2007) e Scrimger (2002) as vantagens na utilização de firewalls de filtragem de pacotes:

- São rápidos porque operam apenas nos endereços IP e nos numero de porta do TCP e ignoram o conteúdo dos pacotes, possuindo um alto desempenho e baixo overhead;

- São independentes de aplicativos porque ignoram o conteúdo de dados dos pacotes;
- São menos caros de todos os tipos de firewalls;
- Não requerem nenhuma alteração de configuração seja feita nos computadores que protegem.

Desvantagens:

- São os menos seguros entre os tipos de firewalls, uma vez que podem realizar apenas um exame limitado de trafego entrante;
- Permite a conexão direta para hosts internos de clientes externos
- Ignoram o conteúdo de dados dos pacotes, tornando impossível o bloqueio de usuários que acessem sites da web não autorizados;
- Não podem implementar um firewall complexo.
- Não oferece autenticação do usuário.
- Dificuldade em filtrar serviços que utilizam portas dinâmicas

Na filtragem de pacotes, nenhum tipo de decisão baseada no histórico de pacotes é possível; as regras se aplicam a cada pacote individualmente, por isso os firewall desse tipo também são denominados de stateless, ou seja, sem estado (CARISSIMI, 2009).

Segundo Nakamura e Geus (2007), pelo fato de trabalharem na camada de rede e de transporte faz com que ele seja simples, fácil, barato e flexível de ser implementado. Assim a maioria dos roteadores, que já atuam como gateways, tem também essa capacidade. Isso torna o filtro de pacotes transparente ao usuário, garantindo também um maior desempenho, em comparação aos proxies. Em contrapartida, o filtro de pacotes garante um menor grau de segurança, pois os pacotes podem facilmente se falsificados ou criados especificamente para que passem pelas regras de filtragem definidas. Alem disso, um filtro de pacotes não é capaz de distinguir entre pacotes verdadeiros e falsificados. A capacidade de verificação do sentido dos pacotes para determinar se um pacote vem da rede externa ou interna e sua apropriada configuração é essencial para evitar ataques como o IP spoofing que consiste em mascarar pacotes IP com endereços de remetente falsificados. Outro problema que pode acontecer com filtros de pacotes está relacionado ao tipo de resposta que é enviado a um pedido de conexão que é bloqueado. Dependendo da configuração, a organização pode ser alvo de port scanning que verifica se as portas de hosts específicos da rede estão abertas ou fechadas, finger printing que reúne o máximo de informação possível sobre os sistemas na rede e usa para identificar o hardware e o software utilizado, entre outras técnicas de mapeamento.

### 2.7.3 Filtro de pacotes baseados em estado

Os filtros de pacotes baseados em estado, também conhecidos como filtro de pacotes dinâmicos, tomam as decisões de filtragem tendo como referencia dois elementos:

- As informações dos cabeçalhos dos pacotes de dados, como no filtro de pacotes.
- Uma tabela de estados, que guarda os estados de todas as conexões.

O firewall trabalha verificando somente o primeiro pacote de cada conexão, de acordo com as regras de filtragem. A tabela de conexões que contém informações sobre os estados ganha uma entrada quando o pacote inicial é aceito, e os demais pacotes são filtrados utilizando-se as informações da tabela de estados.

Assim como o filtro de pacotes, o filtro baseado em estados também trabalha na camada de rede da pilha TCP, tendo, portanto, um bom desempenho.

Nesse tipo de firewall o estado das conexões é monitorado constantemente, sendo que a ação do firewall é definida de acordo com o estado das conexões anteriores armazenadas na tabela de estados, permitindo também a segurança das sessões UDP.

Quando uma sessão TCP é iniciada usando um pacote SYN, ele é comparado com as regras do firewall como em um filtro de pacotes, se ele passar por todas as regras sem ser aceito, será descartado e a conexão rejeitada.

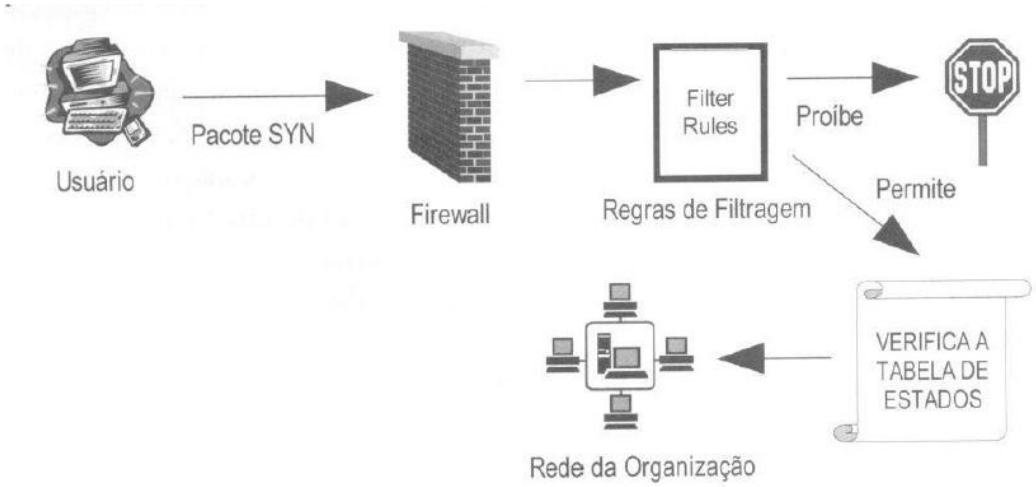


Figura 2.7: Filtro de pacotes baseado em estados trabalhando na chegada de pacotes SYN (NAKAMURA e GEUS, 2007).

Para os demais pacotes, se a sessão estiver na tabela e o pacote fizer parte da sessão, ele será aceito. Entretanto, se os pacotes não fazem parte de nenhuma sessão presente na tabela de estados, eles serão descartados.

O desempenho do sistema melhora, pois apenas os pacotes SYN são comparados com a tabela de regras do filtro de pacotes, e os demais são comparados com a tabela de estados que é bem menor, alem disso a verificação nessa tabela de estados não é feita sequencialmente, como ocorre no filtro de pacotes, mas sim utilizando uma tabela hash (NAKAMURA e GEUS, 2007).

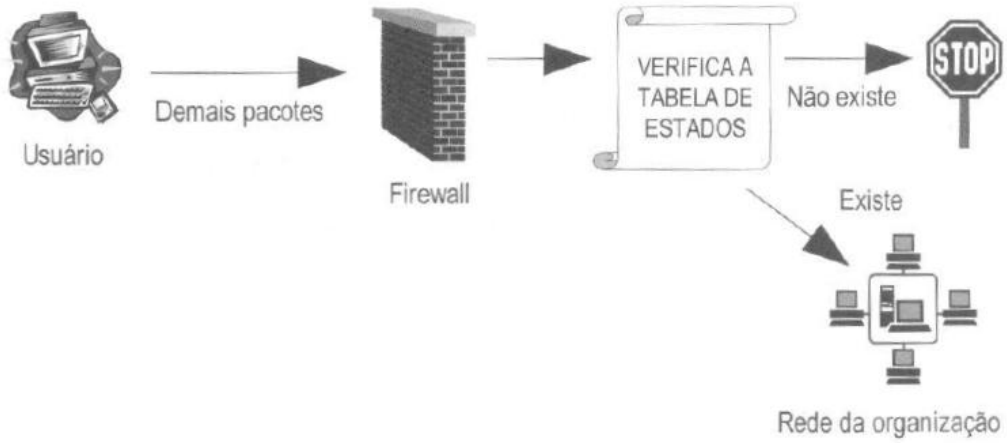


Figura 2.8: Filtro de pacotes baseado em estados trabalhando na chegada dos demais pacotes (NAKAMURA e GEUS, 2007).

Filtros de pacotes baseados em estado são basicamente filtros de pacotes, entretanto com aprimoramentos que visam resolver algumas fraquezas apresentadas no modelo anterior.

### 2.7.4 Gateways de aplicação

Carissimi define: "Os firewalls implementam gateway de aplicação e, devido a sua forma de funcionar, também são denominados de proxies de aplicação ou proxies em nível de aplicação" (CARISSIMI 2009, p. 374).

Nesse caso, ao acessar uma rede externa, tem seu fluxo de informações redirecionado para o gateway de aplicação, ou o contata diretamente, estabelecendo uma conexão. Em seguida, o gateway de aplicação estabelece uma conexão com o servidor esterno o com o qual o cliente deseja conectar-se. A partir desse momento, o gateway de aplicação recebe as requisições do cliente, as analisa e, se elas cumprem a política estabelecida, as encaminham para o servidor destino. As respostas provenientes do servidor são tratadas da mesma forma (CARISSIMI, 2009).

Nos firewalls referidos anteriormente vimos que a filtragem de pacotes permite que seja feita uma filtragem grosseira de conteúdos de cabeçalhos IP e TCP/UDP, incluindo endereços IP, números de porta e bits de reconhecimento. E se a organização quiser oferecer determinados serviços a um conjunto de usuários em vez de endereços de IP? Esse tipo de tarefa vai alem da capacidade de um simples filtro de pacotes. Informações sobre a identidade de usuários internos não estão incluídas nos cabeçalhos IP/TCP/UDP; elas estão nos dados da camada de aplicação.

Para assegurar um nível mais refinado de segurança, os firewalls têm de combinar filtro de pacotes com gateways de aplicação que fazem mais do que examinar cabeçalhos IP/TCP/UDP e tomam decisões com base em dados da aplicação. Vários gateways de aplicação podem executar no mesmo host hospedeiro, mas cada gateway é um servidor separado, com seus próprios processos.

Gateways de aplicação não estão isentos de desvantagens. Em primeiro lugar é preciso um gateway de aplicação para cada aplicação diferente. Em segundo lugar, há um preço a pagar em termos de desempenho, visto que todos os dados serão repassados por meio do gateway. Isso se torna uma preocupação particularmente quando vários usuários ou aplicações estão utilizando o mesmo gateway. Por fim, o software cliente

deve saber como entrar em contado com o gateway quando o usuário fizer uma solicitação, e deve saber como dizer ao gateway de aplicação a qual servidor se conectar (KUROSE, 2010).

O proxy pode trabalhar tanto na camada de sessão ou de transporte (circuit level gateway) quanto na camada de aplicação (aplication level gateway), o que lhe dá mais controle sobre a interação entre o cliente e o servidor externo (NAKAMURA e GEUS, 2007).

A conexão direta entre um usuário interno e o servidor externo não é permitida e o endereço do host interno, garantindo assim uma maior segurança da rede interna da organização.

Uma das grandes vantagens dos proxies é a possibilidade de registrar todo o trafego, seja ele com origem interna ou externa, podendo assim ativar um sistema de alarme quando um tráfego não apropriado estiver em andamento. Alguns proxies podem realizar cache, fato comum em proxies HTTP.

As diferenças entre circuit-level gateway e o application-level gateway estão, além da camada TCP em que atuam também no mecanismo de segurança utilizado. O primeiro funciona como retransmissor entre o cliente e o servidor externo, porém sem realizar a verificação dos serviços. Isso pode causar um problema de segurança: se outro serviço utilizar a porta 80, que é o padrão HTTP, o circuit-level gateway não saberá diferenciar esse pacotes, permitindo que eles passem pelo proxy. Já o application-level gateway, ao trabalhar na camada de aplicação, permite que o payload dos pacotes seja filtrado, como é o caso das filtragens que ocorrem em tags HTML feitas pelo proxy HTTP (NAKAMURA e GEUS, 2007).

Os firewalls gateway de aplicação atuam de forma mais ativa sobre as conexões TCP e no protocolo de aplicação. Inicialmente com um firewall de gateway de aplicação são criadas duas conexões TCP, uma entre o cliente e o firewall e outra entre o firewall e o servidor, em vez de uma só, como ocorre em gateways de circuito. Após isso, o gateway analisa as PDUs do protocolo aplicativo aplicando regras de filtragem, verificando sua validade e sua correta formação (CARISSIMI, 2009).

As vantagens do Proxy são:

- Não permite conexões diretas entre hosts internos e hosts externos;
- Aceita autenticação do usuário;
- Analisa comandos da aplicação no payload dos pacotes de dados, ao contrario do filtro de pacotes;
- Permite criar logs do trafego e de atividades especificas;

As desvantagens do Proxy são:

- É mais lento do que os filtros de pacotes, pois há a necessidade de armazenar tabelas, consumindo memória e realizar processamento;
- Requer um proxy específicos para cada aplicação;
- Não trata pacotes ICMP;
- Não aceita todos os serviços.

## 2.8 Tipos de Sistemas de detecção de intrusão

Existem vários tipos de IDSs disponíveis atualmente, caracterizados por diferentes abordagens de monitoração e análise. Cada abordagem tem suas vantagens e desvantagens. Muitos autores classificam de forma diferente, Bace (2001) classifica em três componentes funcionais fundamentais, baseado na fonte da informação para obter os dados, na maneira como as informações são analisadas, e na maneira como o IDS reage às intrusões.

### 2.8.1 Classificação quanto à fonte de informação

A forma mais comum de classificar IDSs é agrupá-los de acordo com a fonte de informação usada. Alguns IDSs analisam os pacotes de rede capturados em backbones ou segmentos LAN. Por outro lado, alguns sistemas de detecção de intrusão analisam as informações geradas pelo sistema operacional ou alguns software em busca de indícios de intrusão (BACE, 2001).

*2.8.1.1Sistema de detecção de intrusão baseado em rede*

Segundo Rehman (2003) um NIDS nada mais é do que um farejador mais inteligente que além de observar o trafego de rede, é capaz de reconhecer padrões, baseados em regras, reportar e alertar quando um padrão é reconhecido.

Os sensores de um sistema NIDS na maioria das vezes são placas de rede trabalhando no modo promíscuo, ou seja, ela não captura apenas os pacotes destinados ao seu endereço MAC ela captura todos os pacotes que passam pelo segmento de rede em questão.

No modo promíscuo, o NIDS pode escutar todas as comunicações no segmento de rede, no entanto, isso não é tudo o que é necessário para garantir que seus NIDS sejam capazes de escutar todo o tráfego da sub-rede. Se o dispositivo de intercomunicação for um hub, todos os pacotes são automaticamente enviados para todas as portas, facilitando, assim, o monitoramento. No entanto, se esse dispositivo for um switch, é necessária a utilização de uma porta especial para monitoramento para que possa receber os pacotes de todas as demais portas. Depois de configurar seus NIDS, é recomendável executar uma ferramenta de varredura na interface, para garantir que o sensor poderá receber todo o tráfego na sub-rede (BEALE, 2004).

A vantagem de um NIDS é que ele não tem qualquer impacto sobre os sistemas ou redes que está monitorando. Não adiciona qualquer carga para os anfitriões, e um atacante que comprometa um dos sistemas que estão sendo monitoradas não terá, necessariamente, acesso ao NIDS e nem mesmo sabe de sua existência, ou seja, é um sistema totalmente transparente ao atacante.

Em vista das leis de privacidade emergentes, monitorando as comunicações de rede é uma responsabilidade que deve ser considerada com cuidado. Devem-se verificar os requisitos legais locais para tal atividade.

Segundo Beale (2004), se o tráfego é criptografado, ainda é possível ver os cabeçalhos IP e os cabeçalhos dos protocolos da camada de transporte, mas não será possível decodificar o conteúdo do pacote, sem quebrar a criptografia. Você pode ver quanto tráfego é gerado, origem e destino, mas não será possível ver o que eles estão dizendo, portanto, tem essa desvantagem.

A Figura abaixo mostra uma rede com três sensores NIDS estrategicamente posicionados, protegendo os servidores que são visíveis pela rede externa e também as estações de trabalho da rede interna. Quando um servidor público está comprometido em uma sub-rede filtrada, o servidor pode se tornar uma plataforma de lançamento para explorações adicionais. Portanto, monitorações cuidadosas são necessárias para evitar danos futuros (BEALE, 2004).

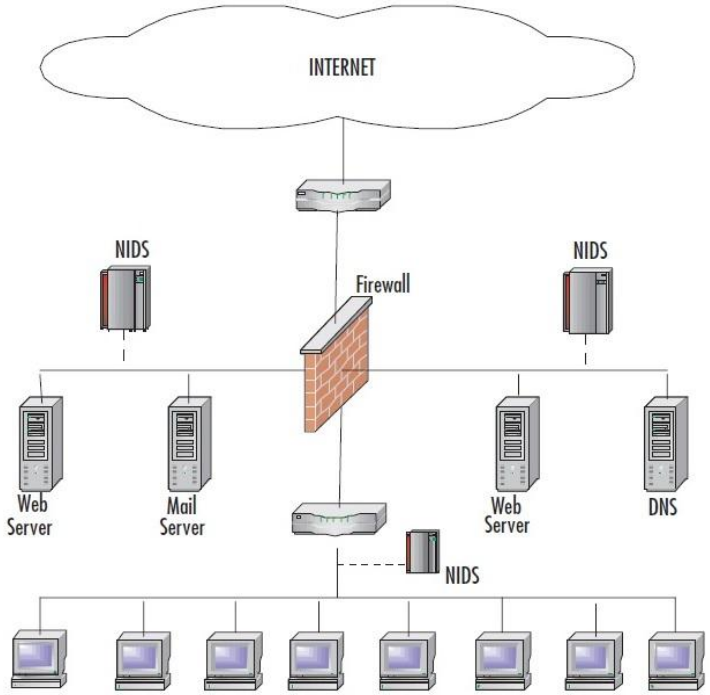


Figura 2.9: Sistema de detecção baseado em rede (BEALE, 2004).

*2.8.1.2 Sistema de detecção de intrusão baseado em host*

Os sistemas baseados em host diferem dos baseados em rede de duas maneiras. Em primeiro lugar, um HIDS monitora somente o tráfego de rede do host, a integridade dos arquivos do sistema operacional e o registro dos eventos, em segundo lugar, a placa de rede de um sistema com um HIDS instalado opera normalmente, ou seja, não está no modo promiscuo, o que pode ser uma vantagem em alguns casos, pois pode gerar um uso intensivo da CPU e deixar a máquina lenta (SCOTT, 2004).

Outra vantagem do HIDS é a capacidade para adaptar o conjunto de regras para ser muito específico para algum sistema em particular. Por exemplo, não há necessidade de configurar múltiplas regras designadas para detectar Network File System (NFS) em um host que não utiliza NFS. Sendo assim, ter a capacidade de refinar as regras vai melhorar o desempenho e diminuir os falsos positivos.

A principal vantagem de um HIDS, no entanto, reside na sua capacidade de detectar alterações específicas para os arquivos do sistema operacional de seu hospedeiro. Pode monitorar o tamanho dos arquivos e checksums para garantir que os arquivos essenciais do sistema não são modificados de forma maliciosa, sem alguém perceber. Ele pode interceptar chamadas de sistema desonestas que podem ser uma tentativa de explorar uma vulnerabilidade local. Além disso, pode vigiar o tráfego dentro de um sistema que nunca atravessa a rede e, portanto, nunca seria visto pelos NIDS (BEALE, 2004).

Os HIDS podem ter suas regras bem específicas para cada estação não precisando ter regras mais genéricas como os sistemas baseados em redes.

Há algumas dificuldades na escolha de um HIDS, dentre elas pode-se citar a escolha daquele que seja o mais adaptado ao sistema operacional em questão. Se a rede a ser monitorada tiver uma grande diversidade de sistemas operacionais, pode ser difícil a utilização de um mesmo fornecedor para todas as suas HIDSs, assim, você será obrigado e levar isso em consideração na escolha. Conheça as funcionalidades que você quer em seus HIDS, e certifique-se que os HIDS que você selecionar vai apoiar essas características em todas as plataformas você precisa.

Uma desvantagem é o fato de que o HIDS irá consumir recursos do host que está monitorando.

A manutenção de uma rede com um grande número de hosts é uma grande desvantagem e um desafio, pois se o sistema não tiver uma gestão centralizada, o trabalho para gerenciar os sistemas individuais pode inviabilizar a adoção de tais ferramentas em sistemas em grande escala.

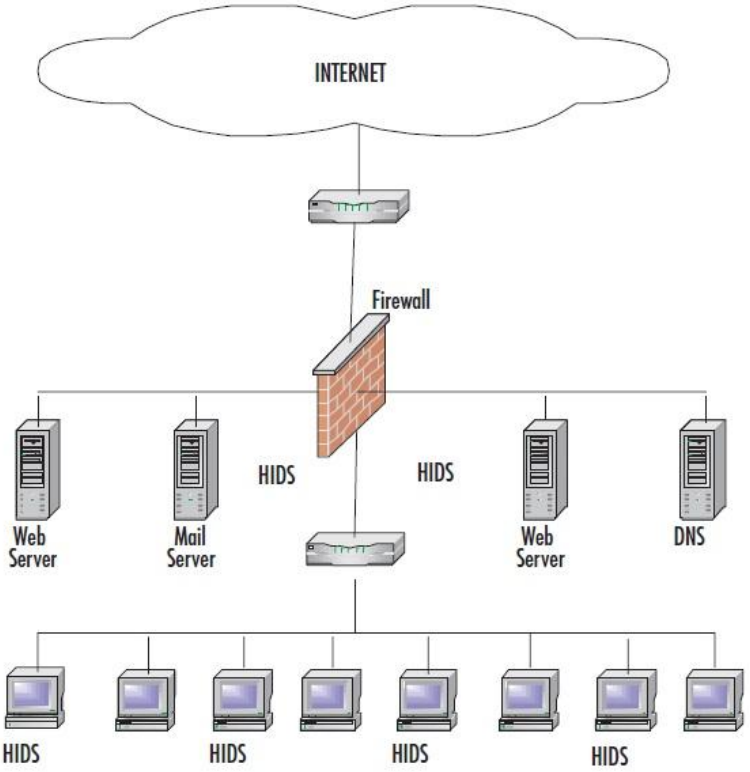


Figura 2.10: sistema detecção intrusão baseado em host (BEALE, 2004).

### 2.8.2 Classificação quanto à análise

Existem duas abordagens principais relativas à detecção por analise de eventos: detecção baseada em assinaturas e detecção de anomalias.

#### *2.8.2.1 Detecção baseada em assinaturas*

A assinatura é um padrão que corresponde a uma ameaça conhecida. Detecção baseada em assinaturas é o processo de comparação de assinaturas com eventos observados para identificar possíveis incidentes.

Detecção baseada em assinaturas é muito eficaz na detecção de ameaças conhecidas, porém muito limitada na detecção de ameaças desconhecidas, ameaças disfarçadas através do uso de técnicas de evasão, e variantes de ameaças conhecidas.

Detecção baseada em assinaturas é o método mais simples de detecção porque apenas compara a unidade atual de atividade, tal como um pacote ou uma entrada de log, com uma lista de assinaturas usando operações de comparação de string. As tecnologias de detecção baseadas em assinaturas têm pouca compreensão de muitos protocolos de aplicação e de rede e não conseguem compreender o estado de comunicações complexas. Por exemplo, eles não podem emparelhar um pedido com a resposta correspondente, como saber que uma solicitação para um servidor Web gerou um código de erro 403 para uma página específica, o que significa que o servidor se recusou a preencher o pedido. Eles também não têm a capacidade de lembrar os pedidos anteriores no momento do processamento do pedido atual, essa limitação impede os métodos de detecção baseados em assinatura de detectar ataques compostos de vários eventos se nenhum deles tiver uma clara indicação de ataques (SCARFONE, 2007).

Embora encontrado em vários contextos como tendo o mesmo sentido, regras e assinaturas são coisas diferentes. O termo assinatura refere-se a nada mais do que uma definição básica de ataque, sinais distintivos ou características presentes num ataque, é um padrão que corresponde a uma ameaça conhecida como se fosse a impressão digital. Uma regra define a metodologia de ataque em termos de identificação do invasor, ela objetiva detectar uma vulnerabilidade real, não apenas um exploit ou dados específicos. Desenvolver uma regra requer um profundo conhecimento de como a vulnerabilidade realmente funciona (SNORT, 2012).

*2.8.2.2 Detecção por anomalia*

Detectores de anomalias identificam comportamento anormal em uma estação ou rede. Eles funcionam no pressuposto de que os ataques são diferentes de uma atividade normal, e podem, portanto, serem detectados por sistemas que identificam estas diferenças. Detectores de anomalias constroem perfis representando o comportamento normal dos usuários, estações, ou conexões de rede. Estes perfis são construídos a partir de dados históricos recolhidos ao longo de um período de funcionamento normal. Os detectores recolhem dados do evento e usam uma variedade de medidas para determinar quando a atividade monitorada desvia do normal (BACE, 2001).

As medidas e técnicas utilizadas na detecção de anomalias incluem:

- Detecção por limiar, em que certos atributos do comportamento do usuário e do sistema são expressos em termos de contagens, com algum nível estabelecido como admissível. Tais atributos de comportamento podem incluir o número de arquivos acessados por um determinado usuário em um período de tempo, o número de tentativas fracassadas de fazer o login no sistema, a quantidade de CPU utilizada por um processo, entre outras. Este nível pode ser estático ou heurístico, ou seja, pode mudar ao longo do tempo.
- Medidas estatísticas, tanto paramétricas, onde a distribuição dos atributos perfilados é assumida para se ajustar a um determinado padrão, quanto não paramétricas em que a distribuição dos atributos perfilados é "aprendido" a partir de um conjunto de valores históricos, observada ao longo do tempo.
- Medidas baseadas em regras são semelhantes às medidas estatísticas não paramétricas em que os dados observados definem padrões de uso aceitáveis,

mas difere pelo fato de esses padrões serem especificados como regras, e não quantidades numéricas.

- Outras medidas, incluindo redes neurais, algoritmos genéticos, e modelos baseados no sistema imunológico.

Infelizmente, os IDSs baseados na detecção por anomalias produzem, frequentemente, um grande número de falsos positivos, visto que os padrões normais de comportamento do usuário e do sistema podem variar muito. Apesar deste inconveniente, os pesquisadores afirmam que os IDSs baseados em anomalia são capazes de detectar novas formas de ataque, ao contrário dos baseados em assinaturas que dependem de correspondência de padrões de ataques anteriores (BACE, 2001).

Além disso, algumas formas de anomalia produzem saídas que podem, por sua vez, serem utilizadas como fonte de informação para outros métodos de detecção como o por anomalias. Por exemplo, um IDS baseado em anomalias pode gerar um perfil que representa o número normal de arquivos acessados por um usurário em particular, o detector baseado em assinaturas pode usar esse perfil como parte de uma assinatura de detecção (BACE, 2001).

Vantagens:

- IDSs baseados na detecção de anomalias detectam comportamento anormal e, portanto, têm a capacidade de detectar sintomas de ataques sem conhecimento específico dos detalhes.
- Os detectores de anomalias podem produzir informações que podem, por sua vez, serem utilizadas para definir assinaturas.

Desvantagens:

- IDSs baseados na detecção de produzem grande número de falsos positivos devido ao comportamento imprevisível dos usuários e redes.
- Exigem extensos treinamentos a fim de caracterizar os padrões normais de comportamento dos registros de eventos do sistema.

### 2.8.3 Classificação quanto à resposta

Uma vez que os sistemas de detecção de intrusão tenham obtido informações sobre o evento a fim de encontrar sintomas de ataques, eles geram respostas. Algumas destas respostas envolvem resultados de relatórios em um local pré-determinado. Outros envolvem respostas automatizadas mais ativas. Embora os pesquisadores sejam tentados a subestimar a importância das funções de resposta dos IDSs, elas são realmente muito importantes. IDSs comerciais suportam uma ampla gama de opções, muitas vezes classificados como respostas ativas, respostas passivas, ou uma mistura das duas.

#### *2.8.3.1 Resposta passiva*

Sistemas de detecção de intrusão que possuem respostas passivas fornecem dados aos usuários do sistema, contando com os seres humanos para tomar ações subsequentes a partir dessas informações. Muitos IDSs comerciais dependem exclusivamente de respostas passivas (BACE, 2001).

Alarmes e notificações são gerados por IDSs para informar aos usuários quando os ataques são detectados. A maioria dos IDSs permite que os usuários tenham uma grande

liberdade para determinar como e quando os alarmes são gerados e para quem eles são exibidos.

A forma mais comum de alarme é uma janela pop-up na tela ou uma mensagem no console do IDS ou em outros sistemas como especificado pelo usuário durante a configuração do IDS. As informações fornecidas na mensagem de alarme variam desde uma simples notificação de que uma intrusão ocorreu até mensagens mais detalhadas descrevendo os endereços IP de origem e de destino, portas etc. Também é possível configurar o sistema para que envie mensagem para telefones celulares, e-mail, entre outras (BACE, 2001).

Portanto, os sistemas com resposta passiva se restringem a gerar alertas e esperar que seja tomada alguma atitude pela equipe de resposta a incidentes de maneira manual.

*2.8.3.2 Resposta ativa*

Respostas ativas de IDSs são ações automáticas tomadas quando certos tipos de intrusões são detectadas. Há três categorias de respostas ativas.

Coletar informações adicionais:

A atitude mais natural é procurar descobrir informações extras sobre a suspeita de ataque. No caso do IDS, isso pode envolver aumentar o nível de sensibilidade de fontes de informação, por exemplo, aumentar o número de eventos registrados no log do sistema operacional, ou aumentar a sensibilidade de um monitor de rede para capturar todos os pacotes, não apenas aqueles visando uma porta específica ou sistema de destino. Coleta de informações adicionais pode auxiliar o sistema a diagnosticar se um ataque obteve êxito ou não. Esta opção também permite que a organização reúna informações que possam ser utilizadas para apoiar uma investigação e descoberta do atacante, e dar sustentação a medidas legais (BACE, 2001).

Alterar o Ambiente:

Outra resposta ativa é parar um ataque em andamento e, em seguida, bloquear um futuro acesso do atacante. Normalmente, IDSs não têm a capacidade de bloquear o acesso de uma pessoa específica, mas sim bloquear o IP a partir do qual o atacante parece estar chegando. É muito difícil bloquear um atacante determinado e experiente, mas muitas vezes é possível impedir atacantes experientes ou novatos, tendo as seguintes ações (BACE, 2001):

- Injetando pacotes TCP reset na conexão do atacante para o sistema da vítima, terminando, assim, a conexão;
- Reconfigurando roteadores e firewalls para bloquear pacotes que aparentemente tenham origem do endereço IP do atacante;
- Reconfigurar os roteadores e firewalls para bloquear as portas de rede, protocolos ou serviços que estão sendo usadas por um atacante;
- Em situações extremas, reconfigurar os roteadores e firewalls para cortar todas as conexões que utilizam determinadas interfaces de rede.

Tomar medidas contra o atacante:

Há também os que acreditam numa medida em que a resposta ativa é tomar medidas contra o intruso. A forma mais agressiva da resposta envolve o lançamento de ataques de volta ou tentando obter informações sobre o exército invasor ou seu local de origem.

Por mais tentador que seja essa atitude não é aconselhada. Devido a ambiguidades legais, esta opção pode representar um risco maior do que o ataque que se pretende bloquear.

A primeira razão para esta opção ser usada com uma grande dose de cautela é que ele pode ser ilegal. Além disso, como muitos atacantes usam endereços falsos de rede ao atacar sistemas, ele carrega consigo um alto risco de causar danos a usuários inocentes. Finalmente, de volta greve pode escalar o ataque, provocando um atacante que originalmente destinado apenas para procurar um local para tomar medidas mais agressivas. Recomenda-se fortemente um aconselhamento legal antes de tomar qualquer atitude agressiva como essa (BACE, 2001).

# 3 SNORT

Snort é um IDS de código aberto de domínio publico com centenas de milhares de implementações existentes. É um software multiplataforma que pode ser executado em Linux, Unix e Windows. Ele usa uma interface Libpcap de analise genérica, que também é usada em varias ferramentas de análise de pacotes como, por exemplo, o Wireshark. O Snort pode facilmente trabalhar com 100Mbps de trafego, velocidades superiores é recomendado o uso de múltiplos sensores. Ele é sniffer que tem como diferencial a capacidade de inspecionar o payload do pacote, área que contém os dados do mesmo, fazendo os registros dos pacotes, além de detectar as invasões (KUROSE, 2010).

Sua principal característica é a capacidade de inspecionar a área de dados dos pacotes e comparar com as informações das regras, que, quando detectado, gera um alerta num arquivo de log, que pode ser usado para uma análise manual ou outro procedimento automatizado.

O Snort é um sistema de prevenção e detecção (IDS/IPS) desenvolvido pela Soucefire. Combinando os benefícios de inspeções baseadas em assinatura, protocolo e anomalia, o Snort é a tecnologia de IDS/IPS mais usada no mundo. Com milhões de downloads e cerca de 400.000 usuários registrados, Snort se tornou o padrão IPS (SNORT, 2012).


Figura 3.1: Mascote (www.snort.org)

Segundo ROESCH (2012) o Snort possui três modos de operação, são eles:

- Modo farejador, que simplesmente lê o os pacotes da rede e apresenta-os para o usuário em um fluxo contínuo na tela.
- Modo registrador de pacotes, que registra os pacotes para o disco.
- Sistema de detecção de intrusão baseado em rede (NIDS), o mais complexo e configurável modo que permite ao Snort analisar o tráfego de rede e

comparar com um conjunto de regras e executa várias ações com base naquilo que captura.

## 3.1 Arquitetura do Snort

A arquitetura do Snort é composta de quatro componentes básicos: o decodificador de pacotes, o pré-processador, o mecanismo de detecção e os plugins de saída.

Em sua forma mais básica o Snort é um farejador de pacotes. Entretanto, ele é projetado para pegar pacotes e processá-los através do pré-processador e depois verificar esses pacotes com relação a uma série de regras. A figura abaixo oferece uma visão de alto nível da arquitetura do Snort (REHMAN, 2003).

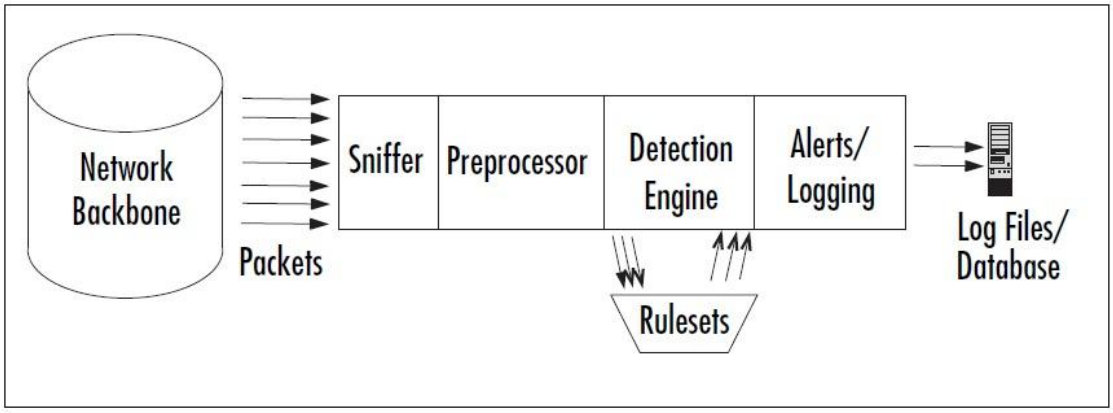


Figura 3.2: Arquitetura do Snort (BAKER, 2007).

### 3.1.1 Decodificador de pacotes

O decodificador obtém os pacotes a partir de diferentes tipos de interfaces de rede e os prepara para serem pré-processados ou para serem enviados para o mecanismo de detecção (REHMAN, 2003).

Os pacotes entram através da interface de rede e são decodificados pelo decodificador de pacotes, que determina qual protocolo está em uso e compara os dados contra o comportamento normalmente permitido para os pacotes desse tipo de protocolo, em outras palavras ele separa a parte de “protocolo” da parte de “payload” do pacote. O próprio decodificador de pacotes pode gerar alertas com base em cabeçalhos de protocolos mal formados, pacotes excessivamente longos, opções TCPs incomuns ou incorretas definidas nos cabeçalhos, entre outros comportamentos semelhantes (BEALE, 2004).

### 3.1.2 Pré-processadores

Segundo Rehman (2003) pré-processadores são componentes que podem ser utilizados com o Snort para organizar ou modificar os pacotes de dados antes que o mecanismo de detecção faça alguma operação para descobrir se o pacote está sendo usado por um invasor. Alguns pré-processadores também realizam a detecção de anomalias encontrando nos cabeçalhos dos pacotes e gerando alertas. Pré-processadores são muito importantes para qualquer IDS para preparar pacotes de dados para serem comparados com as regras do mecanismo de detecção, pois normalmente esses pacotes não estão prontos para detecção por estarem fragmentados. Hackers usam diferentes

técnicas para enganar um IDS de diferentes maneiras. Por exemplo, você pode ter criado uma regra para encontrar uma assinatura "scripts / IISAdmin" em pacotes HTTP. Se você está combinando essa string exatamente, você pode ser facilmente enganado por um hacker que faz pequenas modificações para esta cadeia.

Pré-processadores são também utilizados para a desfragmentação de pacotes. Quando um bloco de dados grande é transferido para um hospedeiro, o pacote é geralmente fragmentado. Por exemplo, o comprimento máximo padrão de qualquer pacote de dados através de uma rede Ethernet é geralmente de 1500 bytes. Este valor é controlado pela unidade de transferência de valor máximo (MTU) para a interface de rede. Isto significa que, ao enviar dados maiores do que 1500 bytes, eles serão divididos em vários pacotes de dados de modo a que cada fragmento de pacotes é menor ou igual a 1500 bytes. Os sistemas de recepção são capazes de voltar a montar esse pequenos pedaços para formar novamente o pacote de dados original. No IDS, antes de aplicar quaisquer regras ou tentar encontrar uma assinatura, você tem que remontar o pacote. Por exemplo, a metade da assinatura pode estar presente em um segmento e a outra metade no outro segmento. Para detectar a assinatura corretamente, você tem que combinar todos os segmentos de pacotes. Hackers usam fragmentação para enganar os sistemas de detecção de intrusão. Os pré-processadores são usados para proteger contra esses ataques. Pré-processadores podem desfragmentar pacotes, decodificar HTTP URI, remontar sessões TCP e assim por diante. Estas funções são uma parte muito importante do sistema de detecção de intrusão (REHMAN, 2003).

Os pré-processadores são muito mais importantes do que pode parecer, pois se a normalização não acontecer de forma correta, de nada adiantarão as regras.

### 3.1.3 Mecanismo de detecção

Como esperado: “The detection engine is the most important part of Snort”. (REHMAN, 2003, p.14) a tarefa mais importante de um IDS é a detecção propriamente dita. Sua responsabilidade é a de detectar se existe alguma atividade de intrusão em um pacote. O mecanismo de detecção utiliza regras Snort para esta finalidade. As regras são lidas em estruturas de dados internas onde elas são comparadas com todos os pacotes já remontados pelos pré-processadores. Se um pacote corresponder a qualquer regra, são tomadas medidas adequadas, caso contrário, o pacote é descartado. Ações apropriadas podem gerar logs ou gerar alertas.

O mecanismo de detecção é a parte crítica em relação ao desempenho do Snort. Dependendo do poder de processamento do servidor e de quantas regras estão definidas para análise, pode levar mais ou menos tempo para responder a diferentes pacotes. Se o tráfego for muito intenso na rede quando o Snort estiver trabalhando no modo NIDS, alguns pacotes podem ser descartados e não será possível obter uma resposta verdadeiramente em tempo real. Segundo Rehman (2003) a carga sobre o mecanismo de detecção depende dos seguintes fatores:

- Número de regras;
- Poder de processamento na maquina onde o Snort está executando;
- Velocidade de barramento interno usado na máquina onde o Snort está executando;
- Carga da rede;

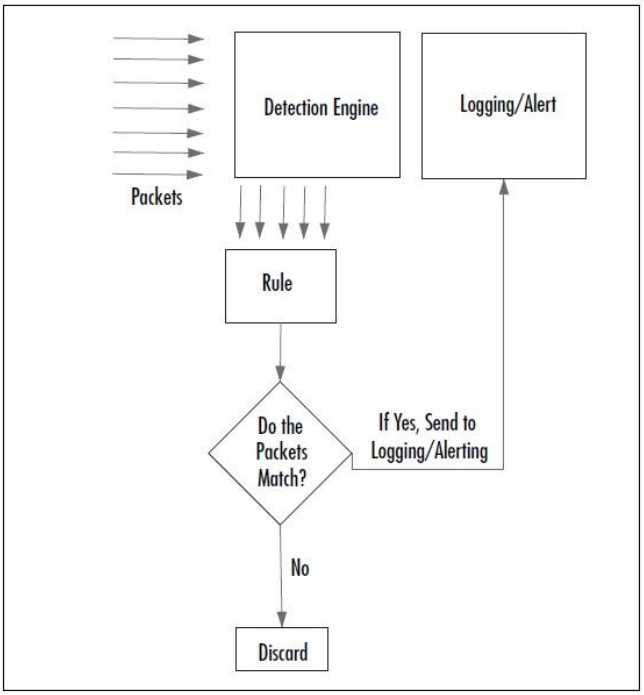


Figura 3.3: Mecanismo de detecção do Snort (BAKER, 2007).

### 3.1.4 Sistema de Registro e alerta

Os plugins de saída são os responsáveis por gerar alertas ou tomar alguma outra medida após a verificação do pacote pelo mecanismo de detecção e a correspondência do pacote com alguma regra. O mecanismo de registro no Snort irá arquivar os pacotes que desencadearam as regras do Snort, enquanto o mecanismo de alerta é usado para notificar o analista que uma regra foi acionada. Dependendo do que o mecanismo de detecção encontra dentro de um pacote, o pacote pode ser utilizado para registrar a atividade ou gerar um alerta.

Os vários tipos de plugins de saída podem ser especificados no arquivo de configuração do snort, onde se pode especificar que componentes de alerta e registro se deseja ativar. Dentre os possíveis módulos podemos destacar os logs no formato tcpdump, banco de dados SQL e o formato binário (ROESCH, 2012).

O plugin de saída do Snort no formato tcpdump armazena os pacotes num arquivo no formato pcap porque este é um formato amplamente aceito e tem permitido uma maior flexibilidade no trabalho com os arquivos de log desse tipo. Existe uma grande quantidade de softwares disponíveis para examinar arquivos no formatado pcap (BAKER, 2007).

O plugin de saída com capacidade para gravar logs em um banco de dados permite que as informações sejam armazenadas e visualizadas em tempo real. Por outro lado, isso pode fazer com que o Snort falhe em sua tarefa prioritária de detectar intrusões por estar utilizando recursos na inserção num banco de dados. Armazenar os logs do Snort dentro de um banco de dados relacional é muito mais eficiente do que armazená-los em arquivos simples. Eles serão muito mais gerenciáveis desta forma. Várias ferramentas estão disponíveis para a extração e formatação das informações obtidas a partir dos registros do banco de dados do Snort (BAKER, 2007).

"Unified logs are the future of Snort reporting, logging, and output" (BAKER, 2007, p.367). A finalidade desse tipo de log é aumentar a velocidade e eficiência no armazenamento de logs, diminuindo a quantidade de recursos utilizados em atividades que não são nas funções de captura e análise, aumentando a probabilidade de que os pacotes não sejam descartados. A saída do log é em formato binário. Muitos administradores preferem este método de registro, porque é aceitável para uso com as ferramentas de relatórios mais populares do Snort: Barnyard e Cerebus (BAKER, 2007).

## 3.2 Regras do Snort

As regras do Snort constituem a parte mais importante relativa ao sucesso ou não da detecção das intrusões. Saber cria-las é uma ferramenta poderosa para o administrador de redes customizar o funcionamento do IDS a fim de atender as suas necessidades. O Snort tem uma sintaxe bem simples para a construção de regras o que facilita tal tarefa (REHMAN, 2003).

A velocidade com que novas pragas virtuais se alastram é algo que é de conhecimento dos administradores de rede, portando a facilidade de criar novas regras é uma qualidade muito importante para um IDS.

Devem ser utilizadas apenas as regras que se apliquem à realidade da rede que se esta monitorando. Por exemplo, não é necessário ter regras relativas a ataques a um servidor de banco de dados se a rede não o possuir.

É importante considerar que quanto maior o numero de regras maior será o poder computacional exigido para processar os pacotes, sendo assim, é importante implementar o maior numero possível de assinaturas no menor numero de regras possível.

As regras descrevem uma condição ou estado e as ações a serem tomadas quando essa condição da regra for atendida. Todas as regras do Snort têm duas partes lógicas: cabeçalho e opções. Todos os detalhes sobre as opções que podem se definidas nas regras são encontradas no manual do Snort (BAKER, 2007).

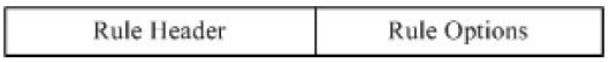


Figura 3.4: Estrutura básica regra Snort (REHMAN, 2003).

### 3.2.1 Cabeçalho das regras

O cabeçalho da regra contém informações sobre quais ações uma regra toma. Ele também contém critérios para verificar a que pacotes a regra se aplica. A parte de opções geralmente contém uma mensagem de alerta e informações sobre qual a parte do pacote deverá ser usada para gerar a mensagem de alerta. A parte de opções contém critérios adicionais para comparar uma regra com os pacotes de dados. Uma regra pode detectar um tipo ou vários tipos de atividades de intrusão. Regras inteligentes devem ser capazes de serem aplicadas a múltiplas assinaturas de intrusão (REHMAN, 2003).

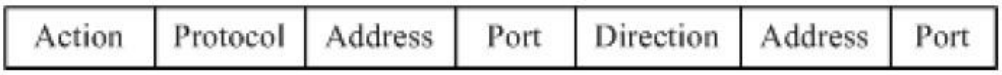


Figura 3.5: Estrutura cabeçalho (REHMAN, 2003).

O campo “Action” da regra determina quais são as medidas tomadas quando os critérios são satisfeitos e uma regra corresponde com o que e encontrado no pacote. Ações típicas são gerar uma mensagem de alerta ou de registro ou invocar outra regra.

O campo do “Protocol” é usado para aplicar a regra a pacotes de um protocolo em particular. Este é o primeiro critério mencionado na regra. Alguns exemplos de protocolos usados são IP, ICMP, UDP, entre outros.

O campo “Address” define endereços de origem e destino que podem ser um único host, múltiplos hosts ou endereços de rede. Também é possível usar esses campos para excluir alguns endereços de uma rede completa. Percebe-se na figura acima que há dois campos de endereço que são determinados como origem e destino com base no campo “Directon”. Por exemplo, se o campo de direção é "->", o endereço do lado esquerdo é a origem e o endereço do lado direito é o destino.

No caso do protocolo TCP ou UDP, os campos “port” determinam a porta de origem e destino de um pacote em que a regra é aplicada. Já no caso de protocolos da camada de rede, como IP e ICMP, os números de porta não tem nenhum significado.

O campo “Direction” da regra determina qual o endereço e porta é usado como origem e qual como destino.

Por exemplo, considere a seguinte a regra abaixo gera uma mensagem de alerta sempre que detecta um pacote de ping ICMP (ICMP Echo Request) com TTL igual a 100.

```
alert icmp any any -> any any (msg: "Ping with TTL=100"; \
   ttl: 100;)
```

### 3.2.2 Opções das regras

O campo de opções das regras vem depois do cabeçalho e é colocado entre parênteses. Pode ser uma opção ou várias e elas são separadas por ponto e vírgula, que, se usadas, formam um “E” lógico. A ação no cabeçalho da regra é acionada somente quando todos os critérios nas opções são verdadeiros. Todas as opções são definidas obrigatoriamente por palavras-chave e podem vir acompanhadas de argumentos separados por dois-pontos (REHMAN, 2003).

Opções das regras formam o coração do sistema de detecção de intrusão, combinando facilidade de uso com poder e flexibilidade.

Existem quatro categorias principais de palavras chaves das opções das regras (SNORT, 2012):

- Geral: Essas opções fornecem informações sobre a regra, mas não têm qualquer efeito durante a detecção;
- Carga: Todas as opções dessa categoria procuram informações dentro da carga do pacote e podem ser inter-relacionadas;
- Não carga: Essas opções procuram informações fora da carga do pacote;

- Pós-detecção: Essas opções são gatilhos específicos de cada regra acontecem depois de uma regra ter sido detectada.

### 3.2.3 Obtendo as regras

Existem dois conjuntos de regras distribuídas no site oficial do Snort. O conjunto de regras da comunidade de usuários que é distribuída gratuitamente para todos os usuários e as regras certificadas pela equipe de pesquisa da Sourcefire que são disponibilizadas da seguinte maneira (SNORT, 2012):

- Assinantes: receberão o conjunto de regras em tempo real, mediante pagamento, à medida que são liberadas para os clientes da Sourcefire;
- Usuários cadastrados: receberão conjuntos de regras quando são publicadas 30 dias após a liberação para os usuários inscritos;
- Usuários não cadastrados: terão acesso a um conjunto estático de regras contendo as regras mais recentes no momento do lançamento de cada versão Snort.

Independentemente da origem das regras, deve ficar claro que novas vulnerabilidades surgem a cada dia e as regras devem ser atualizadas constantemente.

### 3.2.4 Atualizando as regras automaticamente com Oinkmaster

Como já comentado é imprescindível ter as melhores e mais atuais regras do Snort para que se obtenha um bom desempenho na detecção de intrusões. O Snort em si não inclui qualquer meio de atualizar automaticamente as suas regras. Para facilitar essa tarefa foi criado um script em Perl chamado Oinkmaster para verificar atualizações tão frequentemente quanto desejadas no site do Snort (BAKER, 2007).

# 4 EXPERIMENTOS COM O SNORT

Este capítulo destina-se a apresentar os procedimentos de instalação e testes do sistema de detecção de intrusão Snort.

## 4.1 Ambiente de testes

Os equipamentos que foram utilizados como sensores do Snort possuem a seguinte configuração: dois computadores de uso pessoal com Intel Pentium 4 de 3.00 GHz, 2GB de memória e HD de 160GB com sistema operacional Linux Ubuntu 12.04. A instalação do sistema operacional seguiu as definições padrões.

Todos os sistemas precisam ter um sistema de logs eficiente. O Snort pode trabalhar com MySQL, Oracle, ou qualquer outro Open Database Connectivity (ODBC). Dessa forma, foi realizada a instalação do Snort integrado com o SGBD MySql objetivando uma maior facilidade na hora de gerar relatórios e analisar essas informações. O armazenamento feito num banco de dados facilita a utilização com ferramentas como Analysis Control for Intrusion Detection (ACID) com a qual obtemos informações úteis dos padrões de ataque com muita dinamicidade. Podemos ter estatísticas de quais os hosts que constantemente atacam a rede, a distribuição dos ataques pelos diferentes protocolos entre outras informações (REHMAN, 2003).

Nos experimentos, cada um dos sensores do Snort possuía todos os serviços sendo executados no mesmo host, sendo cada um deles conforme figura abaixo.

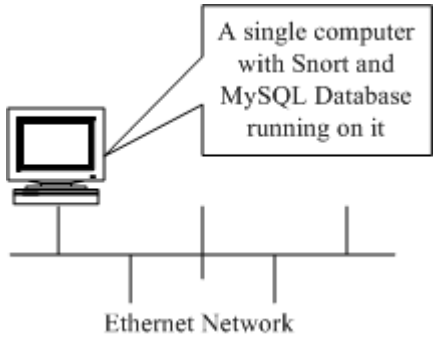


Figura 4.1: Servidor Snort (REHMAN, 2003).

O Roteador e firewall utilizado foi o Routerboard RB750G com o MikroTik RouterOS como sistema operacional. O RouterOS é um sistema operacional licenciado, stand-alone baseado no kernel Linux v2.6, que oferece várias funcionalidades voltadas para redes de computadores.

A rede utilizada para os testes não possuía muitos serviços disponíveis na internet como servidores web, e-mail etc. A rede era bem simples e possuía a topologia da rede era conforme apresentado abaixo.

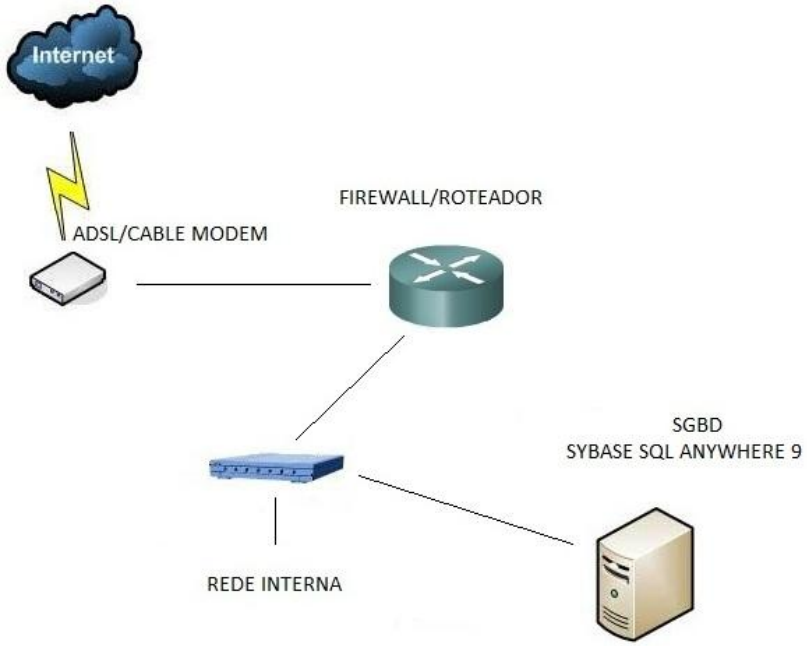


Figura 4.2: Topologia antes dos testes.

## 4.2 Instalação do Snort

APT - Advanced Package Tool é uma ferramenta de gerenciamento de pacotes moderna e fácil de usar que pode ser usada para a instalação de programas. Devido a sua facilidade, eficiência e o arsenal enorme de programas à disposição, essa foi a ferramenta escolhida para instalação dos pacotes necessários para os testes.

A instalação descrita a seguir foi feita da mesma forma em ambos os servidores como o Snort na versão 2.9.2.

Considerando que o sistema operacional foi instalado com as opções padrões, em seguida foram executados os seguintes passos:

```
#apt-get update
```

Como pré-requisito para instalação do Snort foi necessária à instalação de alguns pacotes que foram instalados com o comando abaixo:

```
#apt-get install mysql-server mysql-client
```

Nessa etapa foi escolhida a senha de root do servidor MySQL.

```
#apt-get install apache2 php5 php5-mysql
```

Nesse ponto foi escolhida a opção padrão em todas as perguntas durante a instalação para serem corrigidas posteriormente.

Optou-se pela instalação do pacote do Snort compilado com o modulo de saidaa capacidade de enviar logs para o MySQL.

```
#apt-get install snort-mysql
```

O Snort possui um script para criação das tabelas no banco. Optou-se por introduzir os comandos de criação do banco no inicio do script que cria as tabelas. Para isso foram executados os comandos:

```
#cd /usr/share/doc/snort-myssql
#zcat create_mysql.gz > cria_mysql.sql
#vim cria_mysql.sql
```

Após abrir o script no editor de texto, foram acrescentados os comandos abaixo com a finalidade de criar o banco. Os "xxxxxxxx" representam a senha usada.

```
CREATE DATABASE snort;
GRANT ALL PRIVILEGES ON snort.* TO snort@localhost IDENTIFIED BY 'xxxxxxxx';
FLUSH PRIVILEGES;
use snort;
```

Após, deve-se executar o script "cria_mysql.sql" no MySQL.

```
#cat cria_mysql.sql | mysql -u root -p
```

Foi criado o banco para o Snort. Em seguida foi necessário terminar o procedimento de instalação e configura-lo para utilizar o novo banco. É preciso apagar o seguinte arquivo para isso.

```
#rm /etc/snort/db-pending-config
```

Foi executado o comando abaixo para abrir novamente as perguntas feitas durante a instalação.

```
#dpkg-reconfigure snort-mysql
```

Comando para iniciar o serviço:

```
#/etc/init.d/snort start
```

Nesse momento foi finalizada a instalação do Snort, bastando agora apenas a instalação de uma ferramenta que facilite a visualização dos alertas e gere relatórios. Embora existam inúmeras ferramentas, na maioria delas, projetos atuais e poderosos, optou-se pela instalação do BASE, antigo projeto ACID por ser atender totalmente as necessidades dos testes e ser de fácil instalação.

```
#apt-get install acidbase
```

Por padrão o BASE pode ser acessado apenas pelo próprio host. É necessário liberar o acesso de acordo com as necessidades. Nos testes, o acesso foi liberado para a rede interna editando o arquivo de configuração e realizado recarda das configurações conforme abaixo:

```
#nano /etc/acidbase/apache.conf
allow from 192.168.3.0/255.255.255.0
#/etc/init.d/apache2 reload
```

O acesso é feito através do endereço http://seuip/acidbase.

Após a instalação dos sensores a topologia ficou conforme figura abaixo:

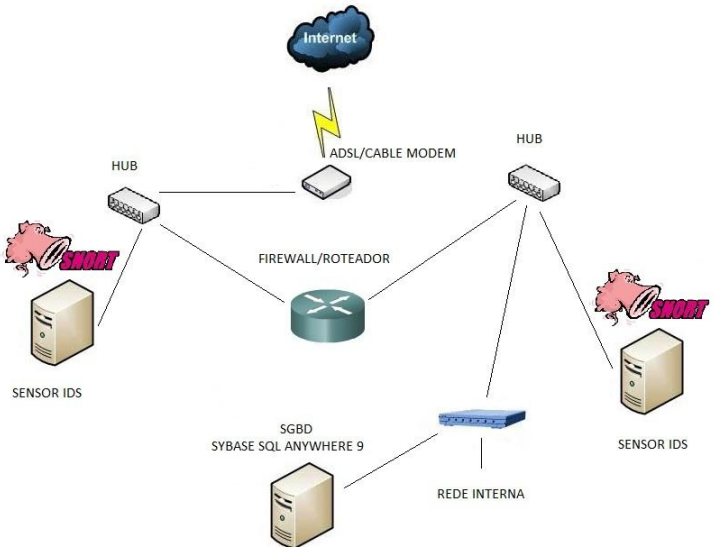


Figura 4.3: Topologia após instalar sensores IDS.

## 4.3 Configuração do Snort

O arquivo “snort.conf”, localizado em “/etc/snort/” controla tudo que o Snort observa, como ele se defende de um ataque, quais as regras que estão sendo usadas, quais pré-processadores entre outras configurações. Nesse arquivo, sem duvida, as informações mais importantes são configuradas e influenciarão o funcionamento do IDS como um todo. Entender o que tem nesse arquivo e como configurá-lo é essencial para o sucesso da implantação do Snort.

Nos testes, cada sensor teve suas configurações personalizadas de acordo com suas necessidades. Por exemplo, as variáveis HOME_NET e EXTERNAL_NET tiveram seus valores configurados conforme a classe de rede que cada uma pertence. Essa etapa foi muito importante para o bom funcionamento do sistema, pois essas variáveis são muito usadas, principalmente, nas regras e caso sejam configuradas erradas as regras não funcionarão.

Nos testes realizados foram alteradas apenas as variáveis que determinam a rede interna e externa, alem da inclusão ou exclusão dos de regras no arquivo de configuração.

### 4.3.1 Regras utilizadas

As regras utilizadas nos testes foram obtidas no site do Snort que são certificadas pela equipe de pesquisa da Sourcefire. A versão do Snort em ambos os sensores foi a 2.9.2 e as regras compatíveis com essa versão.

Alem disso se utilizou regras obtidas em sites e blogs de usuários do Snort, muitas das regras utilizadas foram obtidas num blog de usuários do Snort (BLEEDINGSNORT, 2012). As demais regras usadas foram feitas pelo autor em busca de suprir as necessidades de casos especiais da rede.

Embora não apareçam nos casos descritos neste capitulo, foram feitos inúmeros testes com regras obtidas em fórum de usuários e sites sem relação com o site do fabricante.

## 4.4 Resultados obtidos

Após a instalação e configuração do IDS foram realizados testes a fim de avalizar o funcionamento da solução.

### 4.4.1 Medidas Preliminares

Após a instalação e execução com as regras padrão, milhares de alertas invadiram os logs do Snort. Num primeiro momento deu a entender que eram ataques reais, mas eles eram, no mínimo, duvidosos em função de uma quantidade absurda.

Em pouco tempo centena de milhares de regras já haviam sido detectadas, levantando a forte suspeita de muitos falsos positivos e alertas repetidos para o mesmo evento. Após a analise de cada uma dos diferentes tipos de alertas foi constatado que muitos se tratavam de falsos positivos. Sem dúvida alguma, a principal e maior dificuldade foi conseguir identificar os resultados nos logs, entender os casos, e, principalmente, os falsos positivos.

Foi necessário apagar as informações das tabelas para recomeçar os logs porque num certo momento a marca de um milhão de alertas.

Em muitas situações foi constatado que a mesmas regras geravam dezenas e até centenas de alertas num pequeno intervalo de tempo, exigindo a configuração de um limiar para fazer com que num determinado intervalo de tempo só fosse gerado um alerta para cada evento que envolvesse os mesmo IP de origem e destino naquela regra.

A partir de então se optou por comentar as linhas dos arquivos de configuração que faziam referencia às regras e descomentar uma a uma e testá-las. Nesse momento começaram a surgir resultados mais interessantes.

Definitivamente o Snort não é um software em que se instala e sai usando, é necessária muita customização para obter resultados satisfatórios.

Na maioria dos testes, foram utilizadas apenas algumas regras ativas, e a maioria delas foram obtidas na comunidade de usuários ou desenvolvidas pelo autor para captura algum trafego peculiar da rede de testes.

Os resultados estavam aquém do esperado, inúmeros alertas e na grande maioria deles eram falsos positivos. Isso fez com que o Snort não passasse credibilidade em relação aos seus alertas. Foi a partir desse momento que se percebeu a necessidade de procurar regras na comunidade de usuários, não só as testadas e certificadas pelo time de pesquisa de vulnerabilidades da Sourcefire, regras especificas para cada situação que se desejava monitorar. Foi também nesse momento que foi percebido a necessidade de criar regras personalizadas para cada caso que se desejava monitorar dentro das limitações do ambiente.

Todas as regras, em seu cabeçalho, fazem referencia aos endereços de origem e destino, que na maioria das vezes possui variáveis que representam a rede interna e externa, o que faz com que a definição desses valores nas configurações seja de suma importância para o bom funcionamento do IDS.

Num primeiro momento, foram criadas regras bem simples para testar o funcionamento do Snort. Foram criadas regras que basicamente procuravam o conteúdo dos pacotes TCP em busca de algo que fugisse da política da instituição. No caso da regra abaixo, sendo usada no sensor IDS interno à rede, foi possível perceber quando

algum usuário estava burlando a sistema de proxy. Assim, os resultados começaram a aparecer quando começaram a ser feitos testes direcionados.

```
alert tcp $HOME_NET any -> any any (content:  "www.youtube.com"; msg: "acesso ao youtube!!"; threshold:  type limit, count 1, seconds 60, track by_src; classtype: policy-violation; sid: 1000002; rev: 1;)
```

### 4.4.2 Exemplo de um falso positivo

Snort é uma ferramenta que exige bastante conhecimento para interpretar os resultados e, principalmente, encontrar os casos em que os alertas não são o que parecem ser. O exemplo abaixo representa um falso positivo. Esse alerta foi detectado pelo sensor que fica entre o firewall e a internet e não apareceu no sensor interno.

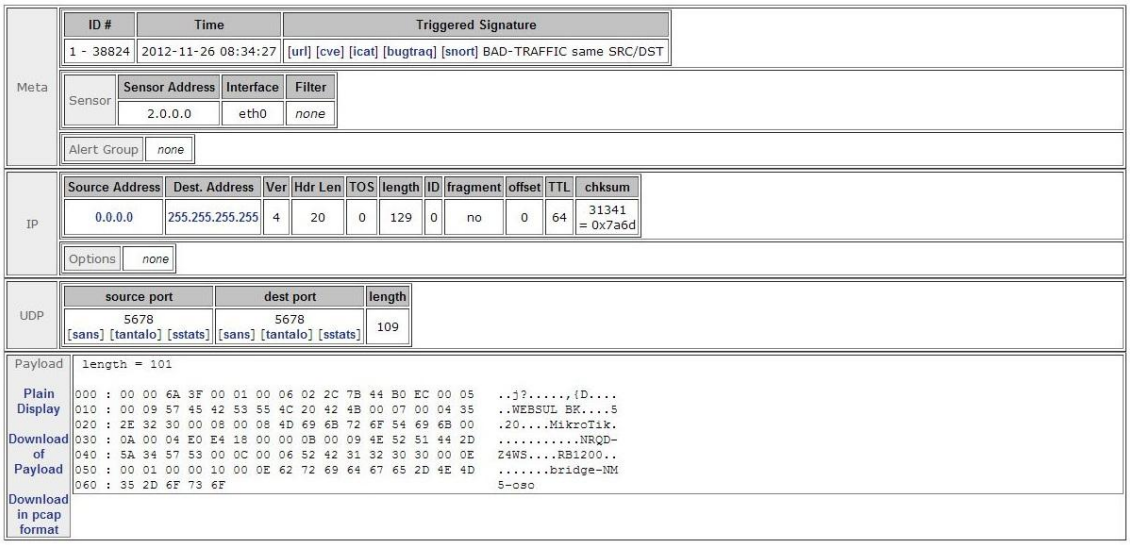


Figura 4.4: Snort alertando um falso positivo.

A regra que disparou o alerta acima foi a seguinte:

```
alert ip any any -> any any (msg:"BAD-TRAFFIC same SRC/DST"; sameip; reference: bugtraq,2666; reference: cve,1999-0016; reference: url,www.cert.org/advisories/CA-1997-28.html; classtype: bad-unknown; sid: 527; rev: 8;)
```

Segundo SnortID (2012) essa regra é capaz de identificar um ataque de ferramentas para ataques de negação de serviço como o Land. Mas no caso desse alerta, como pode ser visualizado na figura abaixo, trata-se de um pacote do protocolo MNDP, usado por roteadores Mikrotik com a função de encontrar outros roteadores Mikrotik vizinhos.

A interface do BASE possibilita fazer download do pacote no formato .pcap para posteriormente ser visualizado por softwares como o Wireshark.

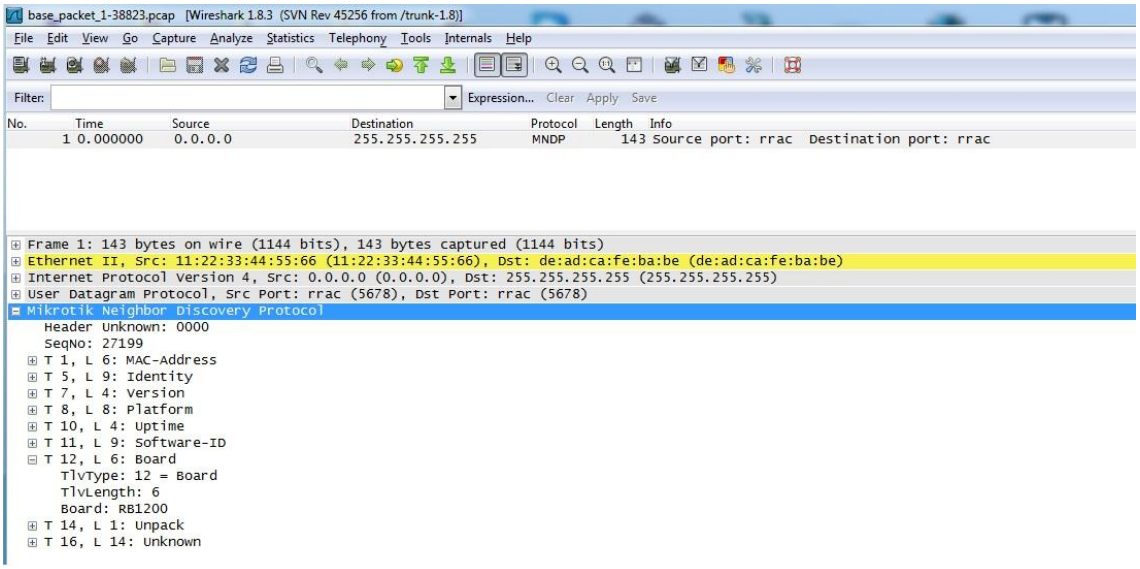


Figura 4.5: Pacote visualizado pelo Wireshark.

Esse é apenas um exemplo de muitos casos similares em que foi necessária uma análise para identificar, entender o alerta e concluir que se tratava de um falso positivo que ocorria dezenas de vezes por minuto.

### 4.4.3 Acesso externo ao banco de dados Sybase

Devido a sua simplicidade, a rede usada para os testes possuía poucos serviços externos, dentre esses serviços existia a disponibilidade de acesso a um banco de dados.

Foi utilizado o Nmap para fazer uma varredura de portas no firewall da rede a fim de verificar as portas abertas para a internet, como se pode verificar na figura abaixo, a porta 2638, referente ao SGBD Sybase, estava aberta.

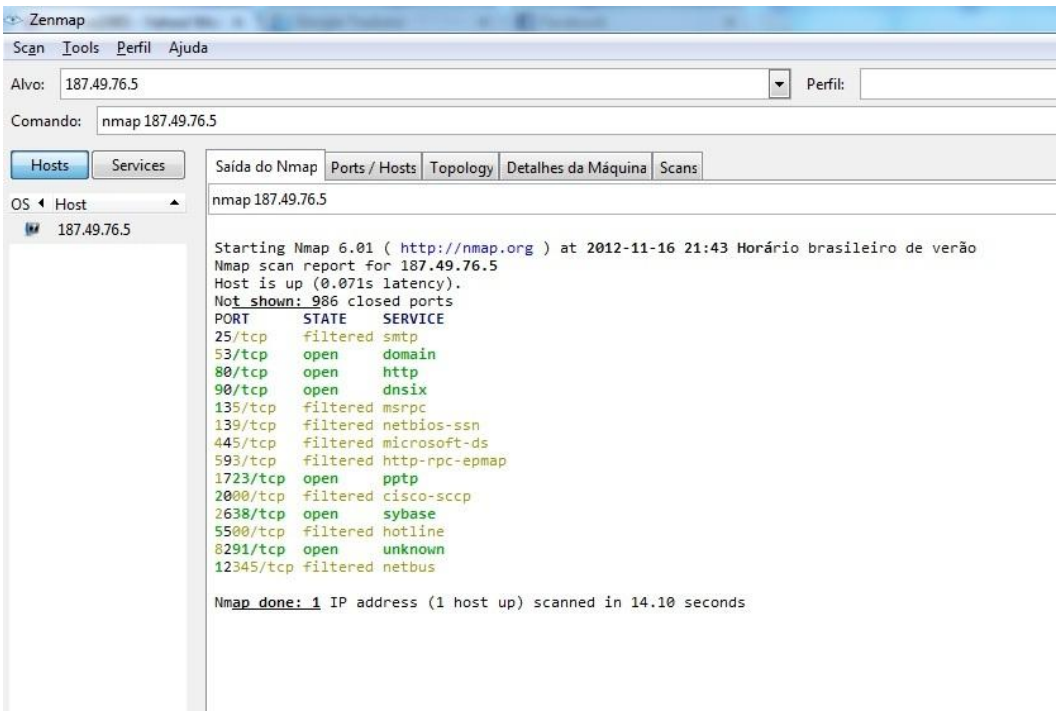


Figura 4.6: Scan NMAP no gateway.

Existe uma aplicação web que faz uso dessas informações, e ninguém além desse servidor, cujo IP é 177.43.56.218, precisaria ter acesso a esses dados.

A regra do firewall que permitia esse acesso ao firewall era dar acesso a aquela porta de uma maneira genérica, ou seja, qualquer equipamento que tentasse acessar.

Regra:

```
/ip firewall nat add chain = dstnat src protocol = tcp dst-port = 2638 action = dst-nat to-address = 192.168.3.250 to-ports = 2638 comment = "Redirecionamento para SGBD" disabled = no
```

Na figura abaixo se verifica o sensor externo à rede cujo IP é 187.49.76.6 capturando o trafego do IP de origem 189.73.134.129 à porta 2638 do IP do firewall da rede.

Com a finalidade de capturar todo o trafego a aquela foi configurada a seguinte regra ao Snort que identificava todos acessos que tinham como destino a referida porta que tivesse origem de um IP que não fosse o do servidor ao qual se desejava liberar acesso.

```
alert tcp !177.43.56.218 any -> any 2638 (msg: "Acesso externo ao Sybase na porta 2638"; threshold: type limit, track by_src, count 1, seconds 30; sid: 20501219; rev:1;)
```

Foram realizados testes de acesso ao banco de dados originados um IP diferente do IP do servidor, e, como era esperado, ambos os sensores do Snort alertaram tais acessos, conforme verificamos na figura abaixo. O sensor externo tem IP 187.49.76.6 e o sensor interno tem IP 187.49.76.5.

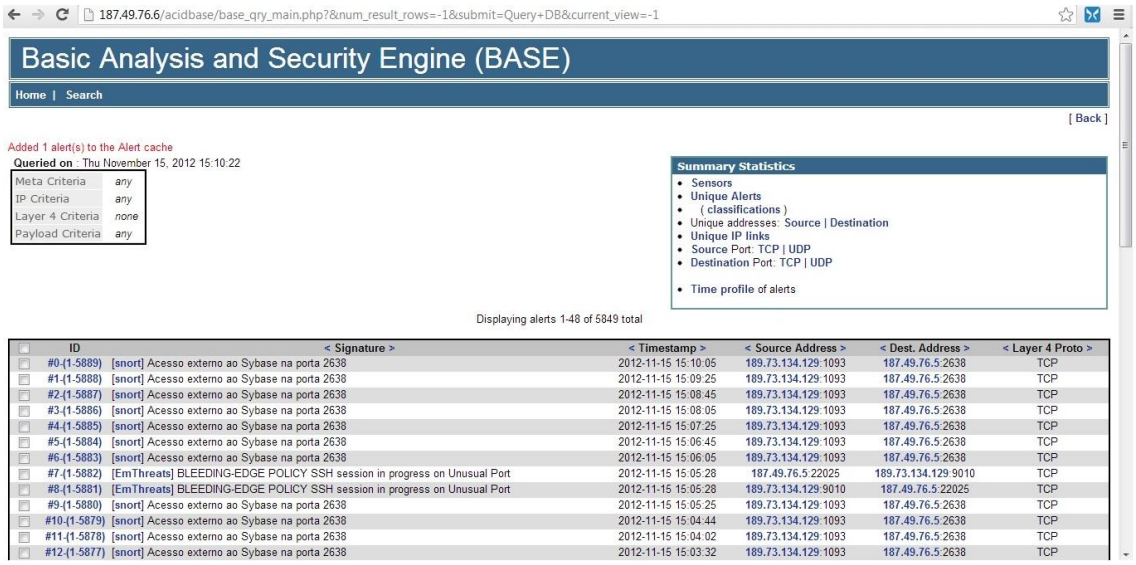


Figura 4.7: Sensor externo a rede detectando acesso ao Sybase.

O mesmo trafego também é capturado pelo sensor interno da rede, conforme pode ser visto na figura abaixo.

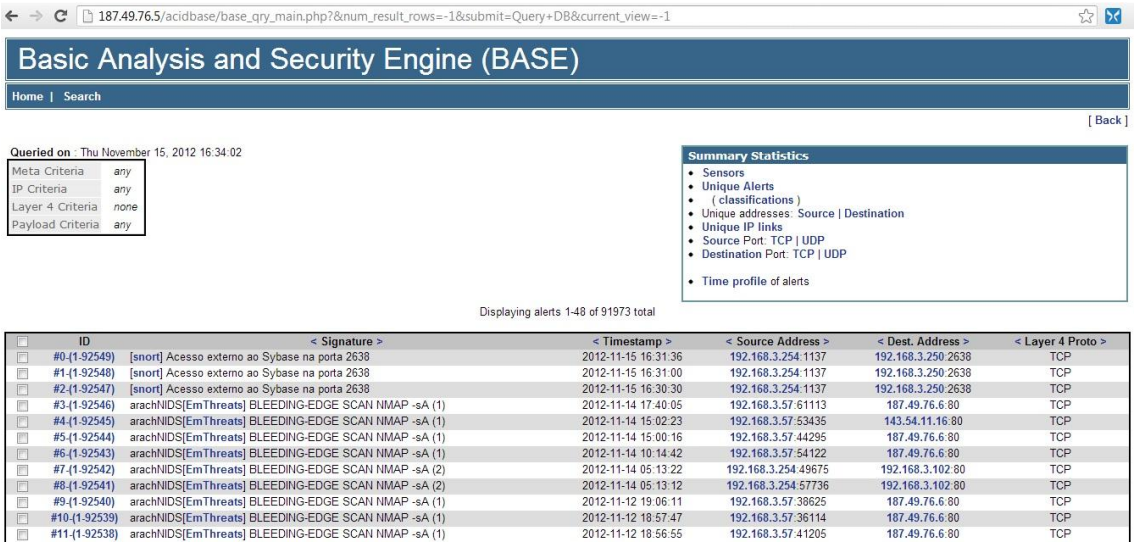


Figura 4.8: Sensor atrás do firewall detectando acesso ao Sybase.

Objetivando diminuir a repetição do mesmo alerta para cada pacote que passasse pelos sensores foi utilizado um limiar que fazia com que fosse emitido apenas uma alerta daquele mesmo acesso a cada 30 segundos.

Após verificar que a regra estava permitindo um acesso desnecessário por parte de qualquer usuário no mundo ela foi modifica para que apenas o IP que realmente precisava ter acesso, e era o único, pudesse ter acesso.

Alterando a regra para:

```
/ip firewall nat add chain = dstnat src-address = 177.43.56.218 protocol = tcp dst-port = 2638 action = dst-nat to-address = 192.168.3.250 to-ports=2638 comment = "Redirecionamento para SGBD" disabled=no
```

Assim que a regra foi alterada foram realizados novos testes e foi observado que apenas o sensor externo à rede capturou a tentativa de acesso ao banco de dados. Evidenciando que a nova regra teve efeito e impediu um acesso por um IP não autorizado.

### 4.4.4 Ataque de dicionário

Foram realizados testes de ataques de dicionário de um host da rede em direção ao firewall rede cujo IP interno era 192.168.3.254.

A origem do ataque foi de uma maquina Linux com a distribuição Backtrack devido às inúmeras ferramentas de teste de penetração presentes nela.

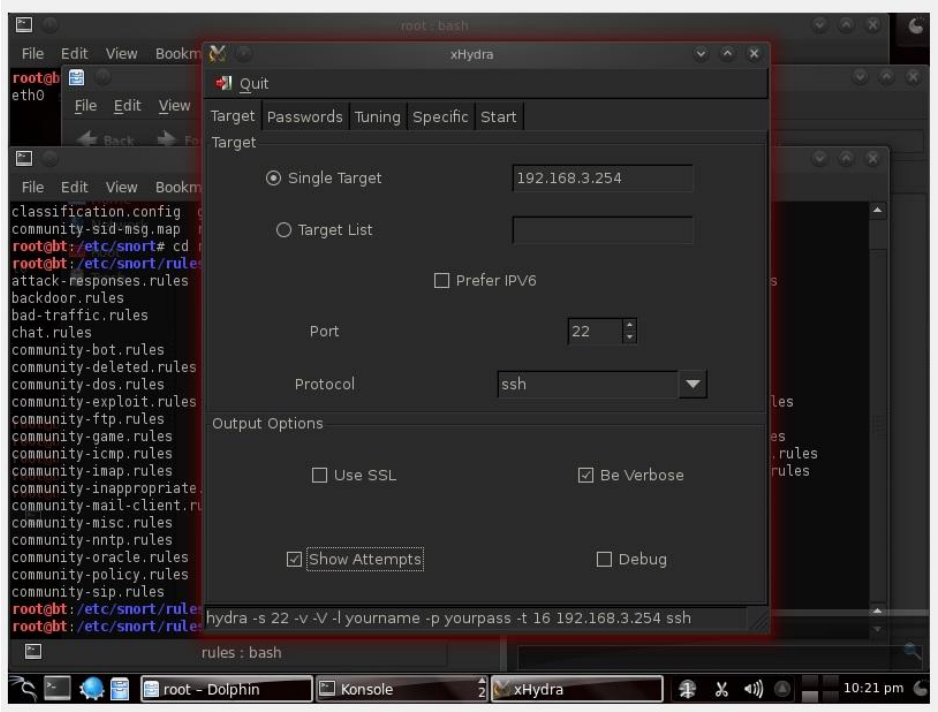


Figura 4.9: Ataque por dicionário com Hydra.

A regra utilizada para tal detecção foi a seguinte:

```
alert tcp $HOME_NET any -> $HOME_NET 22 (msg: "BLEEDING-EDGE Potential SSH Scan INTERNO"; flags: S; flowbits: set,ssh.brute.attempt; threshold: type threshold, track by_src, count 5, seconds 120; classtype: suspicious-login; reference:url,www.whitedust.net/article/27/Recent%20SSH%20Brute-Force%20Attacks/; sid: 20901219; rev:12;)
```

Na figura abaixo podemos verificar que o Snort detectou essa tentativa de ataque por dicionário.

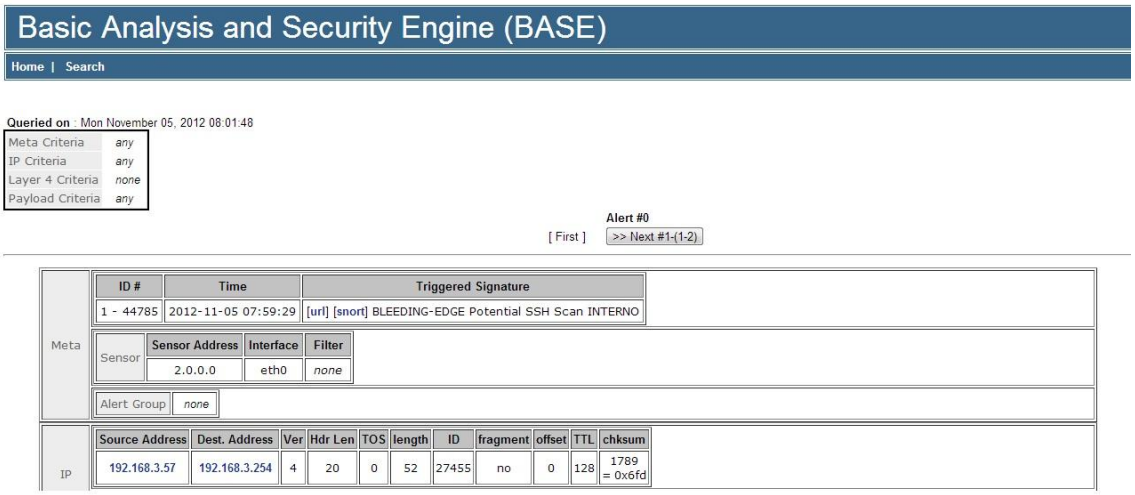


Figura 4.10: Detecção do ataque do Hydra.

### 4.4.5 Identificação de tráfego Peer-to-peer

Um teste que pode ser feito é de dentro pra fora da rede, no caso de aplicações que não deveriam esta sendo usadas por ferirem a política de uso da empresa. Um exemplo disso é o uso de torrent para baixar conteúdo da internet, na maioria das vezes, pirata.

O uso de torrents em redes corporativas é um problema grave que sobrecarrega o link de internet para atividades que, na maioria das vezes, está relacionado com pirataria de filmes e músicas.

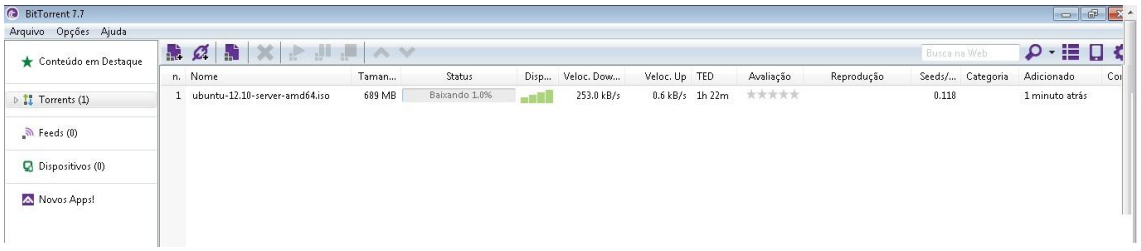


Figura 4.11: BitTorrent em execução.

Para podem identificar o uso de clientes de torrent foi configurado a seguinte regra no Snort:

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any (msg: "BLEEDING-EDGE P2P BitTorrent peer sync"; flow: established; content: "|0000000d0600|"; offset: 0; depth: 6; reference: url,bitconjurer.org/BitTorrent/protocol.html; classtype: policy-violation; sid: 2000334; rev:7;)
```

Essa regra detecta o cliente de torrents BitTorrent.

Propositalmente a regra acima foi deixada sem um limiar nos alertas e é possível verificar que existem inúmeros alertas num intervalo de alguns segundos, mostrando a importância diminuírem o numero de alertas de um mesmo evento.

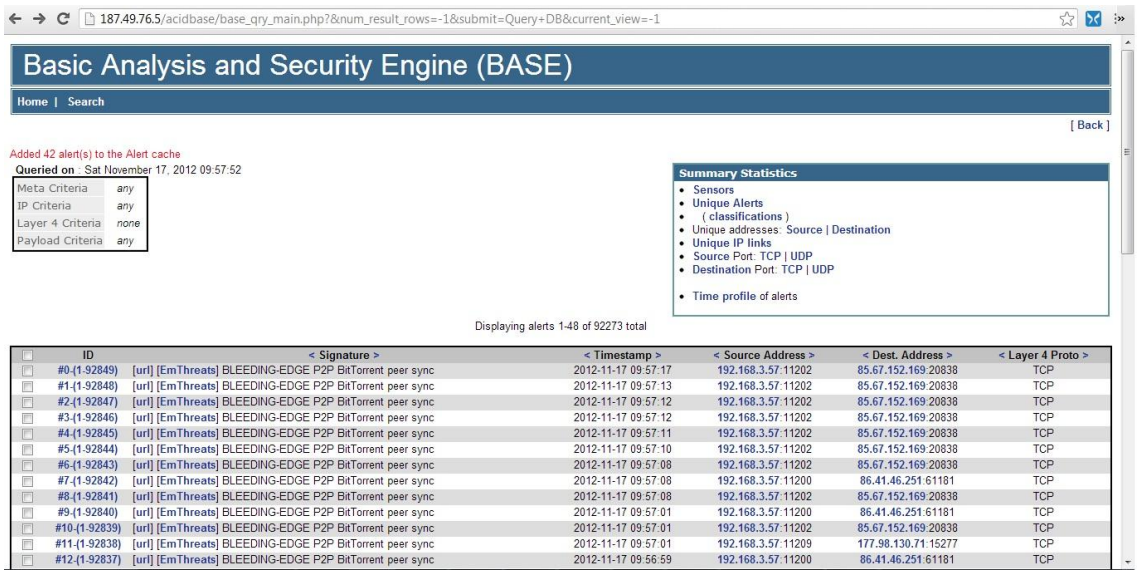


Figura 4.12: Snort detectando BitTorrent.

### 4.4.6 Outros casos identificados

Alguns casos menos importantes que foram identificados, porém fogem um pouco a proposta inicial de avaliação interna e externa ao firewall, caracterizam questões internas à rede.

Ultrasurf é um produto de UltraReach Internet Corporation. Originalmente criado para ajudar usuários de internet na China a encontrar segurança e liberdade online, Ultrasurf se tornou um dos softwares que os milhares de usuários usam para burlar os mecanismos de controle de acesso à internet.

Esse tipo de software, na maioria das vezes, fere a política de segurança e restrições de uma empresa. Um IDS pode ajudar a detectá-lo. Durante os testes foram encontradas pacotes desse software trafegando pelo firewall.

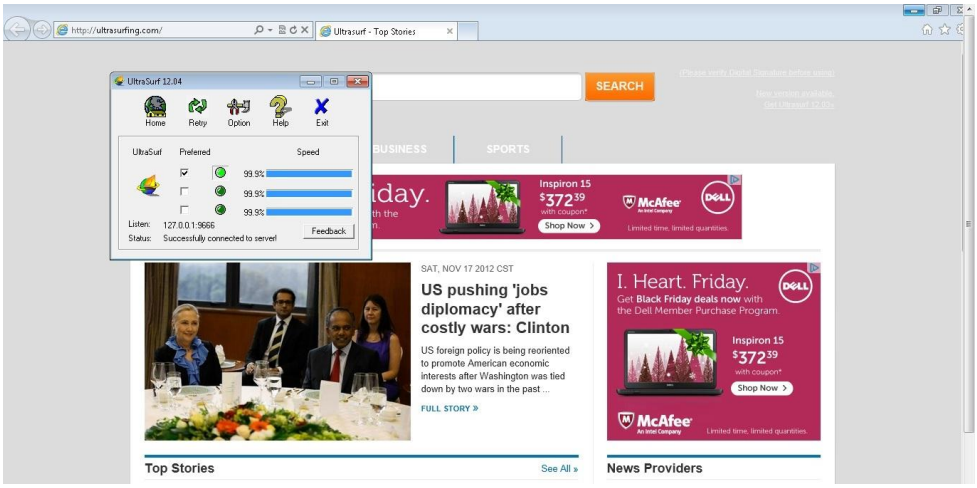


Figura 4.13: Acessando internet com Ultrasurf.

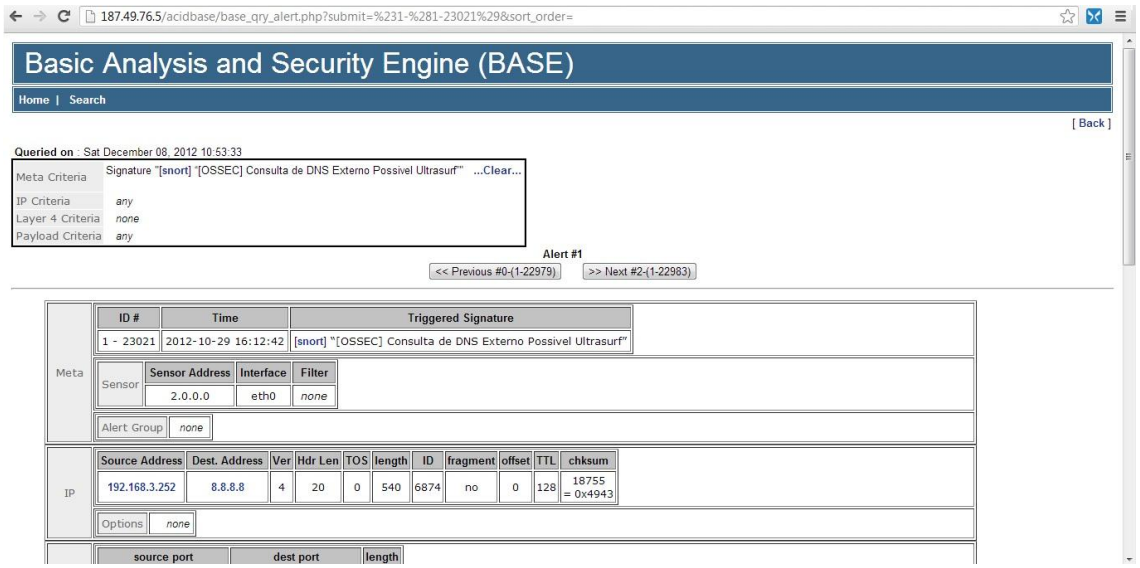


Figura 4.14: Ultrasurf sendo detectado.

A seguinte regra foi utilizada para identificar o trafego:

```
alert udp $HOME_NET any -> $EXTERNAL_NET  53 (msg: "Possible External
Ultrasurf  DNS Query"; content: "|00 00 00 00 00 00 00 00 00 00 00 00
```

```
00 00 00 00 00 00 00 00 00 00|"; classtype: policy-violation;
threshold: type limit, track by_src, count 1, seconds 5; sid: 1000089;
rev: 2;)
```

A regra busca encontrar consultas DNS usadas durante o processo de execução do software para detectá-lo.

Além desses casos também foram detectados tráfego de pacotes de cavalos de troia, ataques de força bruta e exploits, entre outros. Entretanto, nenhum desses casos consistia uma ameaça grave à segurança da rede, e por isso não são descritos neste trabalho.

# 5 CONCLUSÃO

Por ser uma das ferramentas de segurança mais conhecidas, o firewall passa uma falsa sensação de segurança, pois ele não pode defender a rede de ataques internos, engenharia social, vírus e modems internos que são uma grande ameaça para a segurança de uma rede.

Um firewall é apenas uma parte de um conjunto de componentes de um sistema de segurança necessário para a proteção das organizações. Assim a ideia de que um firewall é a solução para todos os problemas de segurança é um conceito equivocado que acaba colocando em risco toda a organização.

Os IDSs tem a capacidade de documentar as ameaças à rede de uma organização, tendo um papel importantíssimo junto ao administrador de redes, ajudando a tomar as decisões sobre as medidas que devem ser tomadas a fim de evitar que essas vulnerabilidades sejam exploradas, diminuindo os riscos.

A presença de falsos positivos é uma medida inversamente proporcional à de falsos negativos, quando mais específica for uma regra menor serão os falsos positivos e maiores serão os falsos negativos. Quanto mais genérica for uma regra maior será a possibilidade de pegar vários ataques diferentes, diminuindo os falsos negativos, porém aumentará a chance de que o tráfego normal seja considerado uma intrusão, ou seja, mais falsos positivos. Essa é a decisão mais importante a ser tomada durando a configuração de um IDS, pois terá papel fundamental no desempenho deste, o que poderá determinar o sucesso ou não de um projeto.

Todo e qualquer sistema de detecção de intrusão gera falsos positivos. Configurando um IDS corretamente, no máximo será possível reduzi-los, mas nunca eliminá-los completamente.

Os resultados obtidos são diretamente proporcionais à qualidade das regras utilizadas e a configurações feitas nos arquivos de configuração que precisam estar adequadas à rede onde o IDS está instalado.

A política de segurança deve expressar o que os usuários devem e não devem fazer em relação aos diversos componentes do sistema, incluindo o tipo de tráfego permitido nas redes. Sem ela previamente definida não faz sentido tentar analisar os resultados de uma detecção de um IDS, ou analisar as regras de um firewall. É ela que determina como um firewall e os demais dispositivos de segurança devem ser configurados. Um IDS tem papel muito importante na verificação do grau de qualidade da implementação dessa política, na maioria das vezes acusando um tráfego que deveria estar bloqueado.

Toda e qualquer ameaça existente faz uso de uma vulnerabilidade e a probabilidade de que essa ameaça se concretize caracteriza o risco de segurança que uma rede possui.

Portanto é importante que sejam usadas ferramentas como IDS para encontrar problemas nas configurações de outros dispositivos relacionados à segurança como firewalls. Sendo assim a segurança de um ambiente computacional se deve ao trabalho conjunto de uma serie de ferramentas, sendo uma complementar à outra.

Que portas estão abertas? Que tipos de pacotes estão passando pela rede? Essas informações são imprescindíveis para que qualquer administrador de redes possa tomar qualquer decisão e saber onde deve concentrar seus esforços para melhorar a qualidade dos serviços e, principalmente, garantir a integridade e segurança das informações.

Na realização desse trabalho, uma das maiores dificuldades foi durante a análise dos resultados, filtrar as regras e conseguir identificar os falsos positivos, pois é necessário bastante conhecimento e experiência para avaliar os alertas e entender cada uma das regras para poder configurá-las de acordo com as necessidades de cada rede.

## 5.1 Sugestões para trabalhos futuros

Para trabalhos futuros se poderia fazer uma integração com o SnortSam, que é um sistema de resposta ativa que interage tanto com firewalls comerciais quando com os de código aberto para bloquear endereços IP. O Snort pode ir muito alem do que foi implementado neste trabalho, ele pode atuar com um sistema de prevenção de intrusão, bloqueando o trafego, juntamente com o firewall. Entretanto para isso é necessário muitos testes e a certeza de que as regras habilitadas gerem uma quantidade mínima de falsos positivos, pois esses podem gerar sérios transtornos para os usuários no caso de bloqueios indevidos. Além disso, os testes devem ser realizados numa rede de maior porte para que se tenha uma diversidade maior de serviços prestados em nível de internet para que seja possível obter melhores casos de intrusão.

# REFERÊNCIAS


ALENCAR, Antonio Juarez. **Análise de risco em gerência de projetos.** 1.ed. Rio de Janeiro: Brasport, 2005.

BACE, r.; MELL, P. **Intrusion Detection Systems**. Scotts Valley: NIST, 2001. (NIST Special Publication SP 800-31).

BAKER, Andrew. **Snort IDS and IPS Toolkit**. Jay Beale's Open Source Security. Rockland: Synress, 2007.

BEALE, Jay. **Snort 2.1 Intrusion Detection.** 2.ed. Syngress Publishing Inc.: Rockland, 2004.

BEZERRA, Edson Kowask. **Gestão de Riscos de TI: NBR 27005.** 1.ed. Rio de Janeiro: RNP/ESR, 2011.

BHARDWAJ, Pawan K. **A+, Network+, Security+ Exams.** 1.ed., Sebastopol: O'Reilly, 2007.

BLEEDINGSNORT. **Bleeding Snort: Blog about computer and network security and protection.** 2012. Disponivel em: <http://www.bleedingsnort.com>. Acesso em: setembro de 2012.

CARISSIMI, Alexandre da Silva; Rochol, Juergen; Granville, Lisandro Zambenedetti. **Redes de computadores.** Porto Alegre: Bookman, 2009.

COMER, Douglas E. **Redes de computadores e a internet**. 4.ed. Porto Alegre: Bookman, 2007.

KOMAR, Brian; BEEKELAAR, Ronald; Wettern, Joern. **Firewall for Dummies**. 2.ed. New York: Ed. Wiley Publishing, Inc, 2003.

KUROSE, James F. **Redes de computadores e a Internet: uma abordagem top-down**. 5. ed. São Paulo: Addison Wesley, 2010.

NAKAMURA, E. T; GEUS, P. L. **Segurança de Redes em Ambientes Cooperativos.** SãoPaulo: Novatec, 2007.

NORTHCUTT, Stephen. **Desvendando segurança em redes.** 1.ed. Rio de Janeiro: Campus, 2002.

OLIVEIRA, Wilson. **Técnicas para Hackers: soluções para segurança**. Porto: Centro Atlântico, 2003.

REHMAN, Rafeeq. **Intrusion Detection with SNORT: Advanced IDS Technique Using SNORT, Apache, MySQL, PHP, and ACID**. 1.ed. New Jersey: Prentice Hall, 2003.

ROESCH, Martin; GREEN, Chris. **Snort Users manual 2.9.1.** [S.1.]: Sourcefire Inc., July 14, 2011. Disponível em <http://www.snort.org/assets/166/snort_manual.pdf> Acesso em abr. 2012.

SCARFONE, Karen; MELL, Peter. Guide to Intrusion Detection and Prevention Systems. **NIST - National Institute of Standards and Technology**, Gaithersburg, 2007, Disponível em: <http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf>. Acesso em: setembro 2012.

SCOTT, Charlie; WOLFE, Paul; HAYNES, Bert. **Snort for Dummies**. Hoboken: For Dummies, 2004.

SCRIMGER, Rob; et al. **TCP/IP: a Bíblia**. 1.ed. Rio de Janeiro: Campus, 2002.

SNORT. **Snort.** 2012. Disponível em: http://www.snort.org/. 2012. Acesso em: agosto 2012.

SNORTID. **Snort ID.** 2012 Disponível em: <http://www.snortid.com/snortid.asp?QueryId=1:527>. Acesso em: novembro 2012.

STRAUB, Marcos; BOOS, Arthur; et al. IPS UFRGS: **A implementação de bloqueios automáticos progressivos integrada ao Sistema de Registro de Estações da UFRGS**. Workshop de Tecnologia da Informação das IFES, Goiânia, 2012.

TANENBAUM, Andrew S. **Redes de computadores.** 4.ed. Rio de Janeiro: Campus, 2003.

TEIXEIRA, Júnior; HELVÉCIO, José. **Redes de Computadores: Serviços, Administração e Segurança.** 1. ed. São Paulo: Makron, 1999.