FACULDADES INTEGRADAS DE CARATINGA

FACULDADE DE CIÊNCIA DA COMPUTAÇÃO

ANÁLISE DE INVASÕES SOBRE HONEYPOTS DE BAIXA INTERATIVIDADE EM AMBIENTES LINUX

GLÁUCIO DOUGLAS MOREIRA

CARATINGA

2010

Gláucio Douglas Moreira

ANÁLISE DE INVASÕES SOBRE HONEYPOTS DE BAIXA INTERATIVIDADE EM AMBIENTES LINUX

Monografia apresentada à Faculdade de Ciência da Computação das Faculdades Integradas de Caratinga como exigência parcial da disciplina de Trabalho de Conclusão de Curso sob orientação do Professor Jacson Rodrigues Correia da Silva.

Caratinga
2010

Gláucio Douglas Moreira

ANÁLISE DE INVASÕES SOBRE HONEYPOTS DE BAIXA INTERATIVIDADE EM AMBIENTES LINUX

Monografia submetida à Comissão examinadora designada pelo Curso de Graduação em Ciência da Computação das Faculdades Integradas de Caratinga como requisito para obtenção do grau de Bacharel.

_____

Prof. Jacson Rodrigues Correia da Silva
Faculdades Integradas de Caratinga

_____

Prof. Msc Fabrícia Pires Souza Tiola
Faculdades Integradas de Caratinga

_____

Prof. Felipe Costa Fernandes
Faculdades Integradas de Caratinga

Caratinga, 01/12/ 2010

DEDICATÓRIA

Dedico essa vitória de uma maneira muito carinhosa e particular a meus pais, Manoel Moreira Filho e Maria das Graças Barros Moreira, os quais um dia foram fundadores da minha existência e hoje nada mais justo de minha parte oferecer esse presente a vocês. Amo muito vocês!

AGRADECIMENTOS

A Deus, pela oportunidade a mim confiada, por todos momentos de superação que o Senhor estava pronto a me acolher e fortalecer.

Agradeço meus pais, minha namorada, e todos familiares de uma forma muito carinhosa e gratificante, pois acima de tudo essa vitória também é de vocês.

Aos professores que compartilharam seus conhecimentos no decorrer desses anos, especialmente ao Jacson, pela orientação no trabalho de conclusão e a Fabrícia por todo incentivo nos últimos dois semestres.

Um agradecimento com sabor de saudades a todos amigos de classe que compartilharam comigo um pouco de suas vidas ao longo desses 4 anos, que com certeza irá ficar marcado na memória aonde quer que estejamos daqui pra frente. E hoje posso dizer com gratidão, que toda essa correria, todo esse sufoco, noites mal dormidas estudando, me traz um sentimento de missão cumprida, e hoje posso bater no peito e dizer que tudo isso valeu.

"…'Melhor que sonhar com o futuro é construí-lo!' Tento lembrar disso todos os dias ... quando leio me incomoda e acelero o 'passo' ... somos muito omissos ... sabemos o que tem que ser feito ... e atrasamos tudo pensando no fato consumado dos resultados como se eles não fossem a consequência natural dos passos certos ..."

Autora: Mariana Carneiro

RESUMO

Redes de computadores ocupam um campo amplo e em grande desenvolvimento na computação. Na área de estudos em redes de computadores tem-se outros desmembramentos, nos quais a segurança da informação se enriqueceu muito com avanço da Internet. Em consequência dessa rápida expansão, o estudo sobre *Honeypots* vem se destacando cada vez mais como aliado dos Administradores de redes e sistemas sobre ataques e descoberta de falhas em diversos softwares.

Honeypots são artifícios de segurança usados para serem invadidos, atacados e totalmente comprometidos de forma a ser levantadas diversas informações a partir dos registros, como por exemplo, o comportamento dos atacantes envolvidos, e os níveis de comprometimento com o sistema, destacando os serviços mais desejados pelo universo Hacker. A partir de novos métodos de ataque são estudadas formas de melhorias nos diversos padrões em computação no intuito de assegurar as informações pertencentes a determinada organização que é o foco principal dos ataques.

Para que esse estudo fosse possível, utilizou-se um ambiente *Honeypot* de baixa interatividade, onde se trata de um ambiente isolado e com serviços totalmente virtualizados conectado a internet e totalmente sem segurança, de forma a facilitar a entrada e registro de ações maliciosas.

O trabalho foi realizado sobre a coleta de informações de caráter genérico, ou seja, não foi descrito informações como, o que ocorreu no sistema a cada hora de cada dia durante 44 dias, e sim informações em comum: origem do IP envolvido no ataque, usuários e senhas mais utilizados no acesso aos serviços, horário onde ocorre maior incidência de ataques, comandos mais executados, serviços mais temidos, número de tentativas e serviços explorados oriundos de um mesmo IP em relação a data, dentre outras. Conseguiu-se levantar algumas informações principais a partir desse trabalho como:

- O chile é o país que mais se originou ataque
- O Brasil ocupa o segundo lugar em índice de reincidência de ataques
- O servidor HTTPD é o mais procurado por Hackers
- O horário de maior incidência é pela madrugada

De posse a essas informações várias conclusões foram levantadas em cima de resultados, comprovando o valor que o *Honeypot* agrega a uma organização que ainda

não utiliza desse recurso para melhorias em segurança.

Palavras chave: *Honeypots*, *Hackers*, Segurança em redes, *Honeyperl*, *Fakeserver*.

ABSTRACT

Computer networks occupy a vast area and great development in computing. In the area of study in computer networks has other spin-offs, in which information security is much enriched with advancement of the Internet. In consequence of this rapid expansion, the study on *Honeypots* has been increasing more and more as an ally of the network and system administrators about attacks and the discovery of flaws in various software.

Honeypots are security devices used to be invaded, attacked and totally committed to being raised in a variety of information from the record as the behavior of attackers involved and levels of commitment to the system, highlighting the services most desired by the universe Hacker. From new methods of attack are studied in various ways of improving standards in computing in order to ensure the information belonging to an organization that is the main focus of the attacks.

