PROJETO DE REDES
www.projetoderedes.com.br

Honeypot

Autores:
Carlos Manoel, Felipe Soares e Simone da Silva

UNIGRANRIO – Universidade do Grande Rio

Escola de Informática

Sistema de Informação

Honeypot:
Enganando e conhecendo o inimigo

Carlos Manoel de Oliveira Junior

Felipe Soares de Deco

Simone da Silva Antonio

Duque de Caxias, RJ – Brasil.

Honeypots: enganando e conhecendo o inimigo.

Carlos Manoel de Oliveira Junior

Felipe Soares de Deco

Simone da Silva Antonio

Projeto final de curso submetido ao corpo docente da escola de informática da universidade do grande rio como parte dos requisitos necessários para a obtenção do grau de bacharel em Informática.

Aprovada por:

_____

Prof. Etienne Oliveira B. Sc.

_____

Prof. Arnaldo Lyrio M. Sc.

_____

Prof. Arnaldo Vieira, Ph.D.

Duque de Caxias, RJ – Brasil.
2004

OLIVEIRA JUNIOR, CARLOS MANOEL DE
DECO, FELIPE SOARES DE
ANTONIO, SIMONE DA SILVA
Honeypot: Conhecendo e Enganando o Inimigo [Duque de Caxias] 2004
xi, 48 p. 29,7 cm (Escola de Informática, B.Sc., 2004)
Projeto de Final de Curso - Universidade do Grande Rio, Escola de Informática
1. Honeypot
2. Sistema de Segurança
3. Honeynet
4. Honeytoken
I. EIN/UNIGRANRIO II. Título (série)

Dedicatória

Dedicamos este trabalho aos nossos pais e irmãos e a todos aqueles que de alguma forma colaboraram para a elaboração deste trabalho.

Agradecimentos

Carlos Manoel de Oliveira Junior

A Deus por mais esta benção, a minha super mãe Neide, as minhas irmãs Cris e Jana, ao meu sobrinho Andrew e a minha noiva Shirley que eu amo muito.

Felipe Soares de Deco

Agradeço a Deus e a meus pais (Márcia e João) pelo dom da vida. Ao meu irmão (Fernando). Aos meus amigos (João Paulo, Ju, Primão) pelo apoio e a força. Ao meu Amor, Marta Patrícia, que sempre me ajuda e mantém minha vida “agitada”.

Simone da Silva Antonio

A Deus por mais um sonho realizado, aos meus pais Sergio e Elienai por toda dedicação e apoio. Aos meus irmãos Paulo, Marcos, Cláudio e Rose, e a todos os meus familiares pelo carinho e compreensão.

Dedicamos aos nosos amigos “Friends Network”, ao professor Etienne que nos orientou durante o desenvolvimento deste trabalho e a todos que nos ajudaram na concretização deste trabalho.

Resumo do Projeto Final de Curso apresentado a Escola de Informática da Universidade do Grande Rio como parte dos requisitos necessários para a obtenção do grau de Bacharel em Informática (B. Sc.)

Honeypot:

enganando e conhecendo o inimigo.

Carlos Manoel de Oliveira Junior

Felipe Soares de Deco

Simone da Silva Antonio

Novembro / 2004

Orientador: Prof. Etienne Oliveira

Honeypots são ferramentas de pesquisa agregados a redes de computadores, projetados para serem explorados e posteriormente comprometidos. Uma vez comprometidos, os *logs* gerados são utilizados para analisar comportamentos, táticas e ferramentas aplicadas no ataque. Este projeto apresentará os conceitos básicos dos *honeypots*, seus tipos, vantagens e desvantagens de se ter um *honeypot* e qual o seu valor para a empresa como um sistema para treinar sua segurança.

Abstract of Thesis presented to EIN/UNIGRANRIO as a partial fulfillment of the requirements for the degree of Bachelor of Science (B.Sc.)

Honeypot:

deceiving and knowing the enemy.

Carlos Manoel de Oliveira Junior

Felipe Soares de Deco

Simone da Silva Antonio

November / 2004

Advisor: Prof. Etienne Oliveira

Honeypots are research tools aggregated to computer networks projected to be exploited and compromised. Once they are hacked, we use the generated logs to analyze the behavior, tactics and tools used in the attack. In this project we will be presenting the Honeypots basics, their types, advantages and disadvantages of having a Honeypot and what is the value of the Honeypot for a company.

Sumário

Índice de Figuras

Lista de Abreviaturas

ASCII	American Standard Code for Information Interchange
DDoS	Distributed Denial of Service
DNS	Domain Name Server
DoS	Denial of Service
DTK	Deception Toolkit
FTP	File Transfer Protocol
HTTP	Hyper Text Transfer Protocol
IDS	Intrusion Detection System
IP	Internet Protocol
IRC	Internet Relay Chat
NDS	Novell Directory System
NTFS	Network File System
POP3	Post Office Protocol 3
S.O.	Operational System
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

Objetivo

Esta monografia tem como objetivo estudar a ferramenta de segurança *Honeypot*, sua utilização e os melhores métodos de implementação. Faremos uma abordagem geral de componentes de segurança utilizados na rede e estudaremos os comportamentos, táticas e ferramentas aplicadas pelos atacantes em seus ataques. Estaremos apresentando os conceitos básicos dos *Honeypots*, e qual o seu valor para as empresas.

1. Introdução

A evolução dos meios de comunicação vem nos proporcionando uma grande facilidade e velocidade para a troca de informações. Na mesma velocidade cresce o número de serviços disponíveis pela Internet.

Como a utilização dos serviços "*on-line*" está tendo um crescimento muito rápido, cresce também, quase que simultaneamente, as vulnerabilidades dos sistemas operacionais: vírus, *worms* e o acesso ilegal por pessoas desautorizadas a informações confidenciais.

Há pouco tempo, a violação das informações nas empresas ocorria através de pessoas com grandes conhecimentos técnicos, mas ultimamente estes conhecimentos podem ser dispensados, pois existem na Internet várias ferramentas e publicações que facilitam e ensinam como realizar ataques a pessoas ou a sistemas de empresas sem necessidade de grande conhecimento.

A gravidade dos ataques a informações tem suas variações, que podem ser desde a simples curiosidade, ou até mesmo a paralisação de sistemas inteiros, causando vários e graves problemas para as empresas que não dispõem de serviços de segurança que impeçam que estes ataques aconteçam e que protejam todo o sistema.

Com o advento das redes de computadores conectadas diretamente à Internet, torna-se necessário a utilização e a criação de novos recursos que garantam a segurança nos sistemas disponibilizados *on-line*, para que estes não se tornem potenciais alvos de diversos tipos de ataque, impedindo-os de atender às solicitações de serviços.

Para criação de um ambiente de Internet seguro são usados alguns recursos, como *Firewalls* e *IDS*. A ferramenta *Firewall* é utilizada para fazer controle do tráfego entre a rede local e a Internet e é baseada nos serviços requisitados através dos endereços de origem e destino dos pacotes, decidindo se o acesso será permitido ou negado. A ferramenta de *IDS* faz uma análise dos pacotes que trafegam na rede baseando-se no comportamento pré-definido e eventos maliciosos conhecidos como assinatura, fazendo a identificação de uma forma mais fácil e mais precisa de qualquer tipo de anomalia que possa ocorrer na rede. Os sistemas de *IDS* devem ser instalados em vários pontos da rede, principalmente na divisa da rede local com a Internet.

A utilização das ferramentas mencionadas acima não garante total segurança de uma rede, pois somente com a análise constante dos ataques alcançaremos um maior nível de segurança da rede.

Surge então uma nova opção de estudo de ataques e de segurança da rede. Para preencher esta lacuna foram criados os *Honeypots*, tema central deste projeto, que tem como objetivo a utilização de sistemas (reais ou virtuais) para serem comprometidos e estudados, visando o entendimento de comportamentos, táticas, ferramentas utilizadas no ataque e até mesmo o tipo de atacantes que o sistema recebeu, possibilitando assim a criação de novas ferramentas e rotinas de proteção.

Abordaremos também alguns conceitos básicos sobre as formas mais freqüentes de ataques, como são classificados e seus principais objetivos. Da mesma forma, estudaremos os tipos de atacantes, suas metodologias e ferramentas utilizadas para concretizar seus ataques.

2. Ameaças Virtuais

2.1. Atacantes

Segundo OLIVEIRA (2002), no início da década de 1960, as universidades americanas começaram a disponibilizar *mainframes* para os estudantes de computação. Nesta época surgiu o termo *hacker*. Eram pessoas extremamente capazes na função, utilizando o máximo dos recursos que o sistema computacional pudesse oferecer. Com o passar do tempo, esse termo passou a representar indivíduos que tinham a finalidade de invadir *sites* da Internet ou computadores corporativos, com o propósito de destruir ou apenas acessar informações sigilosas.

Atualmente os *hackers* são pessoas com grandes conhecimentos em técnicas de programação, análise de sistemas e especialistas em segurança de redes. Conhecem detalhadamente o computador e os sistemas operacionais, protocolos de rede e todas a vulnerabilidades conhecidas, e ainda são capazes de descobrir novas vulnerabilidades, que muitas das vezes são informadas aos desenvolvedores para que possam ser corrigidas.

Um segundo tipo de *hacker* são os *crackers*, os verdadeiros terroristas da Internet. Esses indivíduos utilizam seus conhecimentos de maneira maliciosa, aproveitando as vulnerabilidades existentes para obter informações sigilosas, invadir sistemas, destruindo informações, alterando programas e desfigurando *sites*. Um *cracker* é mais facilmente identificado, pois na maioria de seus ataques deixam evidências indicando a invasão.

Dentre esses também existem outros tipos de atacantes:

- ✓ Os *Pheakers*: *hackers* com vasto conhecimento em telefonia, onde suas especialidades são acessar sistemas telefônicos, permitindo efetuar ligações internacionais e nacionais, sem que sejam tarifadas pela companhia telefônica e sem serem descobertos.

- ✓ Os *Script-Kiddies*: são internautas sem conhecimentos que realizam suas invasões através de publicações e poderosas ferramentas encontradas na própria Internet. Esses tipos de invasores são facilmente identificados pelos profissionais de segurança de redes, pois por falta de conhecimentos sempre deixam seus rastros e na tentativa de realizar ataques geralmente são bloqueados pelos sistemas de segurança.

- ✓ Os *Lammers*: são atacantes curiosos que simplesmente realizam seus ataques repetindo fórmulas de invasão. Tem certo conhecimento em sistemas operacionais e são facilmente descobertos, pois costumam deixar sua assinatura eletrônica por onde passam e cometem uma série de falhas no momento da invasão confirmando que estiveram por lá.

- ✓ Os *Carders*: *hackers* criminosos que utilizam suas experiências para roubar cartões de crédito e utilizar em benefício próprio.

Um dos mais famosos *hackers* de toda a história da Internet foi Kelvin D. Mitnick. Desde a adolescência ele vem utilizando suas habilidades para invadir sistemas seguros. Utilizava o codinome Condor, invadiu sistemas militares, organizações federais, corporações financeiras e *sites* de várias empresas de tecnologia. Ainda adolescente teve vários problemas com o governo americano. Foi

preso pela primeira vez em seu apartamento, rendendo-se pacificamente no dia 15 de fevereiro de 1995, em seu apartamento por agentes federais.

Atualmente, em liberdade, Mitnick possui uma empresa onde trabalha como consultor.

2.2. Outras Ameaças

Alem dos ataques cometidos por pessoas, também existem ataques oriundos de programas planejados com um determinado objetivo. Eles tem sua classificação subdividida nas seguintes caracteristicas:

- ✓ Vírus: São programas de computador, e como tal, só podem agir em *softwares*, e na maioria das vezes não poderão causar danos ao *hardware* (micro, *drives* de disco rígido e flexível, *mouse*, vídeo, teclado, etc). Eles têm comportamento semelhante ao do vírus biológico: multiplicam-se, precisam de um hospedeiro, esperam o momento certo para o ataque e tentam esconder-se para não serem exterminados.
- ✓ *Worms*: Podem ser interpretados como um tipo de vírus mais inteligente que os demais. A principal diferença entre eles está na forma de propagação: os *worms* podem se propagar rapidamente para outros computadores seja pela Internet, seja por meio de uma rede local. Geralmente, a contaminação ocorre de maneira discreta e o usuário só nota o problema quando o computador apresenta alguma anormalidade. O que faz destes vírus inteligentes é a gama de possibilidades de propagação. O *worm* pode capturar endereços de *e-mail* em arquivos do usuário, usar serviços de *SMTP* (sistema de envio de *e-mails*)

próprios ou qualquer outro meio que permita a contaminação de computadores (normalmente milhares) em pouco tempo.

- ✓ Cavalo-de-tróia: São um tipo de praga digital que, basicamente, permitem acesso remoto ao computador após a infecção. Os cavalos-de-tróia podem ter outras funcionalidades, como captura de dados do usuário e execução de instruções presentes em *scripts*. Entre tais instruções, podem haver ordens para apagar arquivos, destruir aplicativos, entre outros. Quando um cavalo-de-tróia permite acesso ao computador, o que ocorre é que a praga passa a utilizar portas *TCP* e de alguma maneira informa a seu criador a "disponibilidade" daquele computador. Ainda, a praga pode se conectar a servidores e executar instruções que estejam disponíveis no momento do acesso.
- ✓ *Backdoors*: Um ponto de entrada deixado numa aplicação que permite acesso ao sistema ou aplicação sem conhecimento dos proprietários. Muitas vezes têm uma razão válida, como testes durante o desenvolvimento, mas não são retirados após entrada em produção. Podem também ser utilizados como parte de um ataque substituindo uma aplicação válida por outra com as mesmas funcionalidades, mais a de b*ackdoor*.
- ✓ *Spyware*: É um *software* que recolhe informações pessoais sem primeiramente informar ao usuário o que está fazendo, e sem que ele possa decidir se aceita, ou recusa a sua presença no sistema. As informações que o *spyware* recolhe podem ser desde informação relativa a todos os *sites* que visitou, até informações mais sensíveis, tais como nomes de usuário e senha. Um usuário pode se tornar alvo de *spywares* se transferir música de programas compartilhadores de arquivos, transferir jogos gratuitos de *sites* em que não confia, ou transferir outros programas de *software* de origens desconhecidas.

2.3. Alvos Escolhidos

De acordo com ROJAS (2003), para se efetuar um ataque em grande escala é necessário se eleger uma empresa que apresente em sua estrutura vulnerabilidades. Atualmente várias entidades são elegíveis. Eis algumas delas:

- ✓ governo e agências de defesa, devido ao baixo orçamento destinado às políticas de segurança;
- ✓ empresas multinacionais, que são as principais vítimas de tentativas de espionagem industrial;
- ✓ provedores de Internet, pois são facilmente acessados através da Internet e podem possuir acesso a conexões rápidas, utilizadas para transferência de grandes quantidades de dados, além do cadastro dos seus clientes;
- ✓ instituições financeiras e bancos, que são testados e atacados para execução de fraudes, como desvio de dinheiro e transferências de fundos, e que normalmente não são levados ao conhecimento público;
- ✓ companhias farmacêuticas, buscando resultados de pesquisas e desenvolvimentos de novos medicamentos.

2.4. Motivações

Segundo ROJAS (2003), é de suma importância identificarmos alguns fatores que impulsionam certas pessoas a escolherem sistemas de computador para serem violados:

- ✓ curiosidade ou busca de emoção: alguns intrusos possuem apenas a necessidade de saber o que o sistema possui;

- ✓ aprendizado: pequena parte daqueles que violam sistemas fazem isso para aprimorar seus conhecimentos;
- ✓ espionagem industrial: empresa ou organização que se dedica a atividades ilegais contra outra;
- ✓ espionagem nacional: é semelhante à espionagem industrial, exceto pelo fato de que um país começa a atacar os recursos de informática de outro;
- ✓ ignorância: não têm consciência de que suas ações são ilegais;
- ✓ necessidade de aceitação ou respeito: muitos intrusos se dedicam a atividades ilegais envolvendo o uso de computador devido à necessidade de aceitação e/ou respeito de outras pessoas;
- ✓ idealismo: alguns intrusos atacam sistemas por razões idealistas, se portando como heróis, protegendo o mundo de operações clandestinas de coleta de dados por parte do governo;
- ✓ ganhos financeiros: com freqüência, os intrusos são funcionários que obtêm acesso a sistemas financeiros para roubar dinheiro;
- ✓ anarquia: os anarquistas penetram nos sistemas simplesmente para produzirem discórdia;
- ✓ vingança: por parte de funcionários insatisfeitos.

2.5. Classificação dos Ataques

2.5.1. Ataques de Exploração

Segundo MARCELO e PITANGA (2003), o ataque de exploração é um tipo de ataque que consiste em obter o maior número de informações, sem causar nenhum tipo de dano ao sistema.

Alguns exemplos de ataques são: a espionagem digital, o furto de senhas e fraudes bancárias. As entidades mais vulneráveis são as instituições financeiras e entidades governamentais, devido ao tipo de informação que trafegam.

Um dos programas mais utilizados pelos atacantes são os *scanners* que tem por função varrer a rede em busca de máquinas com problemas, ou com portas de serviços abertas. Uma das ferramentas mais utilizadas pelos *scanners* é o *nmap*, que detecta os dispositivos na rede, e em seguida verifica se eles apresentam alguma vulnerabilidade (portas, serviços, S.O.). Com o *nmap* é possível fazer a varredura de redes inteiras, com opções variadas.

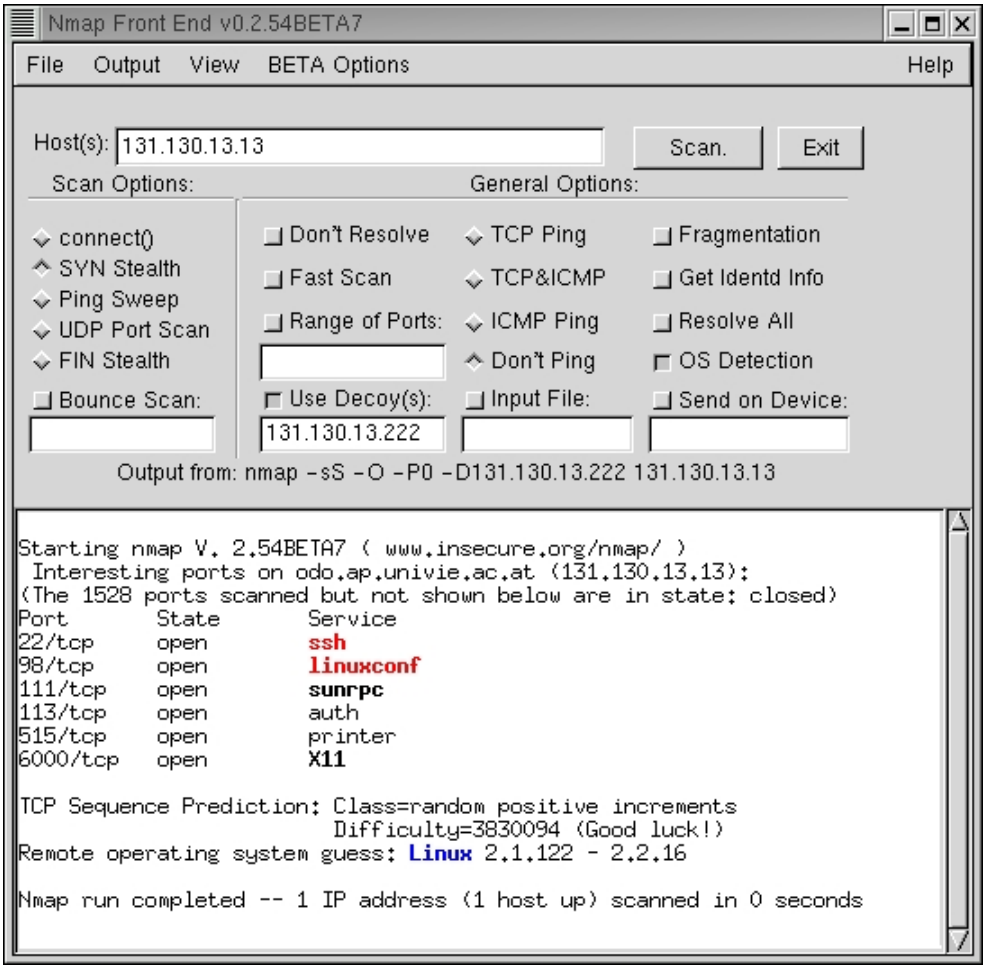

Fig1. Ferramenta NMAP

Fonte: (http://www.insecure.org/ **)**

2.5.2. Ataques de Paralisação

Segundo MARCELO e PITANGA (2003), este ataque tem como objetivo principal indisponibilizar temporariamente recursos de um sistema. É também conhecido como negação de serviços. Este ataque pode ser iniciado com ferramentas básicas (*ping, tracert*, etc.), causando uma sobrecarga de conexões aos sistemas alvos, que acarretará na negação do serviço.

Segundo a definição de STEIN e STEWART (2004), Negação de Serviço (DoS) é um ataque que permite que uma pessoa deixe um sistema inutilizável ou consideravelmente lento para os usuários legítimos através do consumo de seus recursos, de maneira que ninguém consegue utilizá-los.

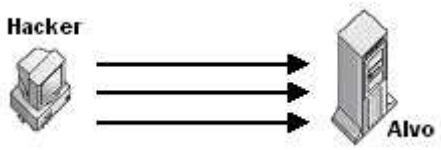

Fig2. Ataque DoS

Fonte: (http://www.microsoft.com **)**

Existe também uma nova forma de ataque DoS, conhecida como *Smurf*. Nela o atacante envia pacotes de *broadcast* na rede forjando o IP de destino do pacote. Ao receber este pacote, todos os componentes ativos da rede respondem a requisição para o IP que foi forjado no pacote, causando assim uma possível negação de serviço.

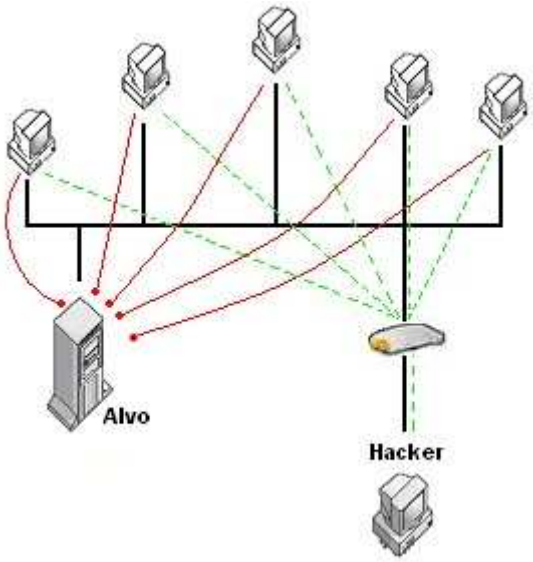

Fig3. Ataque DoS Smurf

Fonte: (http://www.microsoft.com **)**

Hoje esta técnica cedeu lugar a outra mais potente e destruidora demoninada *DDoS*: nesta o ataque é efetuado simultaneamente a partir de vários computadores previamente invadidos e apropriados, sem que o proprietário tenha conhecimento, tornando o sistema atacado lento ou indisponível com maior eficiencia. Ataques *DDoS* podem ser iniciados manualmente, ou mesmo através de vírus/*worms* contendo em seu código instruções de ataques programados.

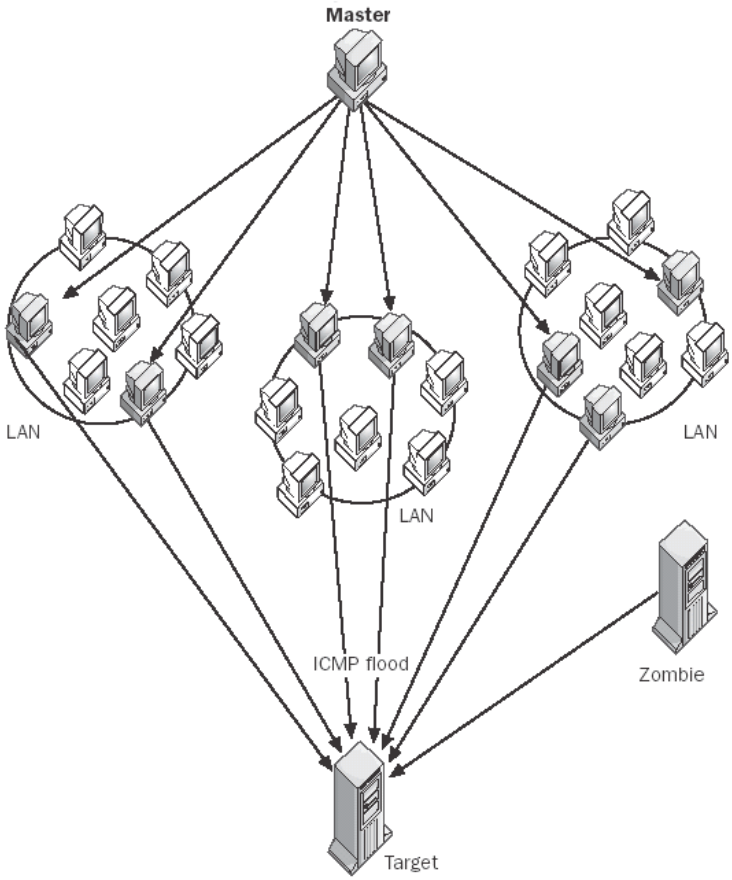

Fig2. Ataque DDoS

Fonte: (http://www.microsoft.com **)**

2.5.3. Ataques de Comprometimento

Segundo MARCELO e PITANGA (2003), HIJAZI (2004), esses tipos de ataques tem a finalidade de comprometer o funcionamento dos dispositivos de uma rede através da desativação de serviços críticos em servidores, comprometimento ou destruição de informações do alvo, desperdício de recursos ou até mesmo comprometer fisicamente os recursos de um sistema.

Os ataques de compromentimento podem ser exemplificados por pixações de *sites,* destruição intencional de dados, suspensão dos recursos do sistema como: processamento, memória.

3. Componentes de Segurança

3.1. IDS

Componente de captura e análise de pacotes que trafegam na rede baseando-se no comportamento pré-definido e eventos maliciosos conhecidos como assinatura, fazendo a identificação de uma forma mais fácil e mais precisa de qualquer tipo de alteração suspeita que ocorra na rede. O posicionamento de um *IDS* na rede deve ocorrer de forma estratégica a fim de aumentar a segurança da empresa. Ao implementar um *IDS*, todo o tráfego na rede pode ser observado e analisado.

Podemos implementar vários *IDS* em uma mesma rede. Os sensores *IDS* são instalados a medida que precisamos aumentar a proteção em determinado ponto de uma rede. As informações coletadas pelos sensores podem ou não ser armazenadas em um gerenciador central de *IDS*.

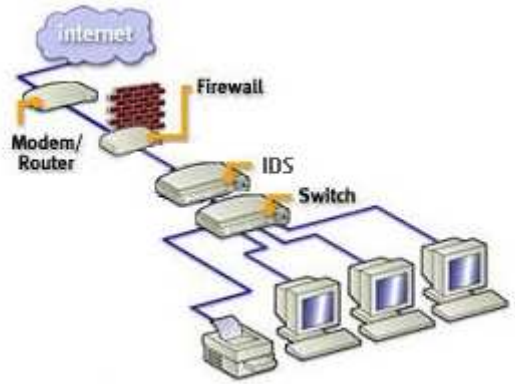

Fig3. Esquema IDS

Fonte: (http://www.fastlanetek.com/ **)**

Um *IDS* pode nos proporcionar a seguintes informações:

- ✓ Quantidades de tentativas de ataque;
- ✓ Tipo de ataques utilizados;
- ✓ Origem dos ataques.

3.2. Firewall

Segundo OLIVEIRA (2002), o *Firewall* é uma ferramenta de segurança de redes que funciona como uma barreira de proteção que controla o tráfego de dados entre a rede local e a Internet. Atua como defesa de um computador por meio de regras e filtragem de dados. Pela lógica, um *Firewall* separa, restringe e analisa datagramas *IP* que passam por ele. Sua implementação pode variar de um *hardware* dedicado até a combinação de alguns elementos de rede distintos.

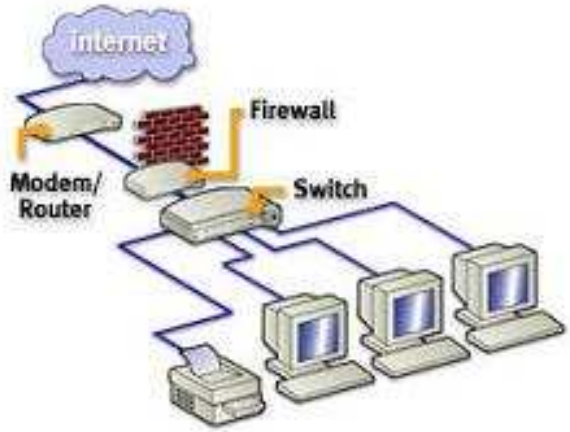

Fig4. Esquema Firewall

Fonte: (http://www.fastlanetek.com/ **)**

Um *Firewall* é um equipamento que faz a intercomunicação e a filtragem de dados entre redes distintas, trabalha com bloqueio a portas de comunicação e em alguns casos pelo conteúdo dos pacotes.

4. Introdução ao Honeypot

4.1. Definição

Segundo MARCELO e PITANGA (2003), *Honeypots* são ferramentas utilizadas para a monitoração de ataques, colecionando informações importantes sobre tendências e permitindo aprimorar a metodologia utilizada para a segurança de uma empresa. O primeiro passo para entendê-los é defini-los. Ao contrário dos *Firewalls* ou *IDSs*, um *Honeypot* não resolve um problema específico de segurança, porém ele é utilizado para ajudar a detectar onde está o problema que deve ser sanado de forma

adaptável e flexível. E é esta flexibilidade que o torna uma ferramenta super poderosa para estudo constante do ambiente no qual foi implementado.

Umas das grandes vantagens de um *Honeypot* é que ele não depende de assinaturas, algoritmos ou regras. Ele simplesmente captura os ataques a ele direcionados, gerando *logs*, que facilitam a análise e a obtenção de uma solução correlata.

Uma desvantagem de se ter um *Honeypot* é o fato de que, quando atacado, ele pode ser usado como ponte para a sua rede, comprometendo a rede real da empresa. Esta desvantagem deve ser observada e estudada, de maneira que combinada com outros recursos de segurança, não permitam que, quando comprometido, o atacante tenha acesso à real rede da empresa.

4.2. Histórico

Segundo MARCELO e PITANGA (2003), o *Honeypot* teve início em 1991 com a publicação do artigo "*The Cucko's Egg*" de Cliford Stool, astrônomo do Laboratório *Lawrence de Berheley*. Durante 10 meses (1986/87), Cliford Stool localizou e encurralou o *hacker* Hunter e, em outro famoso artigo, "*An Evening With Berferd*", onde Bill Cheswicks estudou durante meses as técnicas e criou armadilhas para o *hacker* Berferd, que através de um *bug* no *sendmail* obteve as senhas do sistema. Este artigo é de um grande valor, já que a idéia por trás dos *honeypots* começou a ser desenhada ali.

Em 1992 o especialista Bill Cheswick explicou no artigo "*An Evening With Berferd In Which a Cracker is Lured, Endured and Studied*" os resultados do

acompanhamento de invasões de um dos sistemas da *AT&T,* projetado especialmente para este fim.

Em 1997 Fred Cohen lançou o DTK, ou *Deception Toolkit,* o primeiro *honeypot* que era aberto e gratuito.

Em 1998 surgiu o primeiro produto, o *Sting,* da antiga empresa *Cybercop,* adquirida pela NAI no final deste mesmo ano.

Ainda em 1998, Martin Rasch, criador do *Snort*, desenvolveu um *honeypot* para o governo americano.

Em 1999, surgiu o *Honeynet Project*, criado por Lance Spitzner em uma entidade formada por cerca de 50 especialistas de segurança. Um dos resultados foi o lançamento do *Honeyd,* uma ferramenta com solução em *software* livre. Este foi o grande passo que ganhou repercussão mundial ao demonstrar a importância do estudo do comportamento dos invasores de uma rede para o desenvolvimento de novas ferramentas e sistemas de defesa.

4.3. Tipos

Os *Honeypots* são classificados em duas categorias: produção e pesquisa.

Honeypots de produção são utilizados para distrair a atividade maliciosa de máquinas com maior valor na rede ou como um mecanismo de alerta. Em contrapartida, *Honeypots* de pesquisa são utilizados para a monitoração de um ataque com o objetivo de capturar o maior número de dados possíveis para posterior análise.

4.4. Níveis diferenciados de interação

Os *Honeypots* também são diferenciados por seus níveis de interação: baixa e alta.

Honeypots de baixa interação tem como função principal emular sistemas e serviços. Suas características são:

- ✓ Facilidade de instalação e configuração;
- ✓ Menor risco, pois a emulação de serviços permite o controle do que pode ou não pode ser feito;
- ✓ Captura quantidade limitada de informação, dados transacionais e algumas interações limitadas.

Honeypots de alta interação, não emulam sistemas, e sim usam sistemas reais e utilizam seus serviços. Suas características são:

- ✓ Captura maior quantidade de dados, incluindo novas ferramentas de comunicação;
- ✓ Podem ser complexos de se instalar e configurar (versões comercializadas tendem a ser mais simples);
- ✓ Maior risco, pois os atacantes terão um sistema operacional real para interagir.

4.5. Introdução as Honeynets

4.5.1. Definição

As *Honeynets* consistem em um conjunto de *Honeypots* agregados, formando uma rede virtual de computadores para ser comprometida. Uma vez comprometida, é utilizada para observar o comportamento dos invasores, possibilitando a realização de análises detalhadas das ferramentas utilizadas, das motivações dos invasores e das vulnerabilidades exploradas.

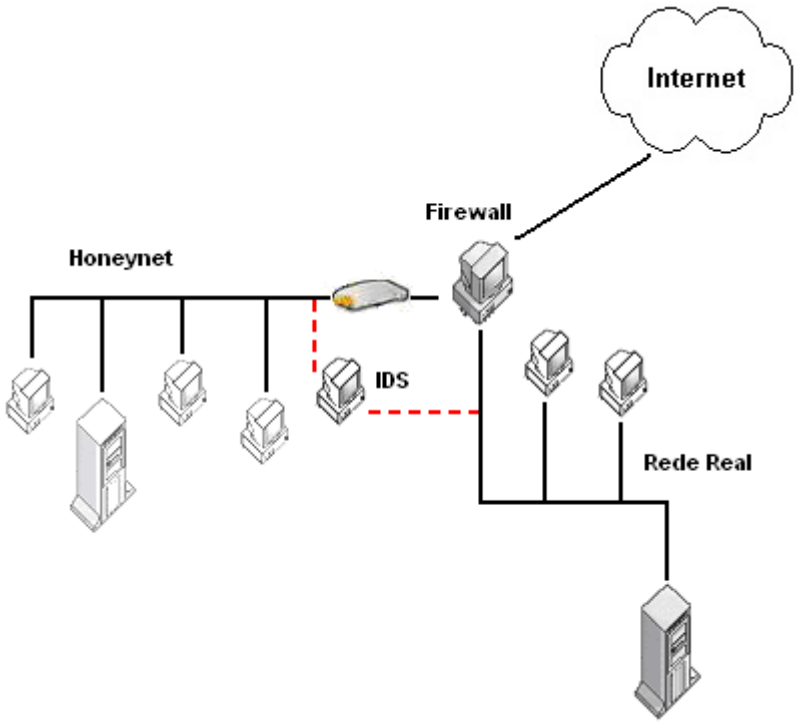

Fig7. Esquema Honeynet

Fonte: (http://www.microsoft.com/ **)**

O sucesso de uma *Honeynet* depende diretamente do controle e captura dos dados gerados pelos ataques. Qualquer falha nestes requisitos implica em uma falha geral.

O controle de dados é o controle de toda a atividade do invasor, desde o início do ataque até o fechamento da conexão. Quando se está trabalhando com invasores, sempre existe risco, e nós devemos minimizá-lo. Deve-se garantir, principalmente, que, uma vez comprometido um *Honeypot* não possa ser usado para danificar qualquer outro sistema. Todavia, o desafio é controlar o fluxo de dados de uma maneira transparente, sem que os invasores desconfiem que estão sendo logados, o que resultaria na desistência do ataque pelo invasor, gerando o fim da captura dos dados para o estudo do ataque referido.

As *Honeynets* são classificadas em dois tipos: Clássica e Virtual.

A *Honeynet* clássica caracteriza-se pelo uso de sistemas reais (físicos e lógicos), com a liberdade de opção de sistemas operacionais e plataformas variadas e independentes.

Vantagens:

- ✓ Dispositivos reais;
- ✓ Maior segurança devido à descentralização do *Honeynet.*

Desvantagens:

- ✓ Maior dificuldade para instalação e gerenciamento;
- ✓ Custo elevado;
- ✓ Complexidade de manutenção.

Com o mesmo objetivo temos as *Honeynets* virtuais, que emulam serviços e sistemas operacionais simulando uma cadeia de computadores em uma rede emulada.

Vantagens:

- ✓ Maior facilidade no gerenciamento;
- ✓ Custo reduzido;
- ✓ Maior facilidade para a instalação e gerenciamento;

Desvantagens:

- ✓ Limitação de realidade dos serviços emulados;
- ✓ Centralização do processamento em um único computador;
- ✓ Segurança do *software* emulador de sistemas e serviços.

4.5.2. Utilização

As *Honeynets* são usadas como fonte principal de captura de dados para estudos e reparos específicos em cada ambiente proposto, e com isso, aprender comportamentos, táticas, ferramentas utilizadas nos ataques e até mesmo os diferentes tipos de atacantes recebidos pelo sistema.

Umas das práticas comumente utilizadas é a repetição da configuração da empresa contendo os mesmos sistemas e aplicativos que são usados no modo de produção, para que os resultados possam ser mais proveitosos e para que a análise reflita sobre a realidade encontrada e possa ser aplicado de forma focada.

Todo o tráfego de uma *Honeynet* gera registros para serem estudados, pois qualquer conexão efetuada é monitorada, facilitando a detecção de riscos e não ocasionando nenhum impacto para a rede corporativa em uso. Com as *Honeynets* podemos aprimorar a capacidade de detecção, reação, recuperação e análise dos sistemas. Todas as técnicas utilizadas são submetidas a constantes análises após cada ataque realizado para que possam ser aperfeiçoadas.

4.5.3. Funcionamento

Para iniciar os testes em uma *Honeynet* devemos adicionar alguns *honeytokens* como contas de usuários, enviar *e-mails* entre eles, criar documentos falsos em alguns diretórios, estabelecer conexões como *HTTP* ou *FTP*, executar alguns comandos que serão armazenados no histórico; tornando a rede aparentemente ativa, o que certamente atrairá mais atenção dos atacantes.

Neste ponto a *Honeynet* já está populada e dispõe de iscas para que se tenha a atenção voltada à ela. Neste momento os dispositivos de captura de *logs* devem estar prontos para capturar os dados oriundos de uma posivel invasão em qualquer um dos computadores da *Honeynet* e armazená-los em um banco de dados para posteriormente gerar relatórios.

4.6. Introdução aos Honeytokens

4.6.1. Definição

Segundo BARROS (2004)**,** *Honeytokens* são informações disponibilizadas nas *Honeynets* que ajudam o trabalho do atacante, ou seja, ele pensa que está conseguindo efetuar um ataque, mas o recurso utilizado foi disponibilizado propositalmente para que isso aconteça. Um *Honeytoken* pode ser caracterizado por uma conta de usuário com senha fraca ou com falsos privilégios, uma vulnerabilidade do sistema operacional ou até mesmo uma abertura em um banco de dados.

Qualquer uso destas informações disponibilizadas pelos *Honeytokens* é considerado uma atividade não autorizada. Está técnica é relativamente simples e mais utilizada para ataques internos.

Para o funcionamento correto de um *Honeytoken*, a *Honeynet* deve estar configurada de forma que qualquer tentativa de utilização dos recursos providos por *Honeytokens* devem ser monitorados e registrados para posterior análise.

Ao manter-se uma conta, aparentemente privilegiada, na base de usuários, cria-se uma situação extremamente atraente para o invasor. Aqueles que executam testes de invasão de redes sabem que um dos estágios mais comuns de uma invasão é a tentativa de conquistar acesso privilegiado. É natural então que o invasor, ao encontrar uma conta com estas características, tente utilizá-la, confiando na facilidade com a qual se encontram contas com senhas fracas. Se os sistemas de monitoração e detecção de intrusos estiverem preparados, o uso (ou tentativa – não é necessário permitir que ela seja utilizável, mas pelo menos dar a impressão de que isso é possível) de tal conta pode ser imediatamente detectado, ajudando a identificar a sua presença no ambiente. Na maioria dos casos outras informações valiosas também podem ser obtidas, como o endereço de origem do ataque e o objetivo do invasor.

Muitos tipos de informação podem ser utilizados como *honeytokens.* Redes bem estruturadas, com políticas de acesso a dados bem implementadas, são as mais beneficiadas desta técnica. Quando você consegue dizer com mais precisão o que é correto, é mais simples identificar o que é incorreto. Se o banco de dados, por exemplo, só pode ser acessado por *Stored Procedures* previamente definidas, podem ser inseridas nas tabelas linhas que agirão como *honeytokens*. Devidamente descartadas pelas *procedures*, tais linhas seriam prontamente detectadas no caso de um acesso direto à tabela.

Uma das vantagens dos *honeytokens* é que eles não dependem de uma tecnologia específica. Praticamente qualquer repositório de informações com recursos de rastreabilidade pode abrigar *honeytokens*. Sistemas de arquivos (como o *NTFS* do *Windows*), diretórios (como o *NDS* ou o *Active Directory*) e bancos de dados são exemplos claros de veículos para o uso de *honeytokens.* Um arquivo chamado "senhas.txt" é extremamente atrativo para o *hacker*. É essa atração que pode acabar por denunciá-lo.

4.7. Captura e análise dos dados

4.7.1. Logserver

Segundo HINES (2002), um *logserver* é nada mais do que um sistema configurado para prover espaço em disco para outros sistemas registrarem os seus registros.

O *logserver* é o componente de maior importância de uma *Honeynet*. Nele são armazenadas todas as informações provenientes de ataques sofridos pelos *Honeypots*. Estas informações são desde o tipo de ferramentas utilizadas para

concretizar o ataque, os comandos inseridos pelo atacante, o tempo de duração do ataque, até as marcas e o caminho percorrido pelo atacante dentro da *Honeynet*. Estas informações são configuráveis e variam dependendo do *Honeypot* implementado. Como o *logserver* armazena as informações necessárias para o sucesso da implementação de uma *Honeynet*, deve ser montada uma estrutura estratégica de modo que o *hacker* não possa comprometer os dados neles armazenados e, assim, invalidar as informações coletadas para análises da invasão. Os dados enviados para o *logserver* poderão ser utilizados para reconstruir os eventos registrados na rede monitorada e, por isso, deve ser armazenar de forma segura e confiável os dados obtidos.

O *logserver* é o primeiro ponto que um atacante procura em uma invasão, para poder substituir os *logs* armazenados ou até apagá-los, por isso a estratégia de implantação de um *logserver* deve ser bastante estudada. Para segurança, algumas *Honeynets* possuem mais de um *logserver*. Se um for deles for descoberto e atacado, o outro ainda armazenará os dados para o estudo do ataque a *Honeynet*, e até ao ataque do outro *logserver*.

Para aumentar a segurança dos *logservers* também podemos utilizar *logservers* remotos. Estes recebem da mesma maneira todas as informações geradas na *Honeynet* e são de mais difícil identificação. Os *logservers* remotos podem ser identificados com a utilização de *sniffers*.

Podemos utilizar outras ferramentas de segurança como o *Firewall* ou o *IDS* para aumentarmos a segurança dos *logserver*, porém esta utilização não deve inibir a atividade do atacante. Se descoberta, o atacante saberá que está em um ambiente controlado e pode não proceder com o ataque.

4.7.2. Captura dos dados

Segundo HIJAZI, MAZZORANA & ROVANELLO (2004) e ROJAS (2003), todo e qualquer dado, recolhido por um *honeypot* dentro de uma *Honeynet* comprometida, deve ser armazenado para uma análise posterior.

Os dados capturados não devem ser armazenados somente no sistema comprometido mas também em um sistema confiável e seguro onde o atacante não possa ter acesso, porém se o mesmo tiver acesso a esses dados que são provenientes de seu acesso a um *honeypot*, a rede pode ser comprometida e esses dados podem ser modificados ou destruídos.

Para fazer a captura de dados em um *honeypot* é necessário uma junção de alguns dispositivos de controle de acesso, como o *firewall* ou roteador, que registram as ocorrências em arquivos de *log*, que neste caso são sempre suspeitas.

A captura de dados pode ser dividida em camadas que podem ser:

- ✓ Camada de controle de acesso: é a primeira camada da captura de dados, podendo ser um *firewall* ou um roteador. Como dito anteriormente, ele manda alertas para *e-mails* do administrador de tudo que entra e sai da *honeynet*. Como todo tráfego da rede *honeynet* é suspeito, esse tráfego é controlado e registrado pelos dispositivos de controle de acesso. Esses alertas são importantes para comparações futuras com o *IDS* (Camada de Rede). O problema desta camada é que não registra a atividade dentro do *honeynet*, apenas o tráfego que passa pelo dispositivo de controle de acesso.

- ✓ Camada de Rede: está relacionada a captura e análise dos pacotes que trafegam na rede. Um *IDS* pode ser utilizado para identificar e capturar todas as informações úteis dos pacotes na rede, usando a análise de assinaturas de pacotes ou detecção de anomalias e alertar o Administrador sobre qualquer atividade suspeita. As informações dos ataques, dos pacotes e até das teclas pressionadas pelo atacante devem ser armazenadas de modo que seja fácil uma análise posterior.

- ✓ Camada de Sistema: a captura do pressionamento das teclas pode se tornar difícil caso o atacante esteja utilizando comunicação criptografada, esse é apenas um exemplo, porém neste caso deve-se fazer a captura das atividades do sistema remotamente, pois logo são apagados após um ataque. Como o *IDS* já capturou os *logs* que trafegam até o servidor, mesmo que o sistema seja comprometido o processo de análise não será afetado.

- ✓ Camada *Off-Line*: nesta camada é feita uma imagem do *honeypot* antes de ser ativado. Com isso, é feita a análise através da comparação da imagem anterior com a atual identificando as alterações que ocorreram durante o ataque. Segundo HIJAZI, MAZZORANA e ROVANELLO (2004), o *Tripwire* é uma ferramenta muito utilizada neste caso, pois faz a imagem do sistema antes de ser comprometido e a compara com o estado do após o ataque, que pode identificar os binários do sistema e os arquivos de configuração que foram modificados através do seu bando de dados.

4.7.3. Controle dos dados

Após o comprometimento de uma *Honeynet*, o fluxo dos dados deve ser controlado sem que o atacante perceba, e garantir que outras redes fora da *Honeynet* não sejam atacadas através deste sistema.

O controle do acesso às informações pode ser feito através de um *firewall* que faz com que os dados que entram e saem da *Honeynet* passem pelo mesmo.

O *firewall* também separa em uma *Honeynet* três redes distintas: a rede externa, que é a Internet de onde provêm os ataques; a rede com os *honeypots*; e uma administrativa que é uma rede confiável onde são concentrados os dados remotamente. Desta forma deve-se definir que qualquer conexão da Internet para *Honeynet* é autorizada, para que qualquer pessoa possa varrer os sistemas conectados a rede. Para não permitir que sejam efetuados ataques na rede externa, as conexões de um *honeypot* com a Internet devem ser controladas pelo *firewall,* e as conexões da rede administrativa não devem possuir nenhuma comunicação direta com a *Honeynet*, para que após a invasão de um *honeypot* não haja tentativas de se invadir a rede administrativa.

Todas essas regras podem ser alteradas, e pode-se optar pelo o mesmo sistema de filtragem utilizado pela produção da organização ou qualquer outra regra de conhecimento do administrador da rede.

Quanto maior o número de conexões, maior será o risco. Neste caso a quantidade de conexões a partir de um *honeypot* deve ser controlada.

Se as conexões forem ilimitadas, o sistema comprometido corre o risco de ser usado para atacar outros sistemas, logo se a quantidade for limitada o *honeypot* deixará de ser útil, pois os atacantes ficarão desconfiados e descartarão a rede projetada.

De acordo com ROJAS (2003), a organização *Honeynet Project* é o órgão mais conhecido desse tipo de pesquisa, e permite de cinco a dez conexões por dia. Já HIJAZI, MAZZORA & RAVANELLO (2004), um número de cinco conexões em um dia é um número adequado, o que permite que o *hacker* tenha flexibilidade pra *downloads* de suas ferramentas, enviar *e-mails*, usar o *IRC* (*Internet Relay Chat*) para comunicação ou qualquer serviço.

O controle de acesso de *Honeynet* para a Internet deve ser feito através de meios automatizados implementados para:

- ✓ enviar alertas ao administrador;
- ✓ controlar e bloquear quando o número de conexões for excedido;
- ✓ controlar o ataque on-line.

Intervenções humanas podem ser um grande risco, quando o *firewall* estiver configurado para mandar alertas via *e-mail*, os mesmos podem não ser enviados por alguma falha de *DNS (Domain Name Server)* ou não serem lidos. Essas regras devem ser controladas através de um *script* específico que bloqueia as conexões quando o limite definido é atingido, podendo mesmo assim enviar avisos para o administrador da rede.

Uma *Honeynet* deve conter um sistema *anti-spoofing* adequado, permitindo a saída apenas de pacotes válidos, diminuindo o risco do uso da rede para ataques de recusa de serviço (DoS).

Um roteador posicionado entre o *firewall* e a *Honeynet* pode ser utilizado para mascarar o *firewall*; controlar *anti-spoofing*, atuando como uma segunda camada de controle de acesso, atuar como *backup* caso ocorra falha no *firewall*; e bloquear grande parte do tráfego inválido, melhorando o desempenho.

4.7.4. Análise dos dados

Segundo ROJAS (2003), esta é a fase mais demorada de uma *Honeynet*, pois nela é necessário transformar as informações em dados relevantes e de fácil entendimento. Porém, de acordo com HIJAZI, MAZZORANA & ROVANELLO (2004), os dados são capturados de forma inteligente, automatizando os processos que os coletam, como por exemplo, um *e-mail* de alerta. Serão examinados os *logs* do *firewall*, alertas de *IDS* e o tráfego capturado, registros do sistema e o pressionamento de teclas.

A análise dos dados pode ser feita das seguintes formas, vejamos:

- ✓ Análise de *IDS*: Segundo HIJAZI, MAZZORANA & ROVANELLO (2004) e ROJAS (2003), o *IDS* captura três fontes de informação, os alertas gerados pelo *IDS*, a captura de tráfego da rede armazenado num arquivo binário e os registros de histórico da sessão ASCII detectados na carga útil do pacote, como o pressionamento de teclas. Pode ser até redundante enviar novamente alertas para os Administradores da rede, mas o *firewall* só informa a tentativa de ataque, já o *IDS* relata o que o invasor está executando, através de um banco de dados que contém informações de como é um ataque e de como ele é detectado, dando informações que permitem a definição de análise mais detalhada nos arquivos de *log* binário, isso é uma vantagem da captura ser feita em camadas.
- ✓ *Logs* de *Firewall*: Todo tráfego em uma *Honeynet* é suspeito e é classificado como um ataque, logo, o número registros gerados é muito grande. Utilizando um *firewall* torna-se possível configurá-lo para enviar alertas para o *e-mail* do administrador quando houver tentativas de invasão ou entrada e saída de

conexões. Porém a análise de registros de um *firewall* torna-se difícil por que o número de acessos é muito grande. É sempre muito importante armazenar toda e qualquer informação manifestada nas tentativas ou caracterização de um ataque, a fim de definir as tendências e direcionar a segurança na rede administrativa, pois um alerta pode evidenciar o que está ocorrendo na rede, proporcionando uma análise mais detalhada no tráfego capturado pelo *IDS*.

- ✓ Registros Históricos de sistema: Em um sistema comprometido, o atacante tentará apagar ou modificar os arquivos de registros do sistema, por isso eles devem ser armazenados em um servidor remoto. Quando um *honeypot* é comprometido, são enviados para o servidor de registro de histórico remoto os dados referentes ao acesso. Esses dados são capturados pelo *IDS* através da rede. Com isso os *logs* de sistema estão em 3 lugares: no próprio sistema, no servidor de arquivos de registro remoto e na captura feita pelo *IDS*. Se o atacante modificar os regitros históricos do sistema, será necessário fazer a comparação entre as informações armazenadas remotamente no servidor de registros históricos remoto e os dados capturados pelo *IDS*.

5. Legalidade dos Honeypots

Segundo HIJAZI, MAZZORANA & RAVANELLO (2004), quando falando sobre legalidade da utilização dos *honeypots*, é necessário considerar três aspectos: armadilha, privacidade e responsabilidade.

- ✓ Armadilha: coagir ou induzir alguém a fazer algo que normalmente não faria, ou seja, instigar a prática de um delito, pode acarretar processo judicial. A

discussão é intensa sobre esse aspecto, mas na maioria dos casos não poderia ser caracterizado crime pelas seguintes razões:

- *honeypot* não induz ninguém, até porque muitas vezes é emulação do sistema de produção da empresa;
- os ataques são por iniciativa do invasor;
- os *honeypots* não estão sendo usados para processar ninguém, e sim como meio para novas descobertas.

✓ Privacidade: o sistema que o atacante está usando não pertence a ele, portanto toda monitoração realizada no sistema não pode caracterizar quebra de privacidade.

✓ Responsabilidade: se o *honeypot* for comprometido e utilizado para prejudicar outras redes pode acarretar processo civil.

Segundo HIJAZI, MAZZORANA & RAVANELLO (2004), juridicamente, em um crime digital, os *honeypots* diminuem a gravidade do crime cometido. Com a invasão de um *honeypot*, diminui a parte danosa do processo cível, porém a premeditação do ato criminoso e sua execução se mantém, visto que o criminoso digital teria efetivamente danificado o sistema real, através do ataque a um servidor qualquer, porém ao fazê-lo contra um *honeypot*, o criminoso afetou apenas um sistema menos importante.

Uma grande vantagem inerente a legalidade do uso de um *honeypot* é que ele se torna uma prova confiável do crime. Primeiramente por armazenar em várias camadas os *logs* de acesso, fazendo com que os registros históricos não sejam

facilmente corrompidos, por permitir alto grau de interação com o criminoso e por ser facilmente removido do ar para análise da invasão.

6. Honeypots comerciais

Alguns *Honeypots* comerciais utilizados no mercado:

- ✓ *NetBait: NetBait* é uma solução de segurança de rede que redireciona ataques contra espaços de *IP* não utilizados para “fazendas de *honeypots*”, anulando o intruso pelo uso da ilusão. O *NetBait* cria esses ambientes projetando um desvio da rede real, assim os nós da rede ficam cercados por múltiplos nós falsos; cada nó falso pode ser configurado com qualquer combinação de sistemas operacionais, serviços e aplicativos. Possui gerenciamento remoto centralizado e configuração de comportamento dinâmico. Classifica-se por baixa interação / produção.

- ✓ *Specter: Specter* foi projetado para ambiente *Windows*, para organizações comerciais de pequeno e grande porte. Pode monitorar até quatorze portas de *TCP*, sendo sete de armadilhas e sete de serviços. As armadilhas bloqueiam e registram as tentativas de ataques; as portas de serviços interagem com o invasor, emulando o aplicativo de acordo com o serviço utilizado. Armadilhas: *DNS*, *IMAP4, SUN-RPC, SSH, SUB-7, BOK2* e genérica. Serviços: *FTP, TELNET, SMTP, HTTP, NETBUS* e *POP3.* Pode emular até quatorze sistemas operacionais diferentes (*Windows* 98, *Window*s NT, *Windows* 2000, *Windows* XP, Linux, Solaris, Tru64 (Digital Unix), NeXTStep, Irix, Unisys Unix, AIX, Maços, MacOS X, FreeBSD). Possui grande variedade de configuração,

características de notificação, banco de dados dos incidentes, além da facilidade no uso. Classifica-se por baixa interação / produção.

- ✓ *KFSensor: KFSensor* é um *IDS* que atua como um *honeypot* utilizado em plataforma Windows, projetado principalmente para proteção. Ele é preparado para bloquear ataques de recusa de serviço e estouro de *buffer*; registra os *logs* do atacante, podendo ser filtrado de várias maneiras para facilitar a análise em determinada porta ou protocolo. É um *honeypot* comercial, mas disponibiliza cópias para avaliação. Classifica-se por baixa interação / produção.

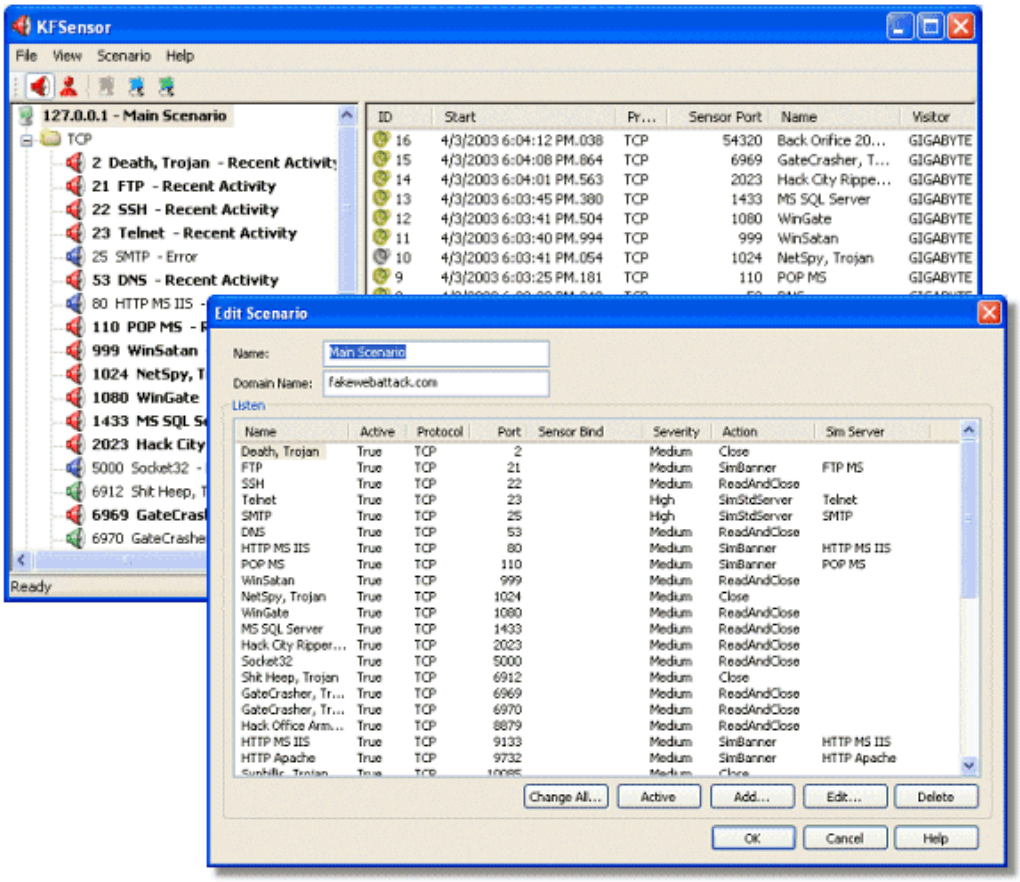

Fig8. Tela principal KFSensor

Fonte: (http://www.theiia.org **)**

- ✓ *PortPeeker: PortPeeker* trabalha escutando portas de comunicação, que usam *TCP/UDP* e armazenam todos os *logs* para as portas solicitadas com alto nível de detalhamento. Provê um relatório de forma clara do *log* de todas as conexões. Classifica-se por baixa interação / produção.

7. Honeypots gratuitos

Existem também alguns *Honeypots* gratuitos e de fácil implementação e administração. Estes podem ser instalados em sistemas *Windows* ou Linux. Selecionamos alguns deles abaixo:

- ✓ *Honeyd*: Emula um pequeno processo que cria *hosts* virtuais em uma rede. Os *hosts* podem ser configurados para rodar serviços arbitrários, e suas 'personalidades' podem ser adaptadas para que pareçam rodar um sistema operacional específico. Melhora a segurança fornecendo mecanismos para avaliação e tratamento de detecção, além de esconder sistemas no meio dos sistemas virtuais.
- ✓ Vanalha: *Honeypot* de simples instalação e fácil administração desenvolvido para ambiente *Windows*. Esta ferramenta se baseia no conceito central do *Honeypot*, mais utiliza uma *engine* central. O programa pode emular: *HTTP, FTP, FINGER, TELNET, SMTP, POP3*, portas *trojan* e alguns serviços.
- ✓ *Honeyperl*: Um *Honeypot* nacional baseado em *Perl*. Aceita a instalação de pluggins para emular programas e serviços do tipo : *WINGATES, TELNET, SQUID, SMTP,* etc.
- ✓ *Joneypot: Joneypot* origina-se de Java *Honeypot*. O objetivo central deste projeto é criar um *Honeypot* multi-plataforma, orientado a objetos e de código

aberto, que foca o desenvolvimento de uma ferramenta de facil utilização, para ser distribuído em diversos sistemas.

8. Estudo de caso

Para nosso estudo de caso utilizamos um dos *Honeypots* citados em nosso projeto. O *KFSensor* é um *Honeypot* comercial, porém, é disponibilizado pela empresa responsável uma versão de avaliação que funciona parcialmente por dez dias. Por cinco noites mantivemos o nosso *Honeypot* on-line para receber ataques e registrar os logs para nosso estudo. No primeiro instante começamos a receber solicitações de serviços. Nosso *Honeypot* estava configurado com o domínio www.cadesi.com.br emulando diversos protocolos e serviços.

Notamos que a maioria dos ataques sofridos ao *Honeypot* eram oriundos de *worms* e *trojans*, ao contrário do que esperávamos. Ataques de *worms* e *trojans* são diretos e não se modificam, ao contrários dos ataques efetuados por pessoas.

Cada atacante tem sua maneira e estilo de efetuar um ataque. Conseguimos registrar diversos tipos de ataques aos serviços disponíveis e apresentaremos abaixo alguns deles.

O primeiro ataque é direcionado ao serviço *SMTP*, onde o atacante tenta enviar e-mails de *sites* pornográficos com conteúdo *HTML* e imagens para diversos e-mails caracterizando um *SPAM*. Utilizando regras básicas de comunicação descritos em RFCs na Internet, este ataque é caracterizado com severidade média oriundo de um IP da França e todos os e-mails são enviados para e-mails de domínios franceses.

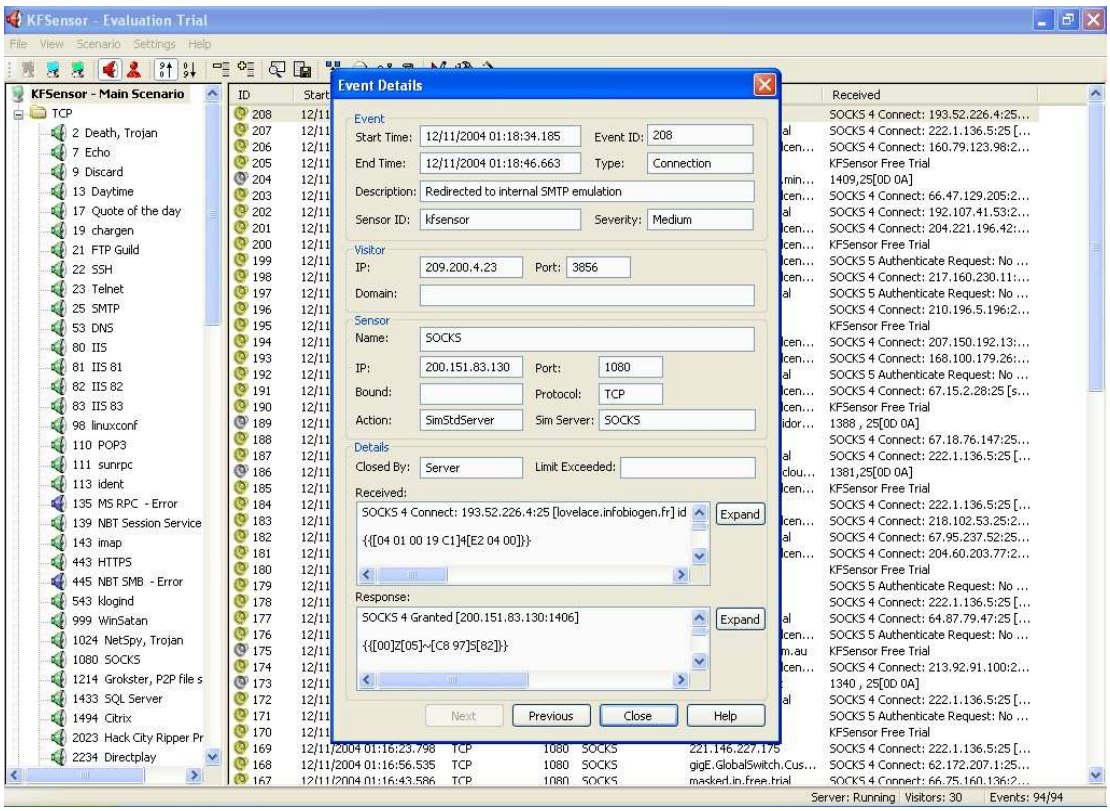

Fig9. Tela KFSensor

Fonte: (Estudo de Caso **)**

SOCKS 4 Connect: 193.52.226.4:25 [lovelace.infobiogen.fr] id [[00]]

{{[04 01 00 19 C1]4[E2 04 00]}}

HELO SHASTA083130.ig.com.br
MAIL FROM: <LRTDSW@compuserve.com>
RCPT TO: <bernot@lovelace.infobiogen.fr>
RCPT TO: <claude@lovelace.infobiogen.fr>
RCPT TO: <clay@lovelace.infobiogen.fr>
RCPT TO: <dessen@lovelace.infobiogen.fr>
RCPT TO: <discala@lovelace.infobiogen.fr>
RCPT TO: <etchever@lovelace.infobiogen.fr>
RCPT TO: <namy@lovelace.infobiogen.fr>
RCPT TO: <pollet@lovelace.infobiogen.fr>
RCPT TO: <pontaro@lovelace.infobiogen.fr>
DATA

Received: from 192.32.206.200 by 200.151.83.130; Fri, 12 Nov 2004 18:59:54 +0400
Message-ID: <MKYODBWDUXZHBFTBETXX@attglobal.net>
From: "Monster Cock" <LRTDSW@compuserve.com>
Reply-To: "Monster Cock" <LRTDSW@compuserve.com>
To: bernot@lovelace.infobiogen.fr
Cc: claude@lovelace.infobiogen.fr, clay@lovelace.infobiogen.fr,
dessen@lovelace.infobiogen.fr, discala@lovelace.infobiogen.fr,
etchever@lovelace.infobiogen.fr, namy@lovelace.infobiogen.fr,
pollet@lovelace.infobiogen.fr, pontaro@lovelace.infobiogen.fr
Subject: SEXUALLY - EXPLICIT: Extreme Stretching
Date: Fri, 12 Nov 2004 17:05:54 +0200
X-Mailer:
MIME-Version: 1.0
Content-Type: multipart/alternative;
boundary="--464020469204468"
X-Priority: 3
X-MSMail-Priority: Normal

----464020469204468
Content-Type: text/html;
Content-Transfer-Encoding: quoted-printable

<HTML>
<HEAD>
<TITLE>All Big Cocks!</TITLE>
<META HTTP-EQUIV=3D"Content-Type" CONTENT=3D"text/html; charset=3Diso-
8859=
-1">
</HEAD>
<BODY BGCOLOR=3D#FFFFFF LEFTMARGIN=3D0 TOPMARGIN=3D0
MARGINWIDTH=3D0 MARGI=
NHEIGHT=3D0>
<FONT FACE=3D"tahoma, verdana, arial"><CENTER>
<!-- ImageReady Slices (fpa01.psd) -->
<TABLE WIDTH=3D600 BORDER=3D0 CELLPADDING=3D0 CELLSPACING=3D10>
<TR>
<TD COLSPAN=3D2 ALIGN=3DCENTER>
<A HREF=3D"http://www.kjomkaasc.biz/fc/bigcocklanding.html">
<IMG SRC=3D"http://kjomkaasc.biz/fc/203.gif" WIDTH=3D447 HEIGHT=3D83 BO=
RDER=3D0></TD>
</TR>
<TR>
<TD ALIGN=3DCENTER><A HREF=3D"http://www.kjomkaasc.biz/fc/bigcocklanding=
.html">
<IMG SRC=3D"http://www.kjomkaasc.biz/fc/204.jpg" WIDTH=3D140 HEIGHT=3D1=
78 ALT=3D"Ouch! It hurts!" BORDER=3D0></TD>
<TD ALIGN=3DCENTER><H2>Monster dicks <FONT
COLOR=3D#FF0000>mercilessly a=
buse</FONT> tight tender cunts!</H2></TD>
</TR>
<TR>
<TD ALIGN=3DCENTER>

```
   <H2>Huge black cocks pound <FONT COLOR=3D#FF0000>squealing white bitche=
s!</FONT></H2></TD>
  <TD ALIGN=3DCENTER><A HREF=3D"http://www.kjomkaasc.biz/fc/bigcocklanding=
.html">
   <IMG SRC=3D"http://kjomkaasc.biz/fc/205.jpg" WIDTH=3D178 HEIGHT=3D140 A=
LT=3D"Too big!" BORDER=3D0></a></TD>
 </TR>
 <TR>
  <TD ALIGN=3DCENTER><A HREF=3D"http://www.kjomkaasc.biz/fc/bigcocklanding=
.html">
   <IMG SRC=3D"http://kjomkaasc.biz/fc/206.jpg" WIDTH=3D140 HEIGHT=3D178 A=
LT=3D"Cum-splattered teens!" BORDER=3D0></a></TD>
  <TD ALIGN=3DCENTER>
   <H2>Big fuckpoles <FONT COLOR=3D#FF0000>ram</FONT> into tiny virgins=92=
 <FONT COLOR=3D#FF0000>cum-splattered faces!</FONT></H2></TD>
 </TR>
 <TR>
  <TD ALIGN=3DCENTER COLSPAN=3D2>
    </TD>
 </TR>
  <TR>
  <TD ALIGN=3DCENTER COLSPAN=3D2>
   <H2>See it all at All Big Cocks!</H2>
      </TD>
 </TR>
 <TR>
  <TD ALIGN=3DCENTER COLSPAN=3D2><A
HREF=3D"http://www.kjomkaasc.biz/fc/bi=
gcocklanding.html">
   <IMG SRC=3D"http://kjomkaasc.biz/fc/207.gif" WIDTH=3D336 HEIGHT=3D103 A=
LT=3D"Click here for INSTANT ACCESS!" BORDER=3D0></a></TD>
 </TR>
</TABLE>
<P ALIGN=3D"center"><BR>
<FONT FACE=3D"Arial" SIZE=3D-3><A HREF=3D"http://www.kjomkaasc.biz/go.php"=
>Please remove me of this list</A></FONT>
improve pronounceable eliot bauer blandish princess birdwatch melamine bau=
sch garish crack contradistinct wilkes disruptive emulsify bobbie workshee=
t paternoster cockleshell stonewall brenda duffy bazaar profligate forever=
 insipid walgreen inflate=20<P>trevelyan elijah placid moonlit fatty killd=
eer vivo dewitt pedro ieee magpie cedar carry flotilla nile nolan adventur=
ous=20
</CENTER>
</FONT>
</BODY>
</HTML>

----464020469204468--

.
QUIT
```

Response

SOCKS 4 Granted [200.151.83.130:1406]

{{[00]Z[05]~[C8 97]S[82]}}

220 lovelace.infobiogen.fr Microsoft ESMTP MAIL Service, Version: 6.0.3663.0 ready at Fri, 12 Nov 2004 01:18:40 -0300
250 lovelace.infobiogen.fr Hello [200.151.83.130]
250 2.1.0 LRTDSW@compuserve.com....Sender OK
250 2.1.5 bernot@lovelace.infobiogen.fr
250 2.1.5 claude@lovelace.infobiogen.fr
250 2.1.5 clay@lovelace.infobiogen.fr
250 2.1.5 dessen@lovelace.infobiogen.fr
250 2.1.5 discala@lovelace.infobiogen.fr
250 2.1.5 etchever@lovelace.infobiogen.fr
250 2.1.5 namy@lovelace.infobiogen.fr
250 2.1.5 pollet@lovelace.infobiogen.fr
250 2.1.5 pontaro@lovelace.infobiogen.fr
354 Start mail input; end with <CRLF>.<CRLF>
250 2.6.0 <LOVELACELvx4efKx00008432@lovelace.infobiogen.fr> Queued mail for delivery
250 2.0.0 OK

O segundo ataque é direcionado ao serviço *IIS*. O atacante recebe uma tela padrão Utilizando regras básicas de comunicação descritos em RFCs na Internet, este ataque é caracterizado com severidade média oriundo de um IP da França e todos os e-mails são enviados para e-mails de domínios franceses.

<event sensorid="kfsensor" id="2" type="Connection" action="SimStdServer" name="IIS" simname="IIS" protocol="TCP" severity="Medium">
<start>2004-11-07 14:12:30:822</start>
<end>2004-11-07 14:12:30:872</end>
<client domain="localhost" ip="127.0.0.1" port="1112" />
<host ip="127.0.0.1" bindip="" port="80" />
<connection closedby="Server" />
<recBytes>231</recBytes>
<received size="231" coding="kf">
<![CDATA[GET /pagerror.gif HTTP/1.1%0D%0A
Accept: */*%0D%0A
Referer: http://127.0.0.1%0D%0A
Accept-Language: pt-br%0D%0A
Accept-Encoding: gzip, deflate%0D%0A
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)%0D%0A
Host: 127.0.0.1%0D%0A

```
Connection: Keep-Alive%0D%0A
%0D%0A
]]>
  </received>
  <sentBytes>3052</sentBytes>
  <sent size="1063" coding="kf">
    <![CDATA[HTTP/1.1 200 OK%0D%0A
Content-Length: 2806%0D%0A
Content-Type: image/gif%0D%0A
Last-Modified: Thu, 07 Feb 2002 23:35:44 GMT%0D%0A
Accept-Ranges: bytes%0D%0A
ETag: "d8427cab732a529:43a"%0D%0A
Server: Microsoft-IIS/6.0%0D%0A
Date: Sun, 07 Nov 2004 14:12:30 GMT%0D%0A
Connection: close%0D%0A
%0D%0A
GIF89a$%000%00%B3%FF%00%00%00%00%80%00%00%00%80%00%80%80%0
0%00%00%80%80%00%80%00%80%80%80%80%80%C0%C0%C0%FF%00%00%0
0%FF%00%FF%FF%00%00%00%FF%FF%00%FF%00%FF%FF%FF%FF%FF,%00
%00%00%00$%000%00%00%04%CF%F0%C8I%AB%A5/%EB%FD%0E%FF
w%00a%E6%95%A8&%91%E1%99%A2%072%B6o%EA%C9%ACX%C3%19%FE%
B9%BB%9F%C6%B7%01%06%8B%1B%A2%E9%082*%8D%CC%CE%87%08eV}%
D5%E3%08%C0%EDz%B3G%D9e%04%1E%1E%18%E8%D8%14%C1n%B3%01%
E5%07%02MG#j%08x%AA%CE%C7%EBQ|u~q%81t%83)s%81w/yqrgi%8B%8C%7F
Q%87%95%96!%85%86%93e%9Ah%98%25%89}%9C5%85%A0%25%A6%A4/%A9
%88%94%80%81%A7%99%B0%AA{%B3%AD%8E%0F%AC(%8D%A5%B6%BB%A
E!%A2u%92%A1%C0%1C%9E%82%C5%9D%C8%9F%CA%97%B1%CFS%C6%D1
%1C%BC%D4%CE%D7%C1%D3%D9r%DB%D9%D6%DC%D5%DE%D7%E0%E1C
pn%E9%EA%EB%ECn^%EF%F0%F1%F2%F1%11%00;%00%00%00%00%00%00%
00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%0
0%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
```

%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
%00%00%00%00%00%00%00%00%00%00%00%00%00%00%0D%0A
[... recorded 778 of 2806 bytes...]%0D%0A

Nesta figura podemos ver a tela principal do *KFSensor*. No lado esquerdo são exibidos os protocolos e serviços emulados pelo *Honeypot*, enquanto ao lado direito são exibidas as tentativas de ataques recebidos e registrados pelo *Honeypot*. Ao duplo clique em um determinado evento, são exibidas as informações do ataque.

È possível filtrar os ataques pelos tipos utilizando a árvore exibida do lado esquerdo. Esta mesma árvore numera a quantidade de ataque sofrido por cada serviço que está sendo emulado.

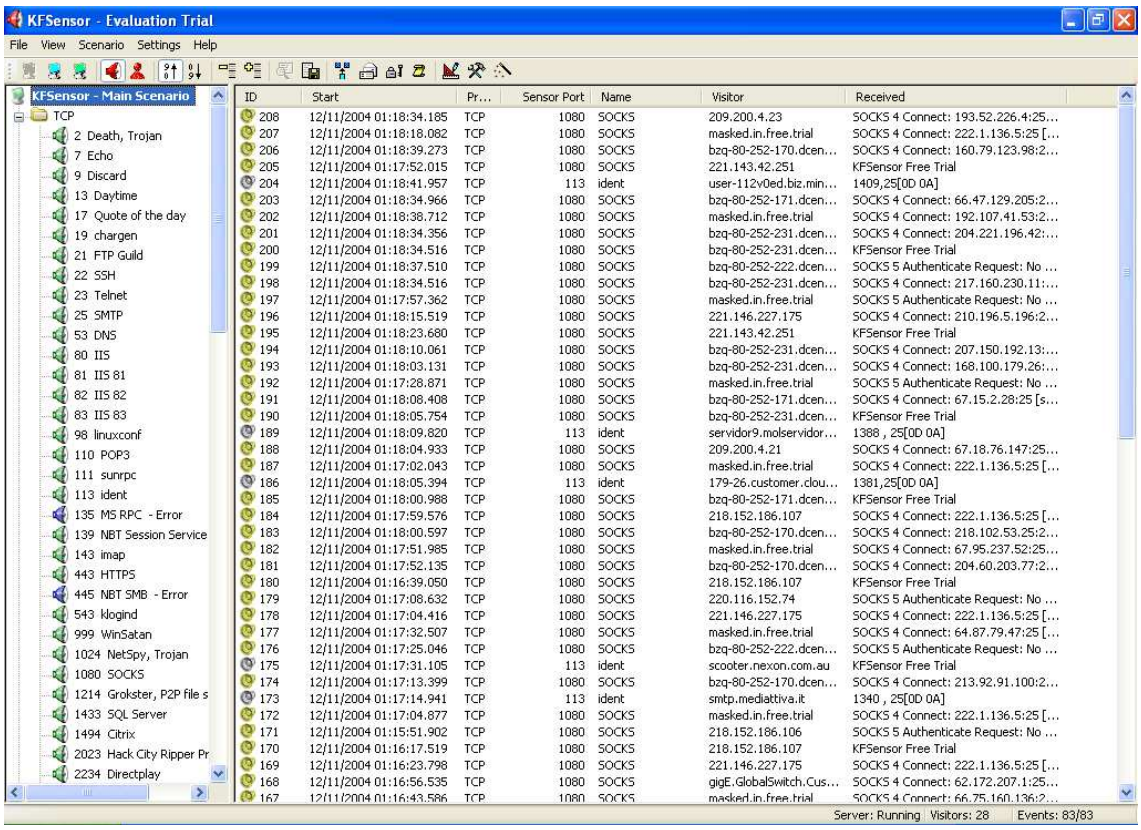

Fig10. Tela KFSensor

Fonte: (Estudo de Caso **)**

10. Referências Bibliográficas

PITANGA & MARCELO (2003), *Honeypot, A arte de iludir Hackers*. Rio de Janeiro, Editora Brasport.

Honeypots: Tracking Hackers. Disponível em: < http://www.tracking-hackers.com >. Acesso em 07/05/2004.

Honeynet.br. Disponível em: < http://lac.inpe.br/security/Honeynet/ >. Acesso em 07/05/2004.

OLIVEIRA, Etienne (2003), *Tópicos Especiais em Redes II, Arquitetura de Segurança, Hackers, Tipos de Ataque*. Página 01 a 05.

ROJAS, Gislaine Aparecida (2003), *Análise de Intrusões através de Honeypots e Honeynets*. Dissertação, Americana, São Paulo.

Honeypots.net, Intrusion Detection, Honeypots and incident Handling Resources. Disponível em: < http://www.honeypots.net >. Acessado em 01/05/2004.

The Honeynet Project, Disponível em < http://www.Honeynets.org/ >. Acessado em 01/05/2004.

Symantec (2004), Honeypots de olho na empresa. Disponível em: < http://www.symantec.com/region/br/enterprisesecurity/content/ >. Acessado em 19/04/2004.

BARROS, Augusto Paes (2004), Honeytokens, o próximo nível dos honeypots. Disponível em: < http://www.csoonline.com.br/AdPortalV3/adCmsDocumentoShow.aspx?documento=25127&Area=11 >. Acessado em 14/08/2004.

Microsoft. Disponível em : < http://www.microsoft.com/portugal/seguranca/artigos/spyware.mspx >. Acessado em 14/08/2004.

Infowester. Disponível em: < http://www.infowester.com/virus.php >. Acessado em 14/08/2004.

HIJAZI, MAZZORANA E RAVANELLO (2004). *Honeypots e aspectos legais*. Dissertação – Instituto de Informática, UCP, Curitiba.

STEIN, L. & STEWART, J. Securing Against Denial of Service Attacks. Disponível em: < http://itpapers.zdnet.com/abstract. aspx?docid=22507&promo=50001 >. Acesso em 16 de Abril de 2003.

HoneypotBr. Disponível em: < http://www.honeypot.com.br >. Acessado em 21/09/2004.

Norton AntiVirus – Symantec. Disponível em: < www.symantec.com.br >. Acessado em 10/10/2004.

McAfee. Disponível em: < http://www.mcafee.com.br >. Acessado em 10/10/2004.

AVG – Grisoft. Disponível em: < www.grisoft.com >. Acessado em 10/10/2004.

HINES, Eric (2002). *Construindo um Log Server utilizando Linux, Unix e Windows.* Disponível em: < http://www.vivaolinux.com.br/artigos >. Acessado em 18/10/2004.