

INSTITUTO DE ESTUDOS SUPERIORES DA AMAZÔNIA  
CURSO DE ENGENHARIA DA COMPUTAÇÃO

Ciro Fernando Preto Borges

Paulo Diego Nogueira Bento

**SEGURANÇA DE REDES UTILIZANDO HONEYPOT**

Monografia de Conclusão de Curso para obtenção  
do Grau de Engenharia da Computação  
apresentada ao Instituto de Estudos Superiores da  
Amazônia.

Orientação Prof<sup>ª</sup> Msc. João Ferreira Santanna

Belém - PA

2006

INSTITUTO DE ESTUDOS SUPERIORES DA AMAZÔNIA  
CURSO DE SISTEMAS DE INFORMAÇÃO

**SEGURANÇA DE REDES UTILIZANDO HONEY POT**

Esta monografia foi julgada adequada para a obtenção do Grau de Engenharia da Computação, e aprovada na sua forma final pelo Instituto de Estudos superiores da Amazônia

DATA: \_\_\_\_/\_\_\_\_/\_\_\_\_

CONCEITO: \_\_\_\_\_

---

Orientador(a)

Profº MSc. João Ferreira Santanna Filho  
Instituto de Estudos Superiores da Amazônia – IESAM

---

Profº MSc. Fabio Cesar Salame

Instituto de Estudos Superiores da Amazônia – IESAM

---

Profº Dr. José Felipe Almeida

Instituto de Estudos Superiores da Amazônia – IESAM

Belém - PA

2006

## **AGRADECIMENTOS**

Agradecemos primeiramente a Deus, que durante toda nossa existência iluminou nosso caminho e sempre dando força para continuar lutando pela vitória e pelo bem, durante nossa caminhada. Aos nossos pais, por todo esforço realizado para que possibilitasse a realização desse sonho. A nossos irmãos e parentes por toda força nos momentos mais árduos dessa jornada. A todos os verdadeiros amigos que sempre estavam de braços abertos para nos acolher e a todos os professores que serviram de fonte de inspiração nesta batalha.

“A grandeza não consiste em receber honras, mas em merecê-las”.  
Aristóteles

## RESUMO

Esta monografia consiste em fazer um estudo sobre ferramentas de segurança honeypot, sendo que o honeypot utilizado foi o Honeyperl.

Os Honeypots são ferramentas que possuem diversas funcionalidades, que podem ser implementadas a partir do objetivo específico que queria alcançar. Honeypots podem ser muito úteis para a segurança de uma rede, pois a ferramenta tem como sua função principal ludibriar um atacante, ou seja, fazer o atacante pensar que está invadindo e utilizando os serviços do sistema sem ser notado, mas na verdade todos os seus passos estão sendo registrados, como por exemplo, tipo de ataque, ferramenta utilizada, IP do atacante e etc. Após a coleta desses dados, será feita uma avaliação das informações capturadas que estão todas armazenadas em log, para que possa ser feito um estudo sobre possíveis vulnerabilidades, para que o administrador possa então melhorar seus métodos de segurança a partir dessas informações coletadas.

Nesse estudo também mostra as vantagens e desvantagens da ferramenta, tipos de Honeypots conhecidos e um pouco da história da ferramenta.

**Palavras-Chave: Honeypots, Segurança de Redes, Honeyperl.**

## ABSTRACT

This monograph has as object to do a study about honeypot's tool of security, the honeypot used was the Honeyperl.

The Honeypots are tools that posses many functions, that can be implement to do a specific object. Honeypots can be very useful in your network, because this tool has as main function to delude a hacker, to do the hacker think that he is invading e using the system's service without someone controlling his actions, but all his actions are resgistried, for example, what kind of attack, the tools used, IP of hacker. After the collect, a valuation is made on the informations in the log file, to valuation what the vulnerability of network, and the administrator can to reinforce the method of security based on the informations collected.

In this study, we approach about the advantages and disadvantages, what kind of honeypot that are famous and a little about the history.

**Key Words: Honeypots, Security network, Honeyperl.**

## LISTA DE FIGURAS

Figura 01: Arquitetura utilizando firewall

Figura 02: Sistema de detecção de intrusos

Figura 03: Arquitetura de uma Honeynet

Figura 04: Honeynet e seu projeto de controle de dados

Figura 05: Honeynet clássica

Figura 06: Honeynet virtual

Figura 07: Honeypot

Figura 08: Localização que um Honeypot pode apresentar

Figura 09: Tela inicial do Honeyperl

## LISTA DE SIGLAS E ABREVIATURAS

<b>IDS</b>	Sistemas de Detecção de Intrusos
<b>IP</b>	Protocolo da Internet
<b>DoS</b>	Ataque de negação de serviço
<b>TCP</b>	Protocolo de controle de transmissão
<b>FTP</b>	Protocolo de transferência de arquivos
<b>NIDS</b>	Sistema de detecção de intrusão de rede
<b>HIDS</b>	Sistema de detecção de intrusão de host
<b>DTK</b>	Posição do comando
<b>DMZ</b>	Zona Delimitarizada
<b>SMTP</b>	Protocolo de transferência de mensagens
<b>HTTP</b>	Protocolo de acesso a WEB
<b>POP3</b>	Protocolo de recebimento de mensagens

## SUMÁRIO

<b><u>SEGURANÇA DE REDES UTILIZANDO HONEYPOT.....</u></b>	<b><u>2</u></b>
---	-----------------



## **1 INTRODUÇÃO**

Nos últimos anos, a segurança no mundo das redes vem ganhando uma grande importância, devido ao crescimento dos sistemas computacionais e ao grande fluxo de informações e sua importância dentro de uma rede, não importando se é de uma grande ou pequena empresa. Além disso, o número de usuários vem crescendo a cada ano que passa, e por muitas vezes, esses usuários podem ser mal intencionados, ou seja, preocupados apenas em roubar essas informações, e com isso lesando outros usuários, portanto aumentando o número de ataques contra redes de grandes e pequenas empresas e instituições. Devido a esse quadro, é de grande importância contar com ferramentas que possam ajudar a manter as trocas de informações dentro de uma relação de confiança entre usuários de uma rede ou até mesmo de redes diferentes.

Honeypots consiste em uma classe de ferramentas que tem como função coletar essas ocorrências, ou seja, ela é intencionalmente vulnerável para que seja atacada. O objetivo maior do uso dessa ferramenta é para fazer a coleta de informações sobre invasões através de simulações de serviços de rede. Com isso, é possível fazer um estudo sobre o comportamento dos atacantes e os tipos de ataques que ele usa. Depois de fazer uma análise minuciosa de tudo que foi coletado, é chegada à hora de definir estratégias de seguranças com o objetivo de dificultar novas invasões a sua rede. Essa ferramenta pode ser usada em ambientes de estudo e também em ambientes de produção, além de ser bastante interessante sua utilização em uma rede corporativa.

## **1.2 JUSTIFICATIVA**

Devido ao crescimento exponencial do número de incidentes reportados ao CERT.br desde do ano de 1999 até 2006, percebemos que cada vez mais empresas e até usuários comuns necessitam ter um pouco mais de informações sobre segurança e as ferramentas que podem auxiliar no combate contra ataques.

## **1.3 OBJETIVOS**

O objetivo desse trabalho é fazer um estudo de caso sobre os Honeypots, com o objetivo de mostrar suas vantagens e desvantagens, e mostrar o funcionamento dos mesmos e os tipos de Honeypots existentes.

Espera-se que os resultados obtidos no estudo dessa ferramenta, possam ser utilizados para soluções de problemas em ambientes de necessitam de segurança.

## **1.4 ORGANIZAÇÃO DO TRABALHO**

Esta monografia está composta por 7 capítulos, descritos sucintamente abaixo:

O capítulo 2 mostra um estudo sobre os tipos de ataques mais utilizados contra um sistema dentro de uma rede.

O capítulo 3 mostra um estudo sobre uma ferramenta de segurança chamada Firewall, que tem como finalidade fazer o bloqueio do que entra e sai da rede, prevenindo a rede de ataques, a partir da política de prevenção utilizada dentro da rede.

O capítulo 4 mostra um estudo sobre uma ferramenta de segurança chamada IDS, onde mostra sua finalidade, que é a de monitoramento.

O capítulo 5 mostra um estudo sobre as Honeynets, que são redes cujo grande objetivo é serem vulnerável a ataques. Esse tipo de ferramenta é projetado com a função de fazer análises sobre ataques para que possam ser feitas medidas preventivas contra futuros ataques.

O capítulo 6 mostra o estudo sobre os Honeypots, que é uma ferramenta de pesquisa que tem como objetivo ser sondada, pois ela irá fazer a captura de informações sobre ataques e atacantes, ele também descreve suas características, funcionamento, tipos, vantagens e desvantagens.

O capítulo 7 mostra um estudo sobre a ferramenta Honeypot chamada Honeyperl, mostrando desde sua configuração até um pouco sobre seu funcionamento.

## **2 FERRAMENTAS E TIPOS DE ATAQUES**

Nos últimos anos cresceu bastante a idéia de segurança, pois o crescimento da complexidade das redes fez com que o desafio de mantê-las seguras fosse ainda maior, portanto há uma grande necessidade de se utilizar ferramentas de segurança para ajudar no controle dos dados de uma rede. Logo, para um administrador é muito importante ter a disposição tais ferramentas que contribuam com a segurança de sua rede, como por exemplo: Firewalls, IDS's (Sistema de Detecção de Intruso) e Honeypots. Porém essas ferramentas sozinhas não tornam sua rede totalmente imune a ataques, pois cada uma tem sua função, o Firewall, por exemplo, faz o controle do que entra e sai da rede, o IDS avisa se a rede está sofrendo algum ataque e o Honeypot funciona fazendo análises de ataques, para depois fazer um estudo sobre os mesmos e evitar futuros ataques. Então, é bastante interessante se essas

ferramentas forem usadas em conjuntos, para que uma possa suprir o ponto fraco da outra e dar mais confiabilidade para a rede.

A segurança apenas ganhou tanto espaço no mundo da computação em virtude do crescimento de ataques contra redes de grandes empresas e instituições, além de ataques contra usuários normais que só utilizam computadores para enviar e-mail e fazer transações envolvendo dados importantes, e que são lesados por fraudes e falsificação de senhas de acesso. No decorrer deste capítulo vamos conceituar os diversos tipos de ataques conhecidos.

## **2.1 TIPOS DE ATAQUES**

Teoricamente podemos classificar os tipos de ataques em 4 grandes categorias de acordo com os resultados que esses ataques produzem (SANTANNA,2006).

### **2.1.1 ATAQUE DE INTERRUPÇÃO**

Os ataques de interrupção de serviço geralmente desativam um ou mais serviços de rede. Se sofrer este tipo de ataque, você pode ser forçado a reinicializar ou reiniciar vários serviços. E, embora isso não seja um risco importante de segurança o tempo de paralisação (downtime) pode ser valioso (SANTANNA,2006).

### **2.1.2 ATAQUE DE INTERCEPTAÇÃO**

Nos ataques de interceptação, o atacante coloca um computador entre dois computadores de uma rede, logo após o computador do atacante se faz passar por um dos computadores originais, permitindo ao atacante conexão funcional com os computadores

originais dando capacidade de ler ou modificarem dados que são trocados entre os computadores originais, enquanto os usuários desses computadores pensam que estão comunicando-se entre si (SANTANNA,2006).

### **2.1.3 ATAQUE DE MODIFICAÇÃO**

Nos ataques de modificação, um terceiro passa a ter acesso não autorizado a um recurso do sistema e modifica o conteúdo da informação ou a configuração do sistema (SANTANNA,2006).

## **2.2 ATAQUE PASSIVO E ATIVO**

Outra classificação muito utilizada é a de ataque passivo e ativo, onde ataques passivos são mais furtivos e difíceis de detectar, os ativos são mais agressivos que podem causar algum dano ao sistema (SANTANNA,2006).

### **2.2.1 ATAQUE PASSIVO**

Esse tipo de ataque é caracterizado pela falta de vigília junto aos recursos, ou seja, neste ataque o atacante fica apenas monitorando as mensagens que circulam na rede, apenas

colhendo informações. Dois tipos comuns de ataques passivos são: análise de conteúdos de mensagens e análise do tráfego da rede (SANTANNA,2006).

### **2.2.2 ATAQUE ATIVO**

Esse tipo de ataque diferentemente do passivo é mais fácil de ser rastreado, pois envolve mudança de dados ou criação de stream não autorizados ou falsificados. Este tipo de ataque é dividido em quatro tipos: mascaramento, replay, modificação de mensagens e Denial of Service (DoS) (SANTANNA,2006).

#### **2.2.2.1 MASCARAMENTO**

Neste tipo de ataque ativo, o atacante simula ser uma entidade autorizada a acessar os recursos de um sistema. Geralmente, ele assume a identidade da máquina de um cliente verdadeiro que tentou se conectar ao sistema, ou seja, ele captura as informações necessárias e faz a entrada no sistema a partir das informações capturadas (SANTANNA,2006).

#### **2.2.2.2 REPLAY**

Neste tipo de ataque ativo, o atacante captura os dados e faz transmissão subsequente para tentar ganhar acesso ao sistema (SANTANNA,2006).

#### **2.2.2.3 MODIFICAÇÃO DE MENSAGENS**

Neste tipo de ataque ativo, o atacante captura os dados e faz alterações ou reordenar os dados para com isso ganhar acesso não autorizado ao sistema (SANTANNA,2006).

#### **2.2.2.4 DENIAL OF SERVICE (DOS)**

Neste tipo de ataque ativo, o atacante realiza uma sobrecarga, através de uma quantidade excessiva de solicitações de serviços, com isso inibindo ou impedindo o funcionamento normal do sistema. Isso é feito devido o atacante invadir as máquinas do sistema e instalar nelas programas zumbis, esses programas ao serem acionados bombardeiam o servidor alvo de solicitações e com causando grande lentidão ou até mesmo indisponibilizando qualquer comunicação com esse computador (SANTANNA,2006).

### **2.3 CONCEITOS DE SEGURANÇA**

#### **2.3.1 COOKIES**

São pequenas informações que os sites visitados por você podem armazenar em seu computador (PC). Estes são utilizados pelos sites de diversas formas, tais como (CERT.BR,2005):

- guardar a sua identidade e senha quando você vai de uma página para outra;
- manter listas de compras ou listas de produtos preferidos em sites de comércio eletrônico;



- personalizar sites pessoais ou de notícias, quando você escolhe o que quer que seja mostrado nas páginas;
- manter a lista das páginas vistas em um site, para a estatística ou para a retirar as páginas que você não tem interesse dos links.

### **2.3.2 ENGENHARIA SOCIAL**

O termo é utilizado para descrever um método de ataque, onde alguém faz uso da persuasão, muitas vezes abusando da ingenuidade ou confiança do usuário, para obter informações que podem ser usadas para se ter acesso não autorizado a computadores ou informações (CERT.BR,2005).

### **2.3.3 VULNERABILIDADES**

Vulnerabilidade é definida como uma falha no projeto, implementação ou configuração de um software ou sistema operacional que, quando explorada por um atacante, resulta na violação da segurança de um computador.

Existem casos onde um software ou sistema operacional instalado em um computador pode conter uma vulnerabilidade que permite sua exploração remota, ou seja, através da rede. Portanto, um atacante conectado à Internet, ao explorar tal vulnerabilidade, pode obter acesso não autorizado ao computador vulnerável (CERT.BR,2005).

### **2.3.4 CÓDIGOS MALICIOSOS**

Códigos maliciosos ou malwares (Malicious Software) é um termo genérico que abrange todos os tipos de programa especificamente desenvolvidos para executar ações

maliciosas em um computador. Na literatura de segurança o termo malware também é conhecido por “software malicioso” (CERT.BR,2005).

#### **2.3.4.1 VÍRUS**

São programas projetados com intenções maliciosas e que intencionalmente se multiplicam através de programas ou arquivos. É denominado desta maneira devido sua emulação ser da mesma característica de seu variante biológico: se propaga, vive num ambiente próprio, e move-se através de infecções (GRIFFIN,2000). Por definição, trata de um código de software, ou parte dele, geralmente é escrito em linguagem Assembly, capaz de se propagar em qualquer localização, tipicamente sem o consentimento do usuário e normalmente causando destruição ao sistema computacional onde está hospedado.

#### **2.3.4.2 WORMS**

Os worms possuem as mesmas particularidades de um vírus comum, porém sua diferença está em ter como objetivo a disseminação através da rede de computadores e não entre arquivos em um sistema operacional, como ocorre naturalmente nos vírus eletrônicos.

Os worms são geralmente escritos em linguagens de script (como VBS, por exemplo) e tendem a explorar falhas de serviços, cujos meios de propagação acontecem através delas, visando assim atingir máquinas que não possuem correções de software atualizadas, consumindo seus recursos ou parte deles.

### 2.3.4.3 BACKDOORS

Uma backdoor se trata de qualquer processo referente a uma futura entrada no sistema de forma não autorizada, sendo inserido clandestinamente e criando um acesso na maioria das vezes irrestrito ao invasor, o mesmo pretendendo controlar o sistema conforme sua vontade (PAXSON; ZHANG, 2000).

Existem diversos modelos funcionais. Conforme seu grau de ocultação com relação ao administrador do sistema eles podem variar, tentando “se esconder” para evitar ser rastreada e fechada a porta de conexão para com o atacante. Com isto, existem aquelas que trabalham em user-space (backdoors mais simples) onde se trata apenas de um processo em execução, e kernel-space, sendo um pouco mais complexa e se escondendo como um módulo do kernel do sistema operacional afetado.

### 2.3.4.4 CAVALO DE TRÓIA

Um “cavalo de tróia” – *trojan horse* – faz analogia à história, onde os gregos presentearam os troianos na com um enorme cavalo de madeira, tendo escondido seu exército no interior do cavalo para abrir os portões e fazer à invasão; essas pragas virtuais seguem a mesma filosofia. Trata-se de um arquivo aparentemente inofensivo, porém com um código malicioso inserido em seu contexto, abrindo portas para futuras conexões a invasores, servindo como *backdoor* (GRIFFIN, 2000). São exemplos comuns de *trojans*: SubSeven e NetBus.

A diferença entre as duas entidades está justamente em seus métodos de inserção. No “cavalo de tróia” há o consentimento do usuário administrador do sistema, que na maioria das vezes é o agente responsável pela sua inserção. Já uma *backdoor* é inserida sem o consentimento do administrador de forma obscura após um comprometimento, no qual o atacante queira garantir sua permanência no sistema.

#### **2.3.4.5 SPYWARES**

Recentemente difundido, *spywares* são implementações que buscam em geral comprometer a privacidade do usuário agindo de maneira distinta entre suas variadas versões.

Alguns buscam dados sigilosos através da coleta de informações pessoais, outros tendem a mudar a configuração do sistema operacional afetado sem o consentimento do usuário, atrapalhando a usabilidade de ferramentas e *softwares* (MICROSOFT, 2005).

Explorando falhas de alguma aplicação ou do próprio sistema operacional, os *spywares* geralmente se instalam em *browsers* para mudar a página principal quando o usuário acessa, alguns adicionam barras de busca do próprio fabricante que criou a praga, outros fazem abrir janelas *pop-up* mesmo não estando conectado à internet. Muitos conseguem monitorar toda a atividade realizada na internet, capturando endereços de e-mail, *sites* que estão sendo visitados, senhas, e até número de cartões de crédito, passando as informações para outra pessoa ou companhia (BEAL, 2004).

#### **2.3.4.6 ROOTKITS**

Um rootkit é um trojan que busca se esconder de softwares de segurança e do usuário utilizando diversas técnicas avançadas de programação.

Rootkits escondem a sua presença no sistema, escondendo suas chaves no registro (para que você não possa vê-las) e escondendo os seus processos no Gerenciador de Tarefas, além de retornar sempre erros de “arquivo inexistente” ao tentar acessar os arquivos do trojan.

Diversos trojans utilizam essas tecnologias com o objetivo de dificultar sua remoção e o fazem com sucesso: os rootkits mais avançados são bem difíceis de serem removidos (ROHR,2005).

### **3 FERRAMENTAS PARA PREVENÇÃO E DETECÇÃO DE ATAQUES**

#### **3.1 INTRODUÇÃO**

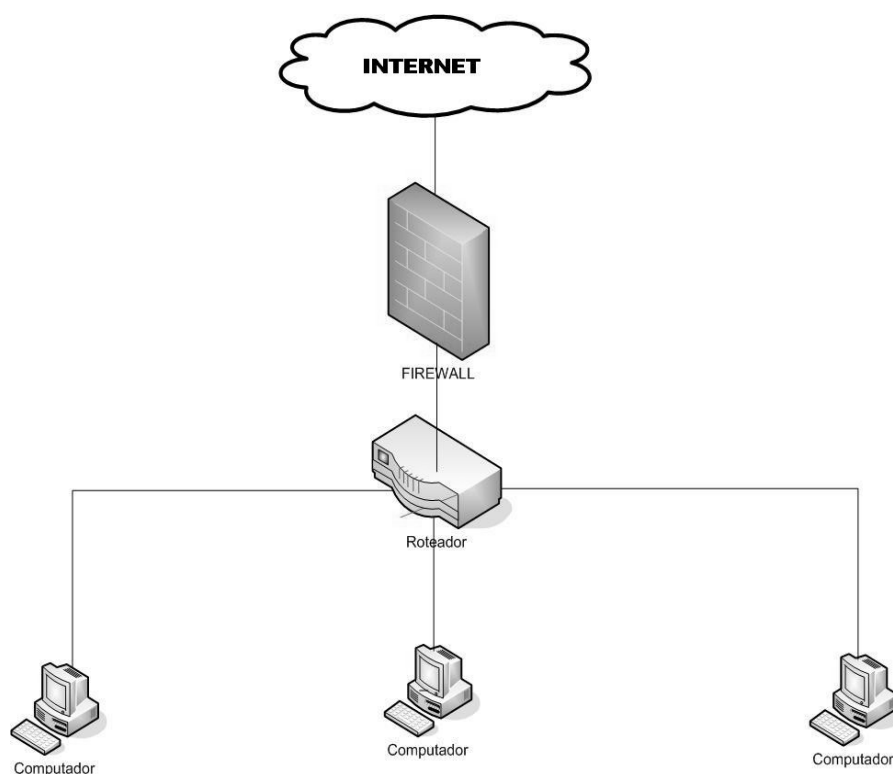
Neste capítulo iremos mostrar a importância das ferramentas de prevenção e detecção de ataques para a segurança de uma rede. Mostrando as utilidades e funcionalidades das duas mais conhecidas ferramentas de prevenção e detecção, que são o Firewall e IDS. Sem esquecer de ressaltar suas vantagens e desvantagens no uso dessas ferramentas.

### 3.2 FIREWALL

Firewall é uma ferramenta de segurança com cada vez mais importância no mundo da computação. À medida que o uso de informações e sistemas é maior, a proteção destes requer a aplicação de ferramentas e conceitos de segurança eficientes (ALECRIM, 2004).

Firewall literalmente, seria uma parede corta-fogo, é como se fosse uma proteção, colocada entre o seu computador e a Internet, tendo como fogo os ataques e outros perigos da Internet, onde o Firewall tem como função bloquear esses perigos (BATTISTI, 2005).

O Firewall tem outras características e funções, como a de ser utilizado para bloquear determinados tipos de tráfegos a partir do seu computador para a Internet. Mais precisamente, o Firewall é um mecanismo que atua como "defesa" de um computador ou de uma rede, controlando o acesso ao sistema por meio de regras e a filtragem de pacotes.



## **Figura 01 – Arquitetura utilizando Firewall**

### **3.2.1 TIPOS DE FIREWALL**

#### **3.2.1.1 FILTRO DE PACOTES**

Filtragem de pacotes é o bloqueio ou liberação da passagem de pacotes de dados de maneira seletiva, conforme eles atravessam a interface de rede.

É muito utilizado em redes pequenas ou de porte médio, por meio de um conjunto de regras estabelecidas, esse tipo de Firewall determina que endereços IPs e dados possam estabelecer comunicação e/ou transmitir/receber dados. Alguns sistemas ou serviços podem ser liberados completamente, enquanto outros são bloqueados por padrão, por terem riscos elevados (ALECRIM, 2004).

Este tipo se restringe a trabalhar nas camadas TCP/IP, decidindo quais pacotes de dados podem passar e quais não. Tais escolhas são regras baseadas nas informações endereço IP remoto, endereço IP do destinatário, além da porta TCP usada (ALECRIM, 2004).

#### **3.2.1.2 FIREWALL DE APLICAÇÃO**

São instalados geralmente em computadores servidores e são conhecidos como Proxy, este tipo não permite comunicação direto entre a rede e a Internet. Tudo deve passar pelo Firewall, que atua como um intermediador, o proxy efetua a comunicação entre ambos os lados por meio da avaliação do número da sessão TCP dos pacotes (ALECRIM, 2004).

Este tipo de Firewall é mais complexo, porém mais seguro, e permite um acompanhamento melhor do tráfego entre a rede e a internet. Sendo assim, deixa claro que

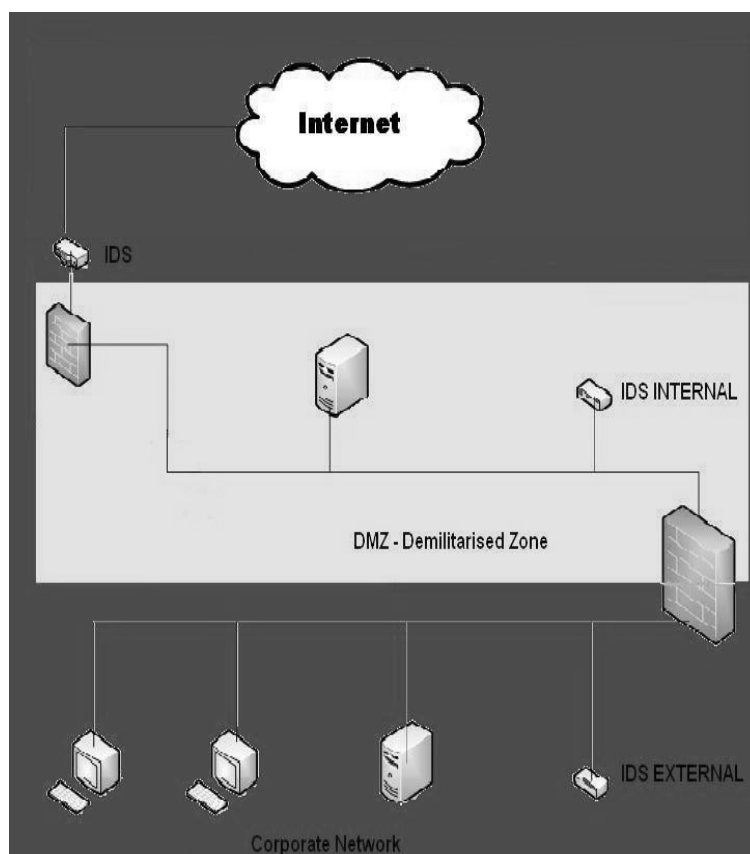
este tipo de Firewall é mais voltado para redes de médio e grande porte, além de ser necessário uma maior experiência na configuração do mesmo (ALECRIM, 2004).

### **3.3 SISTEMA DE DETECÇÃO DE INTRUSO - IDS**

Sistemas de detecção de intrusos são programas cuja função é monitorar uma rede ou um host a procura de sinais padrões de comportamento que sejam considerados maliciosos, ou seja, que podem constituir um ataque de invasão ao sistema ou até menos algum usuário legítimo fazendo mau uso do mesmo.

O IDS é considerado uma ferramenta limitada, pois apenas detecta invasões, apesar dessa limitação, ele funciona como um “alarme” em tempo real muito útil a rede. A ferramenta roda constantemente em background e apenas gera o alarme quando detecta algo suspeito ou ilegal, cujas suspeitas são programadas a partir de ataques previamente já conhecidos pela ferramenta, além disso, a ferramenta possibilita trabalharmos com outras ferramentas de segurança sem interferir negativamente no seu trabalho, mas sim dando uma visão mais ampla do nível de segurança da rede (NED,1999). Na figura 02, podemos observar melhor a arquitetura de funcionamento de um Sistema de Detecção de Intruso, aonde todos os pacotes vindos da rede externa são devidamente direcionados para o sensor IDS interno, o qual poderá gerar um alarme de aviso caso os pacotes determinem um ataque, sendo que essa situação irá ativar o Firewall ou outras ferramentas utilizadas para o combate de ataques maliciosos. O mesmo acontece com o fluxo oposto.





**Figura 02 – Sistemas de Detecção de Intrusos (IDS)**

Quanto ao funcionamento existem dois tipos de IDS, os baseados em sistemas de regras (Rule-based Systems) que usam da base de dados, onde fica todo e qualquer tipo de assinatura dos ataques, e do tipo Sistema Adaptável que possuem técnicas mais avançadas usando desde inteligência artificial até conhecimentos matemáticos e estatísticos.

### **3.3.1 SISTEMAS DE DETECÇÃO DE INTRUSOS DE REDE – NIDS**

Este tipo de IDS é bem mais comum e utilizado. Nele as tentativas de ataque são capturadas e analisadas por meio de vários pacotes de rede, este monitoramento envolve as diversas estações ligadas na máquina que hospeda os sensores IDS.

As grandes vantagens desse modelo é a sua implementação, devido não afetar o funcionamento normal da rede, pois ele atua passivamente na escuta do tráfego. Neste modelo é implantado um conjunto de sensores em vários pontos da rede, realizando assim uma análise local e reportando ataques ao servidor. Com isso se bem posicionados podem monitorar uma grande rede, além de serem invisíveis a muitos atacantes (BACE;MELL,2004).

As desvantagens desse modelo é a grande dificuldade de processar todos os pacotes da rede, se ela for grande ou estiver congestionada, portanto ele pode não reconhecer um ataque em horários de grande tráfego de informações, e a incapacidade de analisar informações criptografadas (BACE;MELL,2004). Além do mais, a maioria deles não pode informar ao gerente da rede se um ataque teve sucesso ou não, ele apenas alerta que um ataque foi feito e cabe ao gerente tomar as providências de investigação para saber o nível de comprometimento dos hosts atacados.

### **3.3.2 SISTEMAS DE DETECÇÃO DE INTRUSOS DE HOSTS – HIDS**

Este foi o primeiro sistema IDS que surgiu, possuindo um funcionamento bem simples de ser implementado e configurado. Sua funcionalidade se baseia no monitoramento de um único host, podendo coletar dados minuciosamente, gerando assim dados ricos em detalhes.

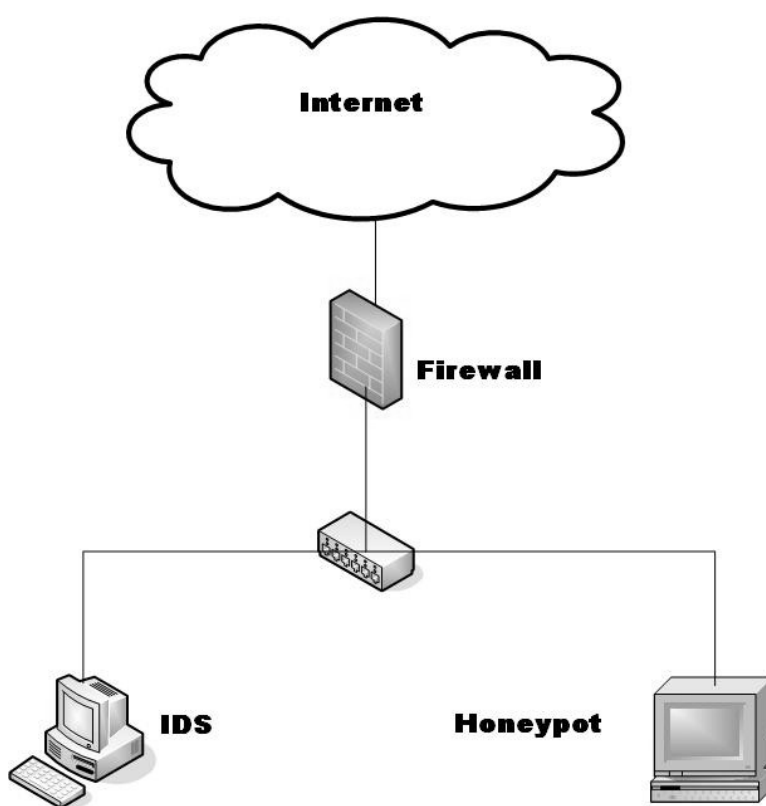
As grandes vantagens dos HIDS se justificam devido à facilidade de se configurar e de uma menor probabilidade de ser descoberto em um eventual ataque, além do mais, ele pode operar em ambientes onde haja tráfego criptografado. As desvantagens são poucas, como por

exemplo, a dificuldade de gerenciamento e a impossibilidade de detecção dos ataques destinados à rede local, pois são apenas monitorados os pacotes recebidos pelo próprio host (BACE;MELL,2004). Além do mais, eles consomem muito dos recursos computacionais do host e com isso diminuindo o desempenho do mesmo.

#### **4 HONEYPOTS**

Nos dias atuais é imprescindível ter uma ferramenta para fazer a análise de ataques dentro de uma rede. Utilizando-se ferramentas desse tipo podemos ter noção dos riscos que corremos ao estamos compartilhando arquivos na rede, pois você pode saber todo o procedimento que ocorrer durante uma invasão e descobrir todas as atitudes tomadas pelo invasor dentro do seu sistema. Os Honeypots é uma ferramenta com uma grande utilidade, podendo ser usada em diversas situações, dependendo da necessidade do usuário. A

ferramenta serve para fazer a análise das atitudes tomadas por atacantes dentro de um sistema, onde após o estudo dessas análises possa ser tomadas medidas que possam prevenir futuros ataques. Geralmente são utilizados em conjunto com outras ferramentas de segurança, como Firewalls que controlam o tráfego de entrada e saída de uma rede e como os IDS (Sistema de Detecção de Intrusos), que são sensores com a função de monitorar o tráfego da rede e identificar atitudes suspeitas. A figura 07 abaixo mostra a arquitetura de uma rede utilizando Honeypot como uma das ferramentas de segurança.



**Figura 07 – Honeypot**

#### **4.1 A HISTÓRIA DOS HONEYPOTS**

Não é de hoje que pesquisadores tentam solucionar o caso da segurança de redes, sendo utilizados mecanismos para observação de atividades de invasores em redes conectadas à internet, esse tipo de medida vem sendo utilizado na prática a algum tempo no mundo da tecnologia da informação.

As primeiras experiências envolvendo mecanismos de acompanhamento de atividades de invasores são de 1988, quando Clifford publicou sua experiência com um ataque chamado “Hunter” no sistema do Lawrence Berkeley Laboratory (LBL). Onde a partir desse momento despertou o seu interesse sobre o assunto, e então decidiu não eliminá-lo do sistema. Clifford ao invés de interromper o ataque, ele tomou a decisão de acompanhar e registrar todos os principais objetivos do ataque, e poder aprender um pouco mais sobre ele. Este acompanhamento levou cerca de um ano e revelou não apenas a origem do ataque, mas também os motivos do atacante e as redes que ele estava interessando em atacar. Esta publicação foi chamada de “The Cuckoo’s Egg” (STOLL, 1990), ela parece e muito com um Honeypot, pois permite o desenvolvimento do ataque para no final obter informações sobre o atacante.

Em 1992, foi à vez de Bill Cheswick de publicar um artigo, cujo nome era “An Evening with Berfered” (CHESWICK, 1990). Neste artigo Cheswick mostra a sua experiência no acompanhamento de uma invasão em um dos sistemas da AT&T, que havia sido propositalmente projetado para ser invadido. Steven Bellovin também participou deste projeto, foi ele que desenvolveu as ferramentas utilizadas como armadilha para capturar as informações dos invasores. Depois de sete anos, Fred Cohen desenvolveu a primeira solução de Honeypot chamada de DTK (Deception Toolkit), ela foi a primeira ferramenta de código aberto com o objetivo de ludibriar atacantes (COHEN, 1997). Esta ferramenta simula vulnerabilidades e coleta as informações sobre os ataques sofridos, ela é muito parecida com o sistema desenvolvido pelo Cheswick.

Alguns anos depois do aparecimento do DTK, foram desenvolvidos os primeiros produtos comercializáveis usando a solução Honeypot, como o CyberCop Sting que rodava no sistema windows NT e poderia ser emulado em vários sistemas diferentes ao mesmo tempo, os outros produtos eram o NetFacade e o NFR BackOfficer Friendly.

## **4.2 VANTAGENS**

Como em toda ferramenta de segurança e no caso do Honeypot não é diferente, existem suas vantagens e desvantagens. Neste tópico iremos mostrar um pouco das vantagens na utilização do Honeypot.

Uma das grandes vantagens da ferramenta é que ela trabalha isolada, ou seja, o fluxo de informações é pequeno se comparado com uma grande rede de produção (ROJAS,2003), além do mais durante a interação com o atacante onde são capturadas todas as informações sobre cada passo do atacante, o Honeypot guarda todos esses dados na forma de registros (logs) com o objetivo de fazer um estudo, pois tais informações têm um grande valor. Como os dados possuem um formato simplificado, isso ajuda para que possa ter um melhor entendimento na hora da análise dos mesmos, facilitando assim medidas preventivas mais eficazes (SPTIZNER,2002).

Outra facilidade da ferramenta é que ela acaba com os falsos positivos gerados pelo IDS, pois neste sistema devido ao grande volume de informações é gerado um número grande de alertas falsos, fazendo com que sejam ignorados para que todas as informações necessárias sejam capturadas. Mas isso não quer dizer que não haverá mais alertas, mas sim uma

diminuição no número dos mesmos, pois no Honeypot todo tráfego é não autorizado e suspeito por natureza, portanto não ocorrerá mais esse tipo de erro e também evitará uma sobrecarga nos componentes. Além disso, o tempo que se levaria para distinguir os ataques reais de informações falsas será minimizado, pois o honeypot não encontra dificuldades com relação à identificação dos ataques e comportamentos suspeitos (ROJAS,2003). Isso, não quer dizer que não se devam utilizar os dois juntos, mas sim fazer a configuração certa para que ambos possam interagir da melhor maneira possível.

Em relação à criptografia usada na segurança das informações de muitas empresas, o IDS encontra uma dificuldade em detectar estes dados, mas o Honeypot consegue detectar, monitorar e capturar os dados dentro da rede até mesmo se estiver criptografados, portanto mostrando mais uma das vantagens dessa ferramenta de segurança (SPTIZNER,2002).

#### **4.3 DESVANTAGENS**

Uma das desvantagens é a visão limitada ao tráfego, pois a ferramenta pode detectar somente os ataques direcionados a ele, sendo assim ignorando as atividades relacionadas a outros sistemas. Portanto, ele não consegue detectar roubo de arquivos confidenciais feitos de usuários da própria rede e não consegue detectar ataques contra seu próprio servidor de serviços da rede, e devido a isso a ferramenta apenas consegue detectar as ações realizadas em sua rota (WEB,2003).

A principal desvantagem é se o inimigo conseguir descobrir qual é a ferramenta, invadi-la e usá-la para prejudicar outros sistemas dentro da rede e até mesmo o atacante manipular as informações, fazendo com que conclusões falsas sejam feitas (SPTIZNER,2002).

É muito importante para a segurança não apenas utilizar o IDS, Firewall e qualquer outra ferramenta sozinha, pois nenhuma ferramenta sempre será 100% confiável, pois cada uma delas tem a sua vulnerabilidade, ou seja, utilizar o Honeypot para substituir essas ferramentas, não é uma decisão muito sábia, a decisão mais correta é utilizá-las sempre uma em conjunto com a outra, pois assim você reforça ainda mais a segurança da rede.

#### **4.4 TIPOS DE HONEYPOTS**

##### **4.4.1 HONEYPOTS DE PESQUISA**

Este tipo de Honeypot é menos utilizado pelos administradores de rede por serem mais complexos e exigirem maior tempo e esforço. Como o próprio nome já diz, seu grande objetivo é de pesquisar as possíveis ameaças, as ferramentas utilizadas num ataque, a origem dele e as ferramentas utilizadas. Esta facilidade em monitorar é devido ao acompanhamento real de cada passo do atacante (SPTIZNER,2002).

##### **4.4.2 HONEYPOTS DE PRODUÇÃO**

Este tipo de Honeypot é mais utilizado do que o de pesquisa, devido à facilidade de sua implementação, apesar de suas limitadas funcionalidades e não armazenarem informações mais precisas sobre as ações dos atacantes. O objetivo geral deste tipo de Honeypot é tentar diminuir ao máximo a incidência de riscos e ajudar é claro na segurança da rede, portanto sua implementação é bastante importante na tomada de algumas decisões em relação a segurança, apesar de este tipo coletar menos informações sobre os passos do atacante (SPTIZNER,2002).



## **4.5 CLASSIFICAÇÃO QUANTO AO NÍVEL DE INTERATIVIDADE**

### **4.5.1 HONEYPOTS DE BAIXA INTERATIVIDADE**

Neste caso, ele se caracteriza por emular serviços e sistemas operacionais, não permitindo assim interação entre o atacante e o sistema (ROJAS,2003). Eles são considerados de fácil instalação e suas funcionalidades são limitadas, já que não pode ser usado para atacar e monitorar outros sistemas, ele apenas controla o que é permitido. Neste sistema ele faz a captura dos dados sobre o ataque como a porta utilizada e não sobre o atacante (SPTIZNER,2002).

### **4.5.2 HONEYPOT DE MÉDIA INTERATIVIDADE**

Nesse nível de interatividade o sistema tolera atacantes que possuem mais habilidades de interagir com o mesmo, logo precisa de um tempo maior para sua implementação, devido ao nível maior de complexidade (SPTIZNER,2002).

### **4.5.3 HONEYPOTS DE ALTA INTERATIVIDADE**

Neste caso, eles são compostos por sistemas operacionais e serviços reais e facilitam que os atacantes interajam com o sistema. Possibilitando uma vasta coleta de dados sobre os atacantes (ROJAS,2003). Devido à complexidade de sua arquitetura, eles levam mais tempo para serem implementado, além de apresentar um grande risco para a empresa, pois oferece acesso livre ao sistema operacional (SPTIZNER,2002).

## **4.6 CLASSIFICAÇÃO DOS HONEYPOTS BASEADOS NA IMPLEMENTAÇÃO**

### **4.6.1 HONEYPOTS REAIS**

Este caso é emulado serviços reais, usados para obter uma melhora no nível de geração de alertas e históricos de registros do sistema.

### **4.6.2 HONEYPOTS VIRTUIAIS**

Este tipo de implementação fornece um software que emula determinados serviços e servidores. Pode ser utilizados para abrirem portas de conexão e responderem a requisições externas para o estabelecimento da conexão.

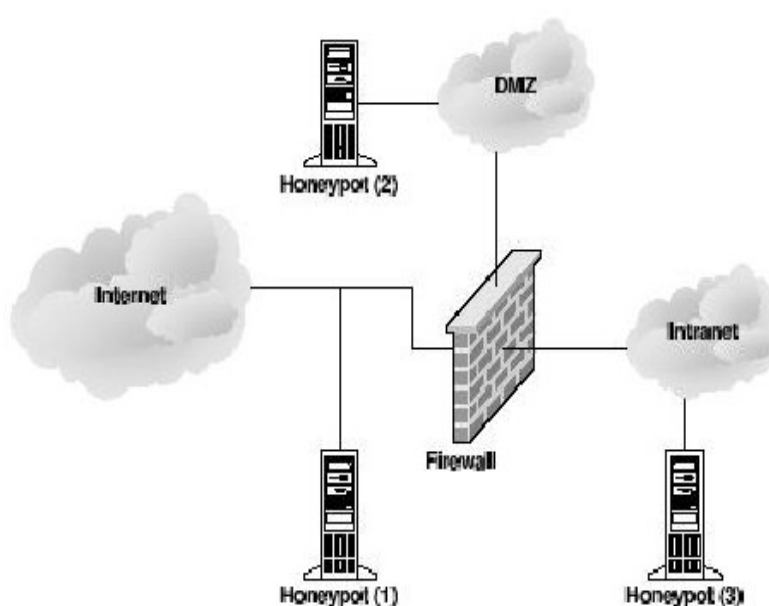
## **4.7 LOCALIZAÇÃO E MODO DE TRABALHO DO HONEYPOT**

Estas ferramentas trabalham como “iscas”, emulando serviços para que o atacante possa pensar que esta em um sistema real. No decorrer da tentativa de um ataque, a ferramenta estabelece uma conexão com o atacante e tenta ganhar tempo utilizando todos os seus recursos, com o objetivo maior de adquirir dados sobre o atacante e também sobre o tipo de ataque. Depois de feita a captura dos dados, eles serão armazenados em um banco de dados para que posteriormente sejam feitas as análises comparativas de novas tentativas de ataque, além de poder ser utilizadas em ataques posteriores.

Geralmente a ferramenta é instalada na intranet ou na internet, de acordo com o objetivo a ser alcançado, podendo ser localizado (BAUMANN; PLATTNER, 2002):

- Em frente ao Firewall, ou seja, fora da rede, não existindo risco para a rede interna e também não gerando logs do Firewall e do IDS. No caso do atacante descobrir a “isca” e dominar o sistema, não será possível controlar o tráfego, portanto não comprometendo o funcionamento de outras redes.
- Atrás do Firewall, a ferramenta é utilizada para tentar descobrir ataques dentro da rede ou detectar configurações vulneráveis do Firewall. Neste caso se a ferramenta for comprometida, o atacante terá acesso a toda rede, não existindo bloqueio do Firewall.
- Quando utilizado na DMZ (Demilitarized Zone), que é a rede adicionada entre uma rede interna e outra externa, a fim de fornecer uma camada adicional de segurança, utilizando o Firewall para fazer o controle de entrada / saída e isolando o Honeypot da rede de produção. Pode ser denominada rede de perímetro.

Na figura 08 abaixo podemos visualizar as três localizações possíveis do Honeypot, tendo assim uma visão melhor dos possíveis riscos.



### **Figura 08 – Localização que o honeypot pode apresentar**

#### **4.8 RISCOS DOS HONEYPOTS**

Apesar de ser uma ferramenta bastante interessante e com um objetivo muito bom, ela pode causar vários incidentes, pois se o atacante descobrir a utilização da ferramenta, ele pode utilizá-la para fazer ataques, podendo se infiltrar e danificar outros sistemas da rede ou até mesmo roubar dados importantes, através da manipulação da ferramenta. Porém, todos esses riscos podem ser diminuídos se for feita uma boa implementação da ferramenta (SPITZNER, 2002).

A eficácia do Honeypot dentro de uma organização depende muito da política de segurança adotada pela empresa, já que elas demonstram a maneira que as empresas trabalham, como os serviços são acessados, a implementação e aplicação de medidas, com o objetivo maior de amenizar os possíveis riscos que podem ocorrer.

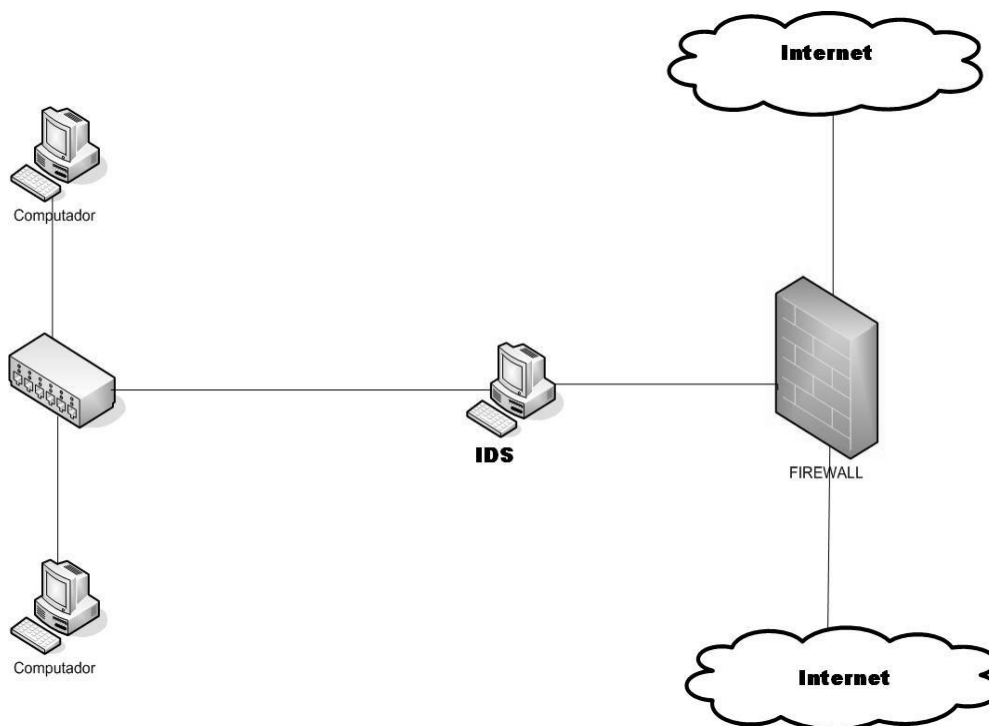
#### **4.9 HONEYPOTS NO BRASIL**

No Brasil existe o projeto HoneynetBR, que foi fundado e é mantido por especialistas do Instituto Nacional de Pesquisas Espaciais (INPE) em associação com o grupo brasileiro de resposta a incidentes de segurança (NBSO) para capturar dados sobre as atividades da comunidade blackhat. O projeto usa Honeypots de alta interação com sistemas reais, mas usando algumas modificações que ajudam a fazer a captura de todos os dados, até mesmo os criptografados. Tempos depois, o projeto recebeu um importante reconhecimento, ele se tornou membro da *Honeynet Research Alliance*, que se trata de um grupo de países de várias

partes do mundo, onde a finalidade é de pesquisar e desenvolver essa tecnologia (PROJETO HONEYPOT...,2000).

## **5 HONEYNETS**

Honeynet é uma ferramenta de pesquisa, cujo o objetivo é projetar uma rede que seja comprometida, ou seja, permanentemente atacada, onde por meio dela serão capturados dados sobre ataques e atacantes para posteriormente ser feita a análise dos mesmos. O comportamento do atacante dentro do sistema é monitorado fazendo à identificação das ferramentas utilizadas, as vulnerabilidades exploradas e o tipo de ataque utilizado. Essa ferramenta de análise é composta de outras ferramentas de segurança, como por exemplo: Firewalls, Honey pots e IDS's, onde juntas formam uma Honeynet, como mostra a figura 03. A Honeynet pode ser utilizada em diversos tipos de sistemas operacionais e arquiteturas variadas, portanto facilitando a observação e análise de atacantes de diversas plataformas.



**Figura 03 – Arquitetura clássica de uma Honeynet**

## **5.1 FUNCIONAMENTO DE UMA HONEYNET**

Uma ferramenta Honeynet, não é considerada um sistema único, ela é sim um conjunto de tecnologias diversas com o objetivo de proteger o ativo da informação.

Uma Honeynet é composta por vários Honeypots ou hosts, sendo que cada um pode possuir diferentes tipos de sistemas operacionais e arquiteturas diversas, logo existira uma grande facilidade em analisar o comportamento de atacantes em diversas plataformas. O servidor central de uma Honeynet será um dos hosts da rede, que possui um serviço chamado syslog, que trabalha como servidor de logs para os outros hosts (SPITZNER, 2002).

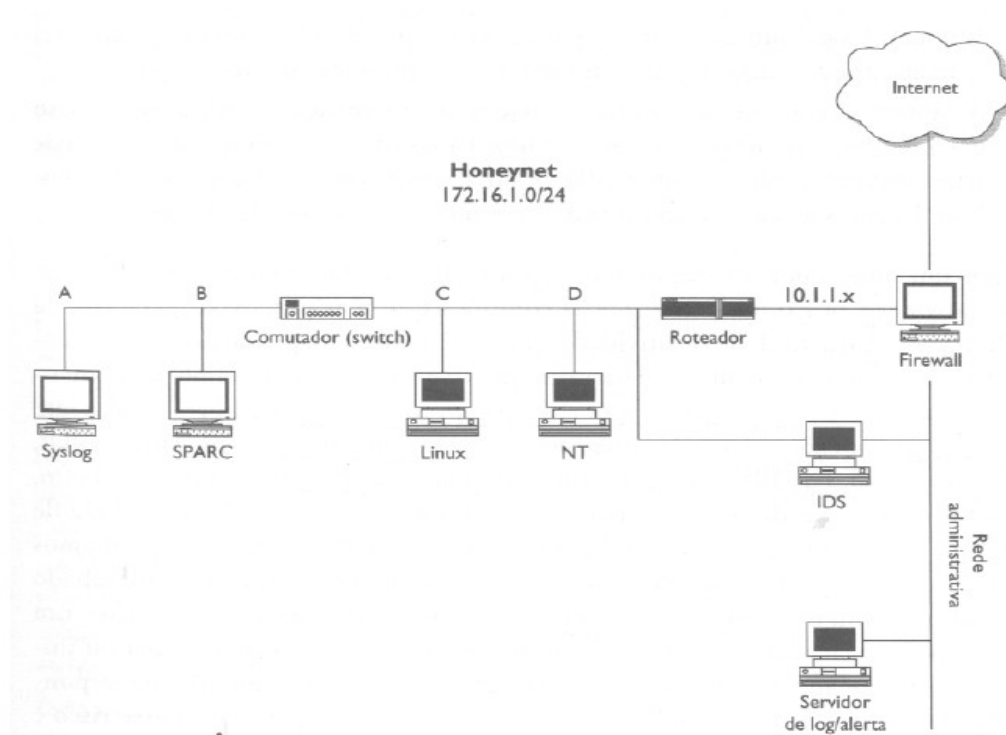
A Honeynet tem como objetivo maior implementar um servidor vulnerável a ataques e tendo como função desviar a atenção dos atacantes dos principais servidores de uma empresa.

Depois do comprometimento de uma Honeynet, o controle, a captura e a análise dos dados tornam-se elementos de suma importância para seu funcionamento.

### 5.1.1 CONTROLE DE DADOS

A Honeynet após ser comprometida, os dados devem ser controlados sem que o atacante perceba, também é necessário garantir que o sistema não seja usado para prover ataques em outros sistemas.

Para fazer o controle de fluxo de entrada e saída de dados, utiliza-se um Firewall em sua arquitetura, tendo como objetivo a passagem de dados da própria Honeynet ou de uma rede externa, como podemos visualizar na figura 04 (SPITZNER, 2002).



**Figura 04 – Honeynet e seu projeto de controle de dados**

Em uma Honeynet o número de conexões feitas a partir de um Honeypot deve ser controlado, logo, quanto maior for o número de conexões maior será o risco. Se houver um número limitado de conexões, facilmente será percebida pelo atacante, assim o Honeypot deixara de ser útil, já no caso de o número de conexões serem ilimitados, sem controle, o sistema poderá ser utilizado para o comprometimento de outros sistemas (SPITZNER, 2002).

### **5.1.2 CAPTURA DE DADOS**

A captura nada mais é que a coleta de dados, ou seja, todas as atividades que são geradas a partir do seu comprometimento. O risco para fazer essa coleta de dados pode ser muito alto, pois podem ocorrer falhas, por isso usa-se um conjunto de vários sistemas para realizar a captura, como logs de Firewall, alertas de IDS, além dos próprios sistemas, para que os registros apresentem um melhor conteúdo de informações (SPITZNER, 2002).

Nessa captura, a forma de armazenamento dos dados deve ser de feita de forma segura, em sistema confiável, e não apenas de forma local no sistema comprometido, já que são informações relevantes para a segurança, pois se o atacante tiver acesso aos registros, o mesmo poderá destruí-los ou fazer um ataque de modificação (SPITZNER, 2002).

### **5.1.3 ANÁLISE DE DADOS**

Através das Honeynets, é realizado o controle e a captura dos dados de um possível atacante, logo essas informações obtidas só serão úteis quando forem transformadas em dados relevantes e com entendimento simples, para serem analisadas posteriormente (SPITZNER, 2002).



Essas informações devem ser capturadas de forma eficaz e inteligente, para que o principal propósito de uma Honeynet seja alcançado, que é a detecção das vulnerabilidades de uma organização, e, além disso, fazer a utilização de ferramentas mais robustas, com a tentativa de combater novos ataques (SPITZNER, 2002).

## 5.2 TIPOS DE HONEYNETS

### 5.2.1 HONEYNET CLÁSSICA

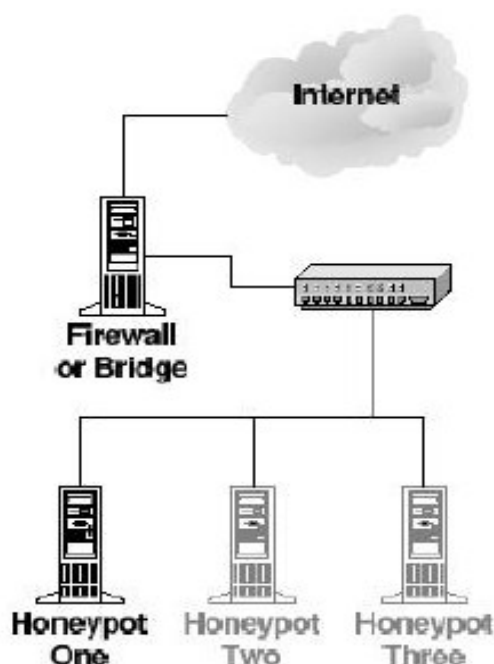
Uma Honeynet clássica ou real é formada por sistemas reais, onde cada um possui suas instalações específicas, neste tipo de Honeynet podem ser utilizados diferentes tipos de sistemas operacionais de forma totalmente descentralizada, ou seja, tornando as máquinas (Honeypots) independentes uma das outras.

Uma Honeynet real é composta dos seguintes dispositivos (HOEPERS;JESSEN;CHAVES, 2005):

- Diversos computadores, um para cada *Honeypot* com um sistema operacional, aplicações e serviços reais instalados;
- Um computador com um *Firewall* instalado, atuando como mecanismo de contenção e de coleta de dados;
- Um computador com um IDS instalado, atuando como mecanismo de geração de alertas e de coleta de dados;
- Um computador atuando como repositório dos dados coletados;
- *hubs/switches* e roteador (se necessário) para fornecer a infra-estrutura de rede da *Honeynet*.

As vantagens desse tipo de Honeynet é que ela utiliza de dispositivos reais e também possui maior segurança pois utiliza Honeypots descentralizados, portanto um dispositivo não está dependendo do outro. Além disso, é mais tolerante a falhas e os atacantes interagem com ambientes reais. Porém, sua desvantagem esta no custo elevado, nas grandes dificuldades de instalação e administração, manutenção mais difícil e trabalhosa, necessidade de mais espaço físico para os equipamentos (SPTIZNER,2002), como mostra a figura 05.

A implantação dessa ferramenta é interessante em organizações que possuem um capital elevado e que tenha como foco principal a segurança de todas suas informações, pois serão necessários treinamentos de funcionários para aprender a lidar com essa tecnologia e além do mais o alto custo dos equipamentos que serão utilizados.



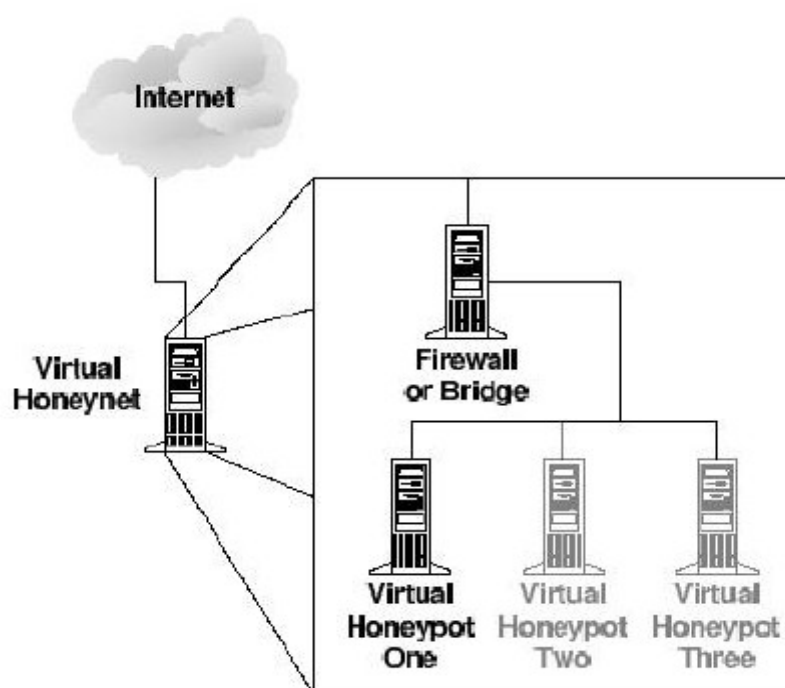
**Figura 05 – Honeynet Clássica**

### **5.2.2 HONEYNET VIRTUAL**

Uma Honeynet virtual é uma máquina onde podem ser executados sistemas operacionais múltiplos ao mesmo tempo, tudo isso devido ao uso de emuladores, criando assim uma rede virtual.

As grandes vantagens de se usar este tipo de Honeynet é devido ao seu custo final ser mais baixo, gerenciamento ser mais simples, manutenção mais simples, facilidade de instalação e administração, necessidade de menor espaço físico para os equipamentos, baixo gasto de energia elétrica, já que se utiliza de um número menor de máquinas. Porém suas desvantagens estão na limitação dos tipos de sistemas operacionais oferecidos pelos softwares virtuais, baixa tolerância à falhas (muitos componentes concentrados em um único ponto), pode haver instabilidade devido ao uso exagerado de memória e o atacante pode obter acesso a outras partes do sistema, pois tudo compartilha os recursos de um mesmo dispositivo, com isso, existe a possibilidade do atacante perceber que está interagindo com um ambiente virtual (SPITZNER, 2002).

A arquitetura de funcionamento desse tipo de Honeynet pode ser mais bem entendida fazendo-se uma análise da figura 06. Este tipo de arquitetura não acarreta muitos custos às empresas interessadas em proteger o ativo de suas informações, apesar de que para isso precise superar alguns problemas como, limitação de sistemas operacionais e sobrecarga de memória.



**Figura 06 – Honeynet Virtual**

### **5.3 A IMPORTÂNCIA DE UMA HONEYNET**

A segurança de informação sempre foi tratada de maneira defensiva, pois sempre era preciso primeiro detectar as falhas e vulnerabilidades para que depois fosse feito algo pra corrigir essas imperfeições.

A ferramenta Honeynet é importante por ser possível detectar e estudar as diversas técnicas utilizadas em um ataque, pois toda e qualquer atividade é considerada suspeita por natureza dentro de uma Honeynet, tal fato é possível devido ao monitoramento intenso na rede, portanto a detecção de riscos é facilitada, pois não ira existir sobrecarga de dados ocasionada pelo tráfego normal de uma rede.

Uma Honeynet é uma ferramenta de análise bastante flexível, que pode ser utilizada tanto como Honeypot de pesquisa como de produção. A grande vantagem da ferramenta é poder funcionar na maioria de todos os sistemas de aplicação e sistemas operacionais.

Geralmente as Honeynets são pouco usadas como Honeypots de produção, pois é muito mais complexo e requer muito mais recursos e tempo para serem implementadas, mas ela é muito mais eficaz quando é utilizada como Honeypot de pesquisa, assim podendo atuar em diversas áreas.

Como Honeypot de pesquisa, as Honeynets fazem a captura dos ataques já conhecidos e dos desconhecidos, utilizando-se da base de dados das assinaturas existentes para analisar os dados, podendo assim descobrir novas vulnerabilidades e com isso evitando ataques futuros (FONSECA; PITANGA, 2001).

A ferramenta faz a captura de informações sobre os ataques baseando-se nas atitudes dos atacantes, mostrando passo a passo a atitude deles dentro do sistema e o motivo pelo qual levou ao ataque, apenas usando ferramentas para sondar e explorar a vulnerabilidade dos sistemas.

Ferramentas desse tipo, não são produtos que se instalam muito menos um instrumento que se pode baixar em sua rede para auxiliar na segurança, mas sim uma arquitetura que tem como função auxiliar no controle da rede, na qual pode ser implantado qualquer sistema ou aplicação, portanto sua arquitetura é bastante complexa, e possui três elementos críticos: controle de dados (a circunstância de risco), captura de dados (registro de atividades do atacante) e a coleta de dados (centralizar a captura e agrupar todas as informações coletadas dentro da Honeynet), (SPTIZNER, 2002).

Essas ferramentas permitiram o registro de domínio do nome da mesma, atraindo assim o cliente que se deseja pesquisar. Existe um alto nível de risco ao usar a ferramenta, pois os atacantes não se preocupam com os limites e fazem o que bem entendem no sistema,

já que utilizam as Honeynets para compilar códigos maliciosos, lançar ataques ou distribuir ferramentas. Uma possibilidade para banir ataques é o controle de dados fora dos Honeypots, outra vulnerabilidade da ferramenta é a variedade de tecnologias envolvidas e o fato de serem utilizadas para a captura e controle de atividades esperadas (SPTIZNER,2002).

## **6 ESTUDO DE CASO: O HONEYPERL**

Honeyperl é um Honeypot relativamente novo e ainda em fase de estudo, pois novas versões do software ainda estão em desenvolvimento pelo projeto Honeypot-BR, com o objetivo de corrigir problemas e adicionar funcionalidades aos que já foram lançados e testados (PROJETO HONEYPOT...,2000).


O Honeyperl é um Honeypot de media interação que simula diversos serviços como: http, smtp, ftp e outros, através de um conjunto de servidores falsos (fake servers) vulneráveis que tem com maior objetivo atrair hackers, enganando-os e pegando todas as atividades não “permitidas”, ou seja, tentativas de intrusão ou outra coisa em seus fakes servers

(PEREIRA,2006). A ferramenta é ideal para sistema Linux em geral (PROJETO HONEYPOT...,2000). Na figura 9 podemos observar melhor a tela inicial da ferramenta, mostrando a inicialização das fakes serves.

Além de ser uma ferramenta muito fácil de instalar e configurar (o arquivo de configuração esta em português do Brasil e muito bem comentado) é uma ferramenta muito boa para administradores que querem descobrir algum invasor na rede (PEREIRA,2006).

Ele funciona da seguinte forma, por exemplo, iremos supor que um invasor tente logar pelo ftp, o ataque irá pensar que ele está logando e não está sendo monitorado, mas com o Honeyperl você verá tudo o que o atacante fizer em seu suposto server ftp (fake server) e irá gerar em registro de log toda informação sobre o atacante dentro do suposto server, como o IP e com a data e hora em que o ataque foi feito (PEREIRA,2006).

O Honeyperl é uma ferramenta Open Source, ou seja, ela aceita novos parâmetros em sua configuração original. A ferramenta também traz em sua configuração uma opção de ser utilizada junto com um firewall do tipo filtro de pacote, com as opções para a utilização o Ipchains ou o Iptables.

The image shows a terminal window titled 'Sessão' with a menu bar containing 'Sessão', 'Editar', 'Ver', 'Favoritos', 'Configurações', and 'Ajuda'. The terminal content displays the Honeyperl version 0.0.7.1, credits to Antonio Marcelo, Daniel B. Cid, Adriano Carvalho, Humberto Sartini, and Fábio Henrique, and the project website. It then shows the server starting, the program running as PID 3121, the user as root, the firewall being disabled, and the execution as root. Finally, it lists the starting of various fake services: fakepop3 (port 110), fakeecho (port 7), fakesmtp (port 25), fakeftp (port 21), fakehttpd (port 80), fakepit (port 20001), and fakesquid (port 3128). A large, stylized graphic of a globe is visible in the background of the terminal window.

```
Sessão  Editar  Ver  Favoritos  Configurações  Ajuda
Shell
#####
#               Honeyperl versao 0.0.7.1               #
#               Por Antonio Marcelo, Daniel B. Cid      #
#               Adriano Carvalho, Humberto Sartini & Fábio Henrique #
#               Projeto Honeypot-BR http://www.honeypot.com.br #
#####

Servidor.....[servidor]
Iniciando o programa no PID.....[3121]
Usuario: root
Firewall Desativado !
Rodando como root !!!

Iniciando o fakepop3 - porta: 110
Iniciando o fakeecho - porta: 7
Iniciando o fakesmtp - porta: 25
Iniciando o fakeftp - porta: 21
Iniciando o fakehttpd - porta: 80
Iniciando o fakepit - porta: 20001
Iniciando o fakesquid - porta: 3128
█
```

**Figura 09 – Tela inicial do Honeyperl**

## 6.1 CONFIGURAÇÃO DO HONEYPERL

Segundo (MARCELO,2005):

O Honeyperl é um software de fácil instalação e configuração desenvolvido em Perl. Pode ser obtido no site do projeto HoneypotBR e instalado de maneira bem rápida. Trata-se de um Honeypot de média interatividade que simula os seguintes serviços: Squid, Apache, servidores de e-mail (Sendmail, Postfix, Qmail e MExchange), FTP, Echo e POP3.



Ele ainda é capaz de capturar assinaturas de vírus num pequeno Tarpit (poço de piche) implementado. A única dependência é que o Perl (versão 5.6.0 ou superior) esteja instalado no sistema. Para configurar o Honeyperl basta executar a seguinte sequência de comandos:

- Descompacte o pacote com o comando ***tar -xzf honeyperl.0.0.7.1.tar.gz***
- Execute o comando ***perl verify.pl*** para verificar se todos os módulos Perl necessários para a execução do programa. Caso seja necessário, essa rotina acessa automaticamente o repositório CPAN e executa o download e a instalação dos módulos Perl faltantes.

O processo de configuração deve ser feito de forma bastante cuidadosa. Inicialmente devem ser escolhidos os serviços que serão executados no Honeypot. O Honeypot fornece dois tipos de arquivos de configuração:

***honeyperl.conf***: Arquivo principal de configuração do programa. Fica localizado no diretório ***conf*** juntamente com o dos outros módulos. Dentro dele há algumas variáveis importantes que devem ser configuradas. São as seguintes:

```
#####
# ##### #
# # Secao1 # #
# ##### #
# #
# # Dominio que será utilizado pelas fakes #
# dominio=honeybot.com.br #
#####
```

Essa variável permite a definição de um nome de domínio. Coloque o da sua instituição/empresa. Esse parâmetro faz com que, na hora em que o intruso fizer o levantamento de informações sobre o servidor, receba como resposta o nome definido.

Exemplo: ***servidor.honeybot.com.br***

```
#####
# #email utilizado nos fakes #
# email=admin@honeybot.com.br #
#####
```

Um endereço de e-mail de sua escolha para aparecer como contato nas respostas falsas que os serviços irão emitir. Recomendamos um endereço falso ou uma conta especialmente selecionada para isso.

```
#####
# Usuário utilizado                                     #
# usuario=root                                         #
#####
```

Usuário para execução do Honeyperl. Deixe-o com o root.

```
#####
# #Deseja ver as mensagens no terminal?                #
# opcoes: (sim/yes) / (nao/no)                        #
# terminal=sim                                         #
#####
```

Mostra as mensagens referentes a ataques em tempo real no terminal, permitindo assim seu acompanhamento.

```
#####
# Deseja ativar firewall                               #
# opções: (sim/yes) / (nao/no)                        #
# firewall=não                                         #
#####
```

Essa opção tem que ser manipulada com muito cuidado. Estando ativa, o Honeyperl irá gerar um arquivo de regras de firewall baseado no iptables, no subdiretório *firewall* e atualizara a tabela do iptables. Alguns usuários tiveram problemas com bloqueio de endereços.

```
#####
# Sistemas de firewall disponíveis.                   #
# Pode-se ter linux22, linu24 ou openbsd              #
# openbsd : trabalha com PF                           #
#linux24 : IPTables                                   #
#linux22 : ipchains                                   #
# so=linux24                                           #
#####
```

Com essa opção, podemos definir se o firewall vai ser feito usando o *PF*, do OpenBSD, o *ipchains* ou o *iptables*.

A segunda seção do arquivo de configuração é que determina os serviços falsos (os fakes) a serem executados. Dependendo da simulação desejada podemos ativar ou não determinado fake.

```
#####
# ##### #
# # Secao2 # #
# ##### #
# # #
# # #
# Fakes a serem iniciados #
# fakesquid:squid:conf/fakesquid.conf:3128:Squid Emul #
# fakesmtp:smtp:conf/fakesmtp.conf:25:Smtp Emul #
# fakehttpd:httpd:conf/httpd.conf:80:Httpd Emul #
# fakepop3:pop3:conf/pop3.conf:110:Pop3 Emul #
# fakeecho:echo::7:Echo emul #
# fakeftp:ftp:conf/fakeftp.conf:21:Ftp Emul #
# fakepit:pit::20001:Pit Emul #
#####
```

A estrutura da definição de um fake é a seguinte:

```
#####
# nomedofake:modulo(service):arquivo de config:porta TCP:comentario #
#####
```

Colocar uma cerquilha(#) na frente de qualquer um dos parâmetros impede a execução daquele serviço. Recomendamos ao usuário não modificar esse parâmetros de inicialização. A configuração dos fakes pode ser feita modificando seus artigos de configuração correspondentes, encontrados no subdiretório *conf*. São eles:

- ***fakesquid.conf***: Arquivo de configuração do *fakesquid*, emulador do proxy Squid. O parâmetro de inicialização é *\$bugsquid=" Squid/2.4 Stable3"*; , que indica o *banner* da versão que deve aparecer nas respostas do *fakesquid*.
- ***fakesmtp.conf***: Arquivo de configuração do *fakesmtp*, emulador de servidores de correio. Possui os seguintes parâmetros: *\$servemul="sendmail"*; indica qual

servidor de correio será emulado. As opções validas são: *exchange*, *sendmail*, *qmail* e *postfix*. Já *\$logdir="logs/smtp"*; indica diretório de *log* (registro) do servidor de SMTP onde serão armazenados os arquivos de log e das mensagens enviadas com o endereço IP da máquina do agressor.

- ***Httpd.conf***: Arquivo de configuração do *fakehttpd*, emulador do Apache. Possui o parâmetro *\$httpd="Apache/1.3.27"*; que indica a versão do Apache que deverá aparecer no banner nas respostas do fakehttp.
- ***Pop3.conf***: Arquivo de configuração do *fakepop3*, emulador de servidores POP3. Possui os seguintes parâmetros: *serveremul="qpopper"*; que indica qual servidor POP3 será emulado. As opções válidas são *teapop*, *qpopper* e *pop3*. Já *\$logdir="logs/pop3.log"*; indica o diretório em que ficarão os arquivos de log do serviço.
- ***Fakeftp.conf***: Arquivo de configuração do *fakeftp*, o emulador do servidor FTP *wuftp*. Os parâmetros de configuração são os seguintes: *\$programaftp="wuftp"*; indica o servidor FTP a ser emulado. Já *\$conteudoftp="total 0\x0d\x0a"*; indica para o fake qual conteúdo(chamado *honeytokens*) será exibido para o agressor.
- O fake *echo* não possui arquivo de configuração. O *fakepit* é utilizado para captura de assinaturas de worms, identificados por sua porta de entrada.

## 6.2 CONFIGURANDO O IPTABLES

Para finalizar o projeto, criaremos um pequeno script de iptables para nosso honeypot (o script esta no anexo).

Esse script irá liberar o acesso às portas dos serviços executados pelo honeypot.

Salve o arquivo com o nome de ***rc.firewall*** no diretório ***/etc/rc.d/*** e em seguida digite o comando ***chmod 755rc.firewall*** para que possamos executá-lo a partir do script de inicialização ***rc.local*** (script em anexo).

### 6.3 EXECUTANDO O HONEYPOT

Para executar o Honeyperl, simplesmente execute o comando ***perl honeypperl.pl***. Uma mensagem surge na tela informando que o programa está em execução. Para maior comodidade, rode o programa em segundo plano (*background*) com o comando ***perlhoneyperl.pl&***.

É possível adicionar parâmetros como ***-h*** (ajuda), ***-v*** (imprime no terminal avisos sobre os ataques mesmo que o arquivo de configuração determine o contrário) e ***-l arquivo.log*** (especifica um arquivo de log e sobrescreve o parâmetro equivalente no arquivo ***honeypperl.conf***).

A partir desse instante o Honeypot escutará qualquer ataque que seja feito às portas ativas dos fakes em execução. Os logs ficarão armazenados no sub-diretório ***logs***, em subdiretórios identificados com o nome do serviço (***echo, ftp, httpd, squid***, etc). A estrutura do nome dos arquivos é:

```
#####
# logs/httpd/12-20—2004(18:15:23).log                                     #
#####
```

Ou seja, ***logs/serviço/mês-dia-ano(hora).log***. Dentro do arquivo temos informações como às mostradas a seguir, que ilustram um ataque ao fakehttpd:

```
#####
# Mon Dec 20 18:21:39 2004 fakehttpd log – Conexão from 10.0.0.1:37378      #
# get http 1.1 : Ataque WEB ! Tentativa de execucao de comando             #
#####
```

## 6.4 ORIENTAÇÃO FINAL

Vamos deixar nosso sistema “pronto” para execução do honeypot logo após a inicialização. Edite o arquivo **/etc/rc.d/rc.local** para que se pareça com o mostrado na listagem 4. Agora, reinicie seu computador. Após o login, digite o comando: **perl /diretoriodohoneyperl/honeyperl.pl** e nosso honeypot estará pronto para o trabalho.

## 6.5 CONSIDERAÇÕES FINAIS

Honeypots devem ser implementados de maneira cuidadosa, e é bastante interessante que seja feito um estudo de completo de viabilidade antes da instalação. Um Honeypot pode se tornar uma fonte enorme de informações, mas se mal-utilizado será uma brecha de segurança em qualquer sistema (MARCELO, 2005).

## 7. CONSIDERAÇÕES FINAIS

Neste trabalho foi abordada a importância do uso de ferramentas de segurança dentro de uma rede. As ferramentas abordadas foram: Firewalls, IDS, Honeynets e Honeypots.

Mostrando suas desvantagens e vantagens, e a melhor maneira de serem utilizadas a partir do resultado que se queira alcançar. Além disso, mostramos a importância de utilizarmos essas ferramentas em conjunto, com o objetivo de dar ainda mais segurança para a rede.

A utilização de Honeypots na proteção e na pesquisa de ataques, ainda é pouco difundida no meio da segurança, pois o Honeypots diferente das outras ferramentas, tem a idéia de interagir com o atacante (sem o atacante ter noção disso) para descobrir um pouco mais da ação dos atacantes dentro de uma rede, por exemplo, as vulnerabilidades exploradas, ferramentas utilizadas e IP do atacante.

Como dito anteriormente, a utilização da ferramenta junto com outras tecnologias de segurança como Firewalls e IDS's, não garantem segurança total da rede, porém aumenta e muito o grau de proteção, quando essas ferramentas são bem gerenciadas para trabalharem em conjunto.

Com essas informações, conclui-se que uma ferramenta por si só não é capaz de ter uma boa eficácia na proteção de uma rede, pois cada uma delas tem o seu ponto fraco e para sanar essa deficiência é melhor utilizar essas ferramentas de forma agregada para que cada uma com sua função possam conseguir o grande objetivo, que é a proteção de informações importantes.

## **7.1 TRABALHOS FUTUROS**

Outros trabalhos relacionados à segurança podem ser desenvolvidos a partir do exposto. Como a criação de uma rede totalmente projetada para ser comprometida onde

possam ser feita a implementação de Firewalls, IDs e Honeypots, com o objetivo de prover a segurança dentro de uma grande rede, a partir do resultado desse experiência.

## REFERÊNCIAS BIBLIOGRÁFICAS

ALECRIM, Emerson. **Firewall: conceitos e tipos.** Disponível em <<http://www.infowester.com/firewall.php>>. Acesso em: 01.abr.2006.



BACE, Rebeca. MELL, Peter. **NIST Special Publication on Intrusion Detection Sustems.**

Disponível em <<http://csrc.nist.gov/publications/nistpubs/800-31/sp800-31.pdf>>. Acesso em: 07.maio.2006.

BATTISTI, Júlio Cesar Fabris. **Tutorial de TCP/IP – Parte 17 – IFC – Internet Firewall Conection (Windows XP).** Disponível em <[http://www.linhadecodigo.com.br/artigos.asp?id\\_ac=651&pag=1](http://www.linhadecodigo.com.br/artigos.asp?id_ac=651&pag=1)>. Acesso em: 03.abr.2006.

BAUMANN, Reto. PLATTNER, Christian. **Honeypots.** Disponível em <<http://security.rbaumann.net/download/diplomathesis.pdf>>. Acesso em: 25.jun.2006.

BEAL, V. **The Difference Between Adware & Spyware.** 2004. Disponível em: <<http://www.webopedia.com/DidYouKnow/Internet/2004/spyware.asp>>. Acesso em: 22 mar. 2006.

CERT.BR. **Cartilha de Segurança para internet – Parte I: Conceitos de Segurança.** Disponível em: <<http://cartilha.cert.br/>>. Acesso em: 24.out.2006

CERT.BR. **Cartilha de Segurança para internet – Parte II: Riscos Envolvidos no uso da internet e métodos de prevenção.** Disponível em: <<http://cartilha.cert.br/>>. Acesso em: 24.out.2006

CHESWICK, Bill. **An Evening with Berfered in wich a Cracker is Lured, Endured, and, Studied,** 1991.

COHEN, Fred. **The Deception Toolkit Home Page and Mailing List**. Disponível em: <<http://www.all.net.dtk/dtk.html>>. Acesso em: 25.ago.2006.

FONSECA, Antonio Marcelo Ferreira da. PITANGA, Marcos. **A Arte de Iludir Hackers**. 1.ed. São Paulo: Brasport, 2001.

GRIFFIN, B. **An Introduction to Viruses and Malicious Code, Part One: Overview**. 2000. Disponível em: <<http://www.securityfocus.com/infocus/1188>>. Acesso em: 08 mar. 2006.

HOEPERS, Cristine. JESSEN, Klaus Steding. CHAVES, Marcelo H. P. C. **Honeypots e Honeynets: Definições e Aplicações**. Disponível em: <<http://www.cert.br/docs/whitepapers/honeypots-honeynets>>. Acesso em: 03.maio.2006.

MARCELO, Antônio. **Honeypots no Linux**. Disponível em: <<http://www.linuxmagazine.com.br>>. Acesso em: 20.jul.2006

MICROSOFT. **What you can do about spyware and other unwanted software**. 2005. Disponível em: <<http://www.microsoft.com/athome/security/spyware/spywarewhat.mspx>>. Acesso em: 22 mar. 2005.

NED, Frank. **Introdução a IDS**. Disponível em <<http://www.rnp.br/newsgen/9909/ids.html>>. Acesso em: 05.abr.2006.

PAXSON, V.; ZHANG, Y. **Detecting Backdoors**. 2000. Disponível em:

<<http://www.icir.org/vern/papers/backdoor/>>. Acesso em: 13 mar. 2006.

PEREIRA, Leandro Tofino. **Enganando invasores com Honeyperl**. Disponível em: <[http://](http://www.vivalinux.com.br/artigos/verartigo.php?codigo=53258pagina=2)

[www.vivalinux.com.br/artigos/verartigo.php?codigo=53258pagina=2](http://www.vivalinux.com.br/artigos/verartigo.php?codigo=53258pagina=2)> Acesso em:

10.nov.2006

PROJETO Honeypot Brasil. Disponível em: <<http://www.honeypot.com.br>>, Acesso em:

20.mar.2006.

ROJAS, Gislaine Aparecida. **Análise de Instruções através de Honeypots e Honeynets**,

2003.

ROHR, Altieres. **Rootkit**. Disponível em:

<<http://linhadireta.uol.com.br/informativos/definicoes/rootkit/>> Acesso em: 21.out.2006

SANTANNA, João. **Notas de aula – Unidade 1.2 – Ataques de Segurança – Modelo de Segurança em redes**. IESAM, 2006.

SPITZNER, Lance. **Traching Hackers**. 1.ed New York: Addison Weslly, 2004.

STOLL, Cliff. **The Cuckoo's Egg**. New York: Pockte Books Nenfiction, 1990.

WEB, Reseller. **Armadilha para os Hackers.** Disponível em:  
<[http://www.resellrweb.com.br/shared/print\\_story.asp?id=42776](http://www.resellrweb.com.br/shared/print_story.asp?id=42776)>. Acesso em: 15.ago.2006.

## [ANEXO]

Este anexo, contém o arquivo de configuração do Honeyperl, do iptables e do arquivo rc.local em sua forma padrão, podendo ser modificado de acordo com a necessidade de implementação.

### Arquivo de Configuração do Honeyperl

```
#####
#### Secao1 ####
#Dominio que sera utilizado pelas fakes
#dominio=teste.com.br
#Email utilizado nos fakes
#email=admin@honeypot.com.br
#Usuario utilizado
#usuario=root
#Deseja ativar firewall
#firewall=nao
#Os sistemas disponiveis para utilizacao de firewall:
#pode-se ter linux22, linux24 ou openbsd
#openbsd : trabalha com PF
#linux24 : IPTables Kernel 2.4 e 2.6
#linux22 : ipchains Kernel 2.2
#so=linux24
#### Secao 2 ####
#Fakes a serem iniciados
#fakesquid:squid:conf/fakesquid.conf:3128:Squid Emul
#fakesmtp:smtp:conf/fakesmtp.conf:25:Smtp Emul
#fakehttpd:httpd:conf/httpd.conf:80:Httpd Emul
#fakepop3:pop3:conf/pop3.conf:110:Pop3 Emul
#fakeecho:echo::7:Echo emul
#fakeftp:ftp:conf/fakeftp.conf:21:Ftp Emul
#fakepit:pit::20001:Pit Emul
#####
```

## Arquivo de configuração do iptables

```
#####
01 #! /bin/sh
02 #
03 # rc.firewall
04 #
05 #Por Antonio Marcelo – amarcelo@plebe.com.br
06 #Script padrão para firewalls baseados em iptables
07 #
08 #
09
10 if [ "$1 = "flush" ] ; then
11     echo "Flushing"
12     iptables -F INPUT ACCEPT
13     iptables -F FORWARD ACCEPT
14     iptables -F OUTPUT ACCEPT
15     iptables -F
16     iptables -t nat -F      # Flush no NAT
17     iptables -X            # Flush nas CHAINS PERSONALIZADAS
18     iptables -Z            # Zera regras específicas. Quando não houver
19                             # argumentos, zera todas as regras. Idem ao -f.
20     Echo "done"
21 else
22     echo "Iniciando firewall"
23
24     iptables -F INPUT DROP
25     iptables -F FORWARD DROP
26     iptables -F OUTPUT ACCEPT
27
28     #Liberando as portas de nosso honeypot
29     iptables -A INPUT -p tcp -m multiport --destination-port 20,21,25,80,110,3128
30     iptables -A INPUT -p TCP -m state --state ESTABLISHED, RELATED -j ACCEPT
31
32 FI
#####
```

## Arquivo de configuração do rc.local

```
#####  
01 # ! /bin/sh  
02 #  
03 # /etc/rc.d/rc.local:  Local system initialization script.  
04 #  
05 #Put any local setup commands in here:  
06 #  
07 #  
08 echo "Iniciando o honeypot"  
09 #  
10 #Iniciando firewall  
11 /etc/rc.d/.rc.firewall  
12 #  
13 #Mostrando regras de firewall  
14 iptables -nL  
15 #  
16 echo "ok"  
#####
```