For this study possible, will use a low-interactivity *Honeypot* environment where it is an isolated environment with services and fully-virtualized and fully connected to the Internet without security to facilitate the entry and registration of malicious actions.

The study was conducted to collect information on generic, ie not described in the statistical data which occurred in the system every hour of every day in those 44 days, but common information such as source IP involved in the attack , more users and passwords used in accessing services, the time where the increased incidence of attacks, run commands, services, most feared, number of attempts and services operated from the same IP over time, among others.

Keywords: *Honeypots, Hackers, Netwok Security, Honeyperl, Fakeserver*.

ÍNDICE DE ILUSTRAÇÕES

ÍNDICE DE TABELAS

LISTA DE SIGLAS

CERT.br	Centro de estudos, respostas e tratamento de incidentes
CSIRT	Grupo de Segurança e repostas a incidentes brasileiros
DOS	Denial Of Service
FTP	File Transfer Protocol
HOST	Computador conectado a rede
IDS	Intrusion Detection System
IP	Internet Protocol
LAN	Local Area Network
NIC.br	Comitê Gestor de Internet no Brasil
NMAP	Software livre que realiza Port Scan
POP3	Post Office Protocol
SMTP	Simple Mail Transfer Protocol
SYN-ACK	Acknowledge
SYN	Syncronize
TELNET	Protocolo cliente servidor que permite comunicação entre computadores
TCP	Transmission Control Protocol
TI	Tecnologia da Informação
UDP	User Datagrama Protocol
WAN	Wide Area Network

Sumário

1. INTRODUÇÃO

Os *Honeypots*, segundo ASSUNÇÃO(2009), também conhecidos como “potes de mel”, exercem função parecida com o mel produzido pelas abelhas, nos quais o mel propriamente dito atrai abelhas até mesmo de outros ambientes. Se tratando de redes de computadores e segurança o termo *Honeypots* tem o objetivo de atrair *Hackers*, de forma a analisar, estudar e projetar ambientes cada dia mais atrativos, para que sejam invadidos e comprometidos por invasores de todo mundo. Como resultado obter seus rastros através do ambiente *Honeypot* de forma a gerar um histórico dos ataques e a qual falha se refere no sistema, podendo assim ser tratada.

ASSUNÇÃO(2009) ainda ressalta que quando vários *Honeypots* são agrupados para funcionarem juntos em uma rede, temos uma evolução do termo, chamado de *Honeynet*s. Neste caso, geralmente tudo é realizado em um ambiente de produção verdadeiro, nada é emulado.

O ganho com os estudos sobre *Honeypots* e *Honeynet*s a uma organização é incalculável, pois muitas falhas que poderiam levar a enormes prejuízos acabam sendo amenizadas onde facilmente consegue-se a reparação.

Os motivos que levam os Hackers a efetuar os ataques são diversos, pois variam desde a curiosidade em aprender, no intuito de testar sua capacidade, ou até o extremo, relativo a ganhos financeiros, extorsão, algum tipo de chantagem, espionagem industrial, venda de informações confidenciais e o que está muito comum na atualidade, ferir a imagem de uma determinada empresa ou pessoa, onde geralmente a notícia de que uma empresa foi invadida é proporcional a sua fama, na maioria dos casos deixando transparecer a ideia que seja uma organização não segura.

O aspecto jurídico também é sempre levantado quando se trata de utilização de *Honeypots*, pois pessoas alegam que um *Honeypot* induz alguém a fazer algo errado. Isso de forma alguma é verdade, ASSUNÇÃO(2009). Um pote de mel não está induzindo ninguém a realizar nada de errado, até porque muitas vezes ele é um computador como outro qualquer, apenas a finalidade de colocá-lo é diferente, dessa forma o *Honeypot* não está sendo exibido para ninguém, o invasor realizou o

acesso por livre vontade.

As empresas hoje em dia investem quantias fantásticas em segurança, infelizmente no Brasil o investimento em segurança ainda é pequeno se comparado a outros países. O retrato de tal descaso à segurança da informação no Brasil é claramente traduzido na falta de leis neste sentido. Além disso, existe um fator agravante que quando existe o interesse em elaborar tais leis, serão por indivíduos que não tem por objetivo principal a segurança em si próprio. O resultado serão leis absurdas, que irão agravar ainda mais o quadro de segurança em redes de computadores no país, ao invés de realmente ajudar.

Ainda se tratando do ambiente corporativo, outro fator agravante que leva a ataques e roubos, não se trata diretamente de falhas em software, mas sim da falta de ética profissional de quem trabalha na organização, que muitas vezes repassa informações sigilosas da empresa para indivíduos externos. O motivo talvez seja vínculo de amizade, ou desentendimento com algum diretor, coisas desse tipo. Esse tipo de atuação é conhecido como engenharia social, que é o lado onde o *Hacker* induz o profissional de uma organização a repassar informações que podem consequentemente gerar um ataque, trazendo grandes problemas para a organização por intermédio de um funcionário que em algumas vezes repassa informação sem saber que ela seja o suficiente para causar um dano, através de um ataque.

Este trabalho trata-se de um ambiente *Honeypot* implementado sobre o sistema operacional GNU/Linux, em uma distribuição Debian com Kernel 2.6. Houve a preferência pela distribuição Debian, devido ao fato de se tratar de um sistema mais seguro e maduro, onde atua como líder majoritário em ambientes de produção e missão crítica em diversos seguimentos, demonstrando robustez, e por consequência sendo o principal alvo de ataques, (MORIMOTO,2009).

Esse trabalho foi desenvolvido valorizando a carência existente hoje, quando se trata de segurança da informação, principalmente voltado a rede de computadores, onde desastres virtuais acontecem diariamente e a maioria com perdas financeiras monstruosas dentre outros danos. Trata-se também de um tema muito polêmico e por outro lado complexo, mas muito compensa as conquistas adquiridas em forma de conhecimentos quando dedicado a seu estudo. Outro fator

que contribuiu para o interesse foram os recentes estudos utilizando os termos *Honeypots*.

No Brasil vivencia-se um grande descaso por parte das empresas em investimentos em infraestrutura e principalmente segurança. Segundo VALLE e ULBRICH(2007), muitas falhas em sistemas que permitem a ação de criminosos poderiam ser corrigidas, mas muitas companhias preferem deixar de lado esses problemas, sendo em muitos casos, até bastante displicentes. É ainda citado numa pesquisa feita pela *Módulo Security Solutions*, empresa especializada em segurança, que segundo dados coletados, a segurança da informação é fator importante para 45% dos executivos, sendo que 16% a consideram crítica e 32% a classificam como vital. Os executivos foram apontados como as principais barreiras na implementação de infraestrutura de segurança nas organizações.

VALLE e ULBRICH(2007) relata que o mundo de uma forma geral tem sofrido uma avalanche de ataques e problemas envolvendo segurança em redes. É o que revela em uma pesquisa do *Gartner Group,* onde 2/3 dos servidores Web no mundo podem ser invadidos de alguma forma, e um outro fator que estimula a atividade *Hacker* é a ampla disponibilidade de ferramentas de ataque na Internet, onde qualquer adolescente com tempo livre e conhecimentos técnicos medianos consegue encontrar as informações e os softwares necessários para uma invasão, mas o principal motivo ainda é a impunidade.

Os índices estatísticos apresentados no decorrer desse trabalho vão de encontro a afirmação de ULBRICH e VALLE (2007), onde os ataques a servidores Web se sobressaem dentre os demais, e além dos serviços Web, serão analisados outros servidores como: *smtp, pop3, squid, ftp, serviço echo*, de forma a levantar informações relevantes como: origem dos ataques, comandos utilizados, usuários e senhas mais utilizados, horário de maior pico de ataques e o grau de reincidência de países em ataques, sobre o conjunto desses servidores em um só ambiente no decorrer de 44 dias.

Para que o levantamento dos dados acima mencionados se concretize, utilizou-se de domínios falsos, criados apenas para comprometimento com o ambiente, com o auxílio do serviço NO-IP, que é muito explorado na Internet para esses fins. Mediante essas informações acredita-se que o trabalho apresentado

desperte nos acadêmicos o interesse em segurança de computadores, servindo de base para trabalhos futuros, já levantando os conceitos inicias de segurança de redes e os testes que podem ser realizados.

2. REFERENCIAL TEÓRICO

2.1 REDES DE COMPUTADORES

2.1.1 O modelo de referência TCP/IP

O protocolo conhecido como TCP/IP (Transmission Control Protocol/ Internet Protocol) é atualmente o padrão utilizado no conceito estabelecido por diversos autores e ressaltado também por ULBRICH e VALLE(2007). Foi concebido justamente para trabalhar nas camadas 3(camada de rede) e 4(camada de transporte) do modelo OSI, e é completamente roteável. Sua criação teve propósitos acadêmicos e militares, pois foi utilizado em diversas redes de universidades e de defesa norte-americanas nas décadas de 70 e 80. O protocolo atingiu a fama com a internet e está implementado em praticamente todos os sistemas operacionais existentes no mercado.

TANEMBAUM(1997), relata também sobre o modelo TCP/IP a grande preocupação do departamento de defesa dos EUA de que seus preciosos hosts, roteadores e gateways de interconexão de redes fossem destruídos a qualquer momento. Eles queriam que as conexões permanecessem intactas enquanto as máquinas de origem e de destino estivessem funcionando, mesmo que algumas máquinas ou linhas de transmissão intermediárias deixassem de operar repentinamente, ainda ressalta que seria necessária uma arquitetura flexível, ou seja, capaz de se adaptar a aplicações com requisitos divergentes, como por exemplo a transferência de arquivos e a transmissão de dados de voz em tempo real.

2.1.2 A camada de rede

Segundo ULBRICH e VALLE(2007), o papel da camada de rede é aparentemente simples, pois se consiste em transportar pacotes de um hospedeiro remetente a um hospedeiro destinatário, e para que isso seja possível três importantes funções da camada de rede devem ser levantadas como:

- Determinação do trajeto: A camada de rede deve determinar a rota ou trajeto tomado pelos pacotes ao fluírem de um remetente a um destinatário. Os algoritmos que calculam esses trajetos são chamados, algoritmos de roteamento, onde determina o trajeto ao longo do qual os pacotes fluirão da origem ao destino.
- Comutação: Processo pelo qual um pacote chega à entrada de um roteador. Este deve conduzi-lo até o enlace de saída apropriado. De forma mais clara, pode-se pensar na seguinte situação: um pacote proveniente do hospedeiro H1 que chega ao roteador R1 deve ser repassado ao roteador seguinte no trajeto até hospedeiro H2.
- Estabelecimento de conexão: para que a comunicação entre os nodos envolvidos aconteça, torna-se necessário a compreensão da representação de 3 vias, conhecido como three-way handshake, que é composto por três flag`s: SYN,SYN-ACK,ACK.

2.1.3 A camada inter-redes

Todas as necessidades de demanda e desenvolvimento de um modelo levaram a escolha de uma rede de comutação de pacotes baseada em uma camada de interligação de redes sem conexões. Essa camada, chamada de inter-redes, integra toda a arquitetura. Sua tarefa é permitir que os hosts injetem pacotes em qualquer rede e garantir que eles trafegarão independentemente até o destino (talvez em uma rede diferente). Eles podem chegar até mesmo em uma ordem

diferente daquela em que foram enviados, obrigando as camadas superiores a reorganizá-los, caso a entrega em ordem seja desejável. Observe que, nesse caso, a expressão "inter-redes" é usada em sentido genérico, muito embora essa camada esteja presente na Internet.

Na definição de TANEMBAUM(1997), essa camada define um formato de pacote oficial e um protocolo chamado IP. A tarefa da camada é entregar pacotes IP onde eles são necessários. O roteamento de pacotes é uma questão de grande importância nessa camada, assim como a necessidade de evitar o congestionamento.

2.1.4 A camada de transporte

No modelo TCP/IP a camada de transporte está localizada acima da camada inter-redes, com a finalidade de permitir que as entidades pares dos hosts de origem e destino mantenham uma conversação. Segundo TANEMBAUM(1997), o protocolo fim a fim definido foi o TCP, que é orientado a conexão, é confiável, pois permite a entrega sem erros de um fluxo de bytes originário de uma determinada máquina em qualquer computador da camada inter-rede. Esse protocolo fragmenta o fluxo de bytes de entrada em mensagens discretas e passa cada uma delas para a camada inter-redes. No destino o processo TCP receptor volta a montar as mensagens recebidas no fluxo de saída. O TCP também cuida do controle de fluxo, impedindo que um transmissor rápido sobrecarregue um receptor lento com um volume de mensagens maior do que ele pode manipular.

2.1.5 A camada de Aplicação

O TCP/IP é apresentado com uma versão enxuta do modelo OSI, não tendo a representação das camadas de sessão e apresentação, sendo essas englobadas

pela camada de aplicação. A camada de aplicação é a camada que está realmente visível e notada pelo usuário com mais facilidade, pois contém os protocolos de nível mais alto, dentre alguns deles: http,telnet,ftp.

3. *FIREWALL*

3.1 HISTÓRICO E DEFINIÇÃO

Na tradução obtém-se “parede de fogo”, ou seja, um programa de computador que detém autonomia pelo próprio sistema operacional a disciplinar todo o tráfego existente entre o mesmo e a comunicação entre redes. Independentemente dos argumentos conceituais utilizados para se expressar a utilidade e eficiência das ferramentas *firewalls*, estas são sem sombra de dúvida o meio mais seguro de se levar serviços de interconectividade a hosts e redes. Mais seguro que ele, só removendo a interface de rede e desconectando os cabos, de um modo sarcástico de se expressar, (NETO,2004).

Segundo NETO(2004), o primeiro *firewall* divulgado foi desenvolvido pela *Bell Labs*, em meados de 80 sob encomenda da gigante em telecomunicações AT&T, foi desenvolvido com o intuito de filtrar todos os pacotes que entrassem e saíssem na rede corporativa, de modo a manipulá-los de acordo com as especificações das regras previamente definidas pelos cientistas da *Bell Labs*.

3.2 *FIREWALL* FILTRO DE PACOTES

O Linux conta com o apoio da filtragem de pacotes através do *firewall* desde a primeira geração dos *kernels*(1.x). Em meados de 1994, Alan Cox portou-o nativamente para o mesmo. Esse tipo de *firewall* possui a capacidade de analisar cabeçalhos de pacotes enquanto os mesmos trafegam. Mediante essa análise, pode-se decidir o destino de um pacote como um todo. A filtragem, pode então deixar tal pacote trafegar livremente pela rede ou simplesmente parar sua trajetória ignorando-o por completo, (NETO,2004).

NETO(2004) ressalva que essa classe de *firewall* é responsável por filtrar todo

o tráfego direcionado ao próprio *host firewall* ou a rede que o mesmo isola, tal como todos os pacotes emitidos por ele ou por sua rede. Todo esse quadro ocorre mediante a análise de regras previamente inseridas pelo próprio administrador da rede, podendo ser personalizadas a seu critério. Independente de se tratar de filtragem de pacotes, o sucesso de um *firewall* está na forma de sua implementação, ou seja cabe ao administrador sólidos conhecimentos para alcançar sucesso na implementação. Para cada tipo de servidor é aconselhável que se utilize uma arquitetura de *firewall* diferente.

3.3 *FIREWALL* NAT

O NAT é a técnica através da qual roteadores traduzem um endereço falso dentro de uma rede classe A com endereços ditos inválidos, como por exemplo uma rede 10.0.0.x, para um endereço válido para a Internet ou para uma outra rede de classe 200.0.0.x por exemplo, (MARCELO e ALVES, 2003).

A Ilustração 1 demonstra uma disposição clássica de um *firewall/NAT,* onde as requisições originadas dos computadores integrantes da lan necessitam atravessar esse *firewall* para que consigam acesso a rede wan. A linha amarela/verde na Ilustração demonstra essa afirmação.

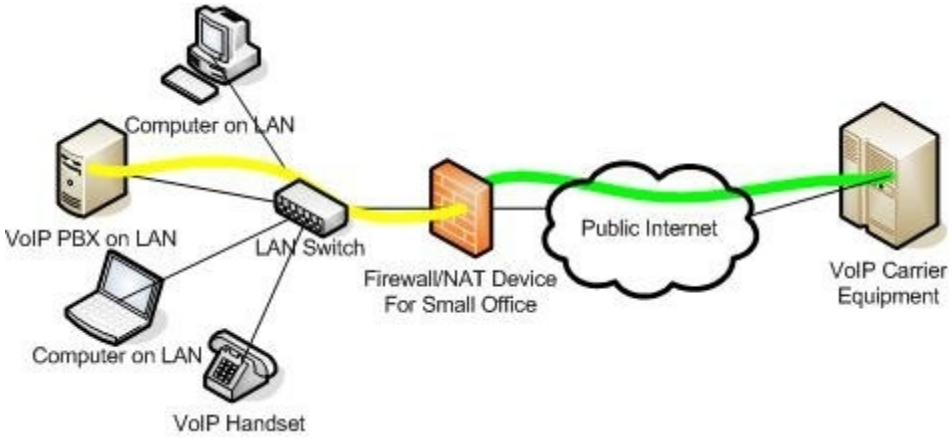

Ilustração 1: Exemplo de um *firewall*/NAT. Obtido em (*WALKERTALKS*, 2010)

3.4 *FIREWALL* HÍBRIDO

Um *firewall* agrega tanto funções de filtragem de pacotes quanto de NAT, na verdade trata-se da união de ambas as classes e não somente de uma classe isolada com propriedades próprias. Na prática como pode-se observar, as finalidades do *firewall* atuando como ferramenta de segurança de sistemas e redes foram ampliadas e novas funções foram agregadas ao projeto inicial, (NETO,2004).

Quando utiliza-se diversas implementações de um *firewall*, seja no trabalho ou a critério de estudos, geralmente utilizamos um *firewall* híbrido, muitas vezes até sem perceber a junção dessas funcionalidades, dando a ele essa nova nomenclatura. Basicamente, um servidor de internet básico agrega as funcionalidades, e o faz ser caracterizado como um *firewall* híbrido.

4. *HONEYPOTS* E *HONEYNETS*

Estudos sobre segurança utilizando os termos *Honeypot* e *Honeynet* são extremamente novos, iniciados a cerca de 15 anos. Até então não se tinha uma definição correta dos termos, pois cada autor propunha sua abordagem, assim logicamente algumas eram mais aceitas e outras nem tanto divulgadas. Algumas das definições levantadas no decorrer dos estudos que hoje respondem pelos termos: sistemas para atrair *Hackers*, sistema para detecção de intrusão.

De acordo com SOUZA(2005), a definição mais aceita é a de *Lance Spitzner,* que define *Honeypot* como: “São recursos de segurança, cujos valores podem ser sondados, atacados ou comprometidos”. Embora esta definição pareça um tanto vaga, ela permite que utilizemos um *Honeypot* para deter ataques, detectar ataques, capturar e analisar ataques automatizados como *worms*, além de fornecer informações importantes sobre a comunidade *Hacker*.

SPITZNER(2002), autor de livros e muito respeitado pela comunidade *Hacker* no mundo inteiro se destaca pelos estudos aprofundados em segurança, prova disso a definição mais aceita do termo *Honeypot* parte do próprio, onde ele consegue descrever de maneira clara e objetiva o intuito de se estudar, implementar ambientes *Honeypots* e *Honeynets*.

> “os *Honeypots* e *Honeynet*s podem ser considerados ambiente de monitoramento de ataques. Diferentes estruturas possam ser construídas na captura de ataques, as quais dependem de diversos fatores tais como o que se deseja monitorar e quais os tipos de informações que se espera obter” (Andrucioli,2005).

ANDRUCIOLI(2005), ainda ressalta algumas características importantes a serem levadas em consideração no uso de tal tecnologia como:

- Tráfego de ataques separados de uma rede real de produção;
- Emulação de serviços e/ou sistemas, quando necessário;
- Utilização de diversas ferramentas com o objetivo de capturar informações relevantes;

- Possibilidade de estudar o atacante e suas técnicas, antes e após uma invasão bem sucedida;
- Captura completa de todos os pacotes envolvendo um ataque;

De acordo com BELCHIOR et al(2004) uma definição para *Honeynet*: é uma ferramenta que consiste de uma rede projetada para ser comprometida e desta forma se observar o comportamento dos invasores, suas táticas, ferramentas e motivações. Assim tem-se de forma clara os conceitos dessas novas tecnologias para o estudo da segurança como um todo.

4.1 NÍVEIS DE ENVOLVIMENTO

O envolvimento de um atacante em um *Honeypot* é caracterizado por níveis segundo ANDRUCIOLI(2005), onde o grau de interação do atacante com a máquina em questão é avaliado. São exemplificados em 3 níveis: baixa interatividade, média interatividade e alta interatividade

4.1.1 *Honeypot* de baixa interatividade

Caracteriza-se por falsos serviços gerados por um *Honeypot*. Nesse nível de interação o atacante possui um mínimo envolvimento com o *Honeypot*, mais os dados coletados traduzem ricas informações e são de total proveito. Segundo MARCELO e ALVES(2003), o *Honeypot* de baixa interatividade pode ser descrito como um *listener* TCP/UDP, aguardando conexões em uma determinada porta e respondendo ao atacante com respostas falsas.

4.1.2 *Honeypot* de média interatividade

Nesse nível a interação entre o atacante e *Honeypot* se torna maior, mais ainda não equivale a um sistema real, pois da mesma forma que aumenta interação, por outro lado cresce também os riscos, o que exige um maior cuidado em relação as ferramentas de *scripts* que interagem com o atacante, (MARCELO e ALVES, 2003).

MARCELO e ALVES(2003) ainda complementam que esse ambiente possui similaridade com o de baixa interação sendo a principal diferença é que o *Honeypot* irá simular com mais detalhes um ambiente falso.

4.1.3 *Honeypot* de alta interatividade

Nesse nível de envolvimento não existe emulação de serviços, onde os *Honeypots* são configurados com sistemas operacionais e serviços reais como qualquer máquina conectada a internet. O fato de todos os serviços serem reais permitem uma interação maior entre o atacante e o ambiente, assim sendo realizado um registro completo de todos os passos do atacante, e a obtenção de todos os dados possíveis de ataque.

Exige-se por parte do ambiente um pouco mais de conhecimento na implantação do mesmo, pois como nada é emulado, tudo fica exposto de uma forma muito clara ao invasor, assim alguns cuidados devem ser tomados durante a instalação, configuração e monitoração desse ambiente como:

- Quando um *Honeypot* dentro da *Honeynet* estiver comprometido, este deve ser imediatamente bloqueado para que ataques a outras redes não se originem.
- A automação desse processo, deverá ser feita sem que intrusos suspeitem disso.
- Ao realizar a captura dos dados relevantes aos eventos de segurança, estes

devem ser armazenados de forma segura, não sendo feito de forma alguma dentro da própria *Honeynet*, pois estaria vulnerável aos invasores.

Cuidados como estes devem ser tomados para assegurar a estabilidade de um ambiente e para garantia de sucesso na captura das informações sem demais imprevistos, pois um ambiente bem configurado, bem parametrizado é a primeira premissa para se estudar *Honeypots* e *Honeynet*s (ASSUMÇÃO, 2009).

5. CLASSIFICAÇÃO

5.1 *HONEYPOT* DE PRODUÇÃO

Pesquisadores alegam que *Honeypots* são importantes também para prevenção de ataques as redes reais, por adicionarem alguns fatores, dentre eles: o tempo e os recursos gastos na tentativa de ataque a um *Honeypot*, uma vez que a rede de produção poderia nessa altura estar sendo comprometida pelo atacante. Segundo SOUZA(2005), uma clássica tradução de *Honeypots* de produção:

> "são sistemas que aumentam a segurança de uma organização em específico e ajuda a mitigar riscos. São mais fáceis de construir porque requerem menos funcionalidades. Usualmente possuem as mesmas configurações que a rede de produção da organização e transportam para ela todo o aprendizado obtido com os ataques sofridos" (Souza, 2005).

Uma vez identificado o ataque, inicia-se o processo de análise, de forma a levantar como aquele invasor conseguiu penetrar no sistema, o que ele causou e como causou, origem do ataque, dentre outros. Enfim, toda essa informação é de suma importância para administrador da rede, sendo assim deve ser toda coletada e analisada para que conclusões sejam tomadas a partir desses resultados.

5.2 *HONEYPOT* DE PESQUISA

Os *Honeypots* de pesquisa oferecem uma estrutura voltada a compreender a comunidade *Hacker*, não apenas estudar as ferramentas, mas sim tentar diagnosticar a metodologia e os rastros deixados no sistema durante o comprometimento. Aqui há realmente um crescimento notável na contribuição de estudos e busca de novas melhorias em todo ambiente e o que geralmente é envolvido por parte do sistema, (SOUZA, 2005).

Segundo ASSUNÇÃO(2009), um *Honeypot* de pesquisa não tem como objetivo primário ser utilizado como uma ferramenta IDS, e sim ser realmente atacado várias vezes, e com isso estudar todos os detalhes de cada ataque, ou seja, cada arquivo que o invasor acessar, cada senha que ele digitar, cada comando, absolutamente tudo foi salvo e estudado.

A grande utilização desse tipo de *Honeypot* acontece nos centros universitários, onde pesquisadores querem entender as ações do início, ao fim e levantar todas as informações em torno das ferramentas que foram utilizadas sendo elas mecanizadas ou não.

5.3 REPRESENTATIVIDADE DE UM *HONEYPOT*

A Ilustração 2 exemplifica o uso de *Honeypot* situado em um ambiente misto, onde se tem rede wireless, uma conexão com a Internet e um servidor com VMware instalado emulando mais servidores, sendo eles: um *firewall* e um *Honeypot*.

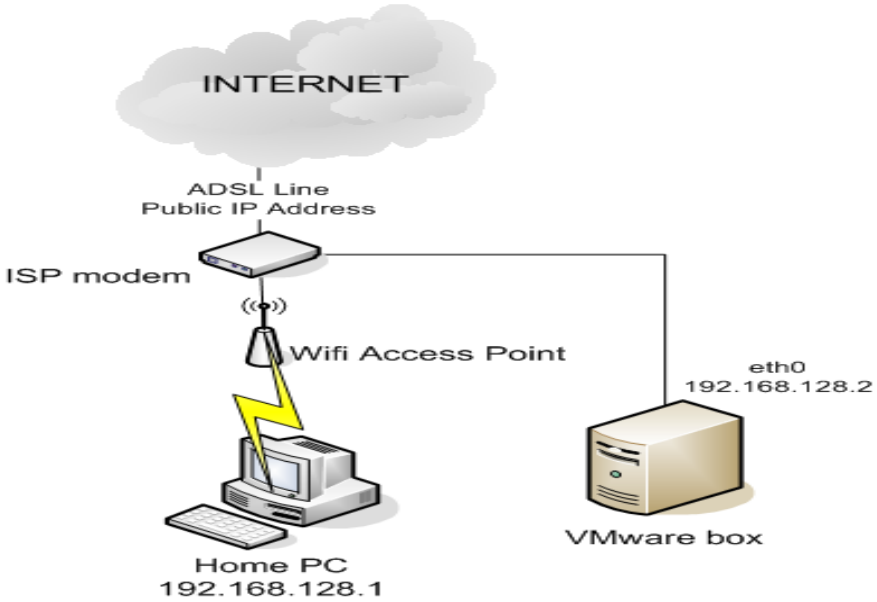

Ilustração 2: Representação de um Honeypot. Obtido em (THE Honeynet PROJECT, 2010)

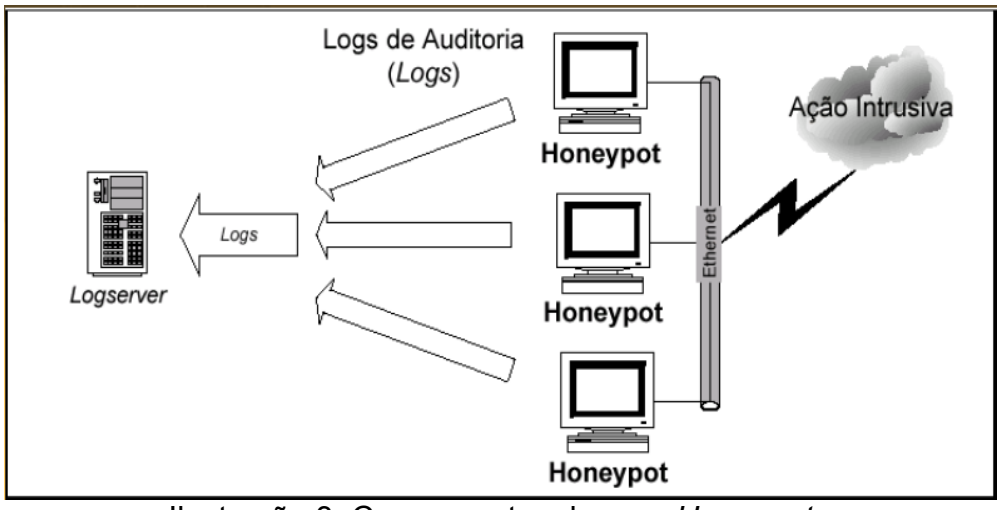

Ilustração 3: Componentes de uma *Honeynet.*
Obtido em (*THE Honeynet PROJECT*, 2010)

A Ilustração 3 exemplifica o modo de disposição de uma *Honeynet* onde toda e qualquer ação intrusiva é desviada a um servidor de *logs*, mantendo o ambiente mais seguro.

As *Honeynet*s se classificam em dois tipos principais: Clássica e Virtual. Abaixo segue exemplificação de cada um dos modelos, afim de facilitar o entendimento por meio da disposição do ambiente.

A Ilustração 4 representa uma *Honeynet* clássica e é composta por sistemas reais, instalações específicas e ambiente composto por sistemas operacionais diversificados. A *Honeynet* está sempre situada por detrás de um *firewall*.

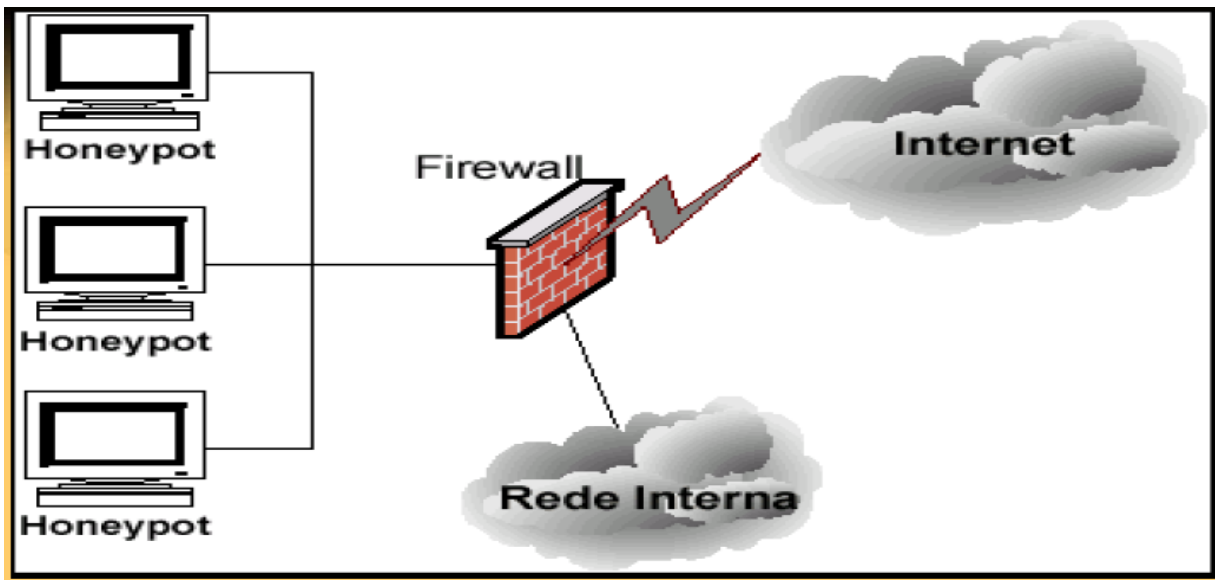

Ilustração 4: *Honeynet* Clássica. Obtido em (*THE Honeynet PROJECT*, 2010)

A Ilustração 5 representa a *Honeynet* virtual que é composta por dois ou mais *Honeypots* virtuais com o auxílio de emuladores, dessa forma todo ambiente é composto por uma única máquina. Existe uma virtualização de todo de todo o ambiente.

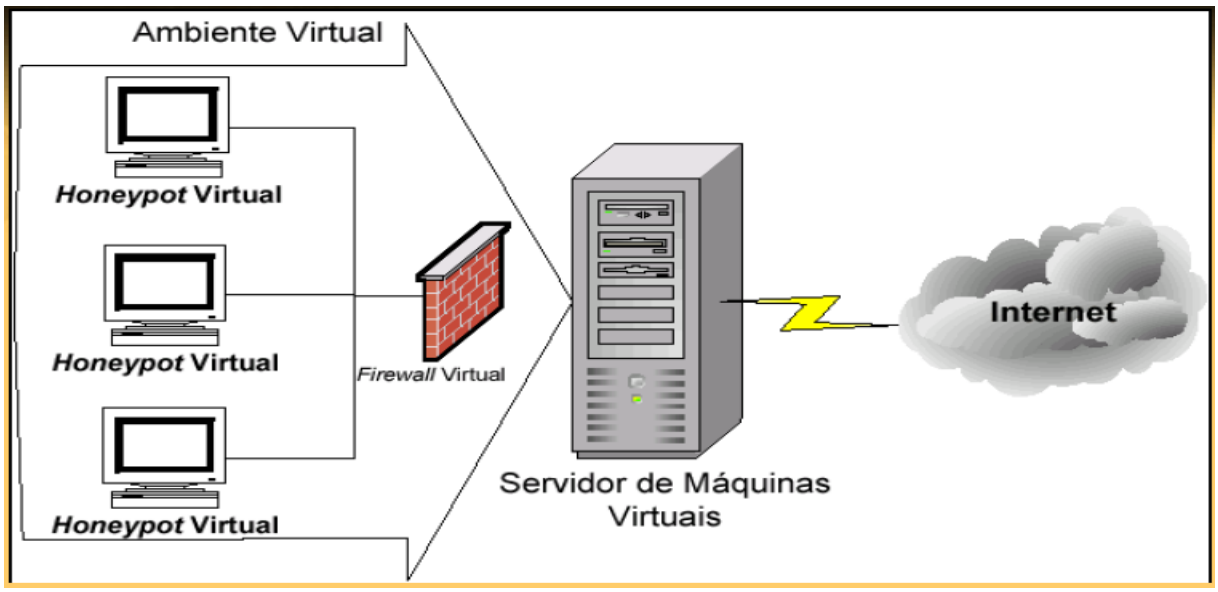

Ilustração 5: *Honeynet* Virtual. Obtido em (*THE Honeynet PROJECT*, 2010)

6. TIPOS DE ATACANTES

Na famosa série “Conhecendo seu inimigo” de *Lance Spitzner*, o conceituado autor apresenta alguns personagens que podem ser capturados por *Honeypots*. No seu livro, segurança em *Linux* existe uma definição da classe de cada um dos atacantes MARCELO(2003).

6.1 *SCRIPT KIDDIES*

É uma espécie de papagaio eletrônico, ou seja, invasor que compra livros ensinando a ser *Hacker*, ou então através da leitura de tutoriais mais que defasados. O mesmo tenta repetir receitas de bolo, ou seja, repete o ataque exaustivamente.

Geralmente esse tipo de invasor não tem conhecimento profundo do que está fazendo, simplesmente fica na esperança de conseguir invadir, caso numa remota possibilidade consiga entrar no site, ou outras formas de acesso, o mesmo não saberá o que fazer, normalmente atacantes com essas características são barrados pelo *firewall* (MARCELO e ALVES, 2003).

6.2 *LAMMER*

Este tipo de atacante é curioso. Apesar de conhecer um pouco mais sobre sistemas operacionais, redes, é também um repetidor de fórmula de invasão/ataque. Normalmente estes atacantes conseguem invadir sites inseguros com facilidade. Esse atacante adora aparecer, gosta de executar as pichações eletrônicas em sites e dizer que foi ele o responsável por aquilo. Normalmente é pego devido a esta arrogância que gera um série de falhas no momento da invasão. No caso de máquinas *Linux*, em muitos casos o comando *history* é um grande delator destes

indivíduos (MARCELO e ALVES, 2003).

6.3 *HACKER*

O melhor de todos, o mais perigoso invasor, o mais comentado no decorrer desse trabalho. O termo *Hacker*, significa um indivíduo que conhece um sistema operacional e o modifica, por exemplo *Linus Torvalds*, criador do *Linux*, é um *Hacker*.

Com isso o termo acabou sendo utilizado para o indivíduo que conhece muito o sistema operacional, sistemas de redes e suas fraquezas. O objetivo do *Hacker* é o conhecimento, ficando oculto em um sistema para utilizá-lo na realização de outros ataques. A informação é o seu bem mais precioso e sua vitória é entrar em um sistema, roubar as informações, plantar um *backdoor* a qualquer custo mantendo-se totalmente oculto (MARCELO & ALVES, 2003).

6.4 *CRACKER*

O guru, aquele que além de ter conhecimentos profundos do sistema e já ter sido um *Hacker* em muitos casos, possui conhecimentos profundos em programação, a ponto de depurar programas, encontrar vulnerabilidades e criar o chamado *exploit*, um programa que explora as vulnerabilidades e permite invadir o sistema alvo, normalmente o *Cracker* tem contatos com outros *Hackers* de sua extrema confiança e libera os *exploits* de forma privada (exploits private) para que os mesmos os testem em campo (MARCELO & ALVES, 2003).

6.5 *CARDER*

O objetivo deste tipo de atacante é roubar cartões de crédito e trocar por informações ou mesmo por dinheiro ou mercadorias. É normalmente um *Hacker* que utiliza seus conhecimentos para este tipo de atividade. No Brasil essa categoria de invasores estão agindo com incidência cada vez maior e são criminosos procurados pela polícia (MARCELO e ALVES, 2003).

6.6 *PHREAKER*

Relato de casos são mais raro em nosso país, por se tratar de um indivíduo com conhecimentos voltados para a telefonia fixa e celular. Sua especialidade é burlar sistema telefônicos, clonar celulares, executar operações que permitam ligações gratuitas nacionais / internacionais. O *Phreaker* conhece eletrônica, principalmente na área de telecomunicações e é capaz de roubar informações de usuários. *Kevin Mitinick,* também foi capturado por esse motivo (MARCELO e ALVES, 2003).

6.7 USUÁRIO

A maior ameaça para qualquer sistema seguro é o usuário do mesmo. Por descuido, desinformação, ou até mesmo vingança, o usuário pode comprometer a estrutura de um sistema com atitudes como: senhas simples, passagem de informação, recebimento de e-mail com vírus, acesso a sites não autorizados, execução de arquivos sem mesmo saber sua procedência. O usuário é ainda responsável pelo que chamamos de *hacking from inside*, ou seja, invasões feitas de dentro da própria estrutura na qual trabalha. O administrador esquece que na maioria das vezes ele pode ser seu calcanhar de Aquiles (MARCELO & ALVES, 2003).

7. TIPOS DE AMEAÇAS

7.1 VÍRUS E WORMS

Segundo MORIMOTO(2006), uma definição de vírus, seria programas espúrios inseridos em computadores contra vontade do usuário e desempenham funções indesejáveis. Alguns vírus têm a capacidade de se reproduzir e infectar outros dispositivos por toda a rede. Já outros não se reproduzem, mas são distribuídos em falsos programas na rede ou em CDs, vendidos em publicações. A cada dia surgem centenas de vírus e o combate a esse tipo de invasão é uma tarefa constante.

As principais vacinas são a instalação de programas antivírus atualizados em todas as estações de trabalho e servidores. É recomendável deixar programas antivírus residentes na memória para proteção em tempo real de qualquer infecção possível. Também se deve restringir as permissões de acesso especialmente a programas executáveis, impedindo que sejam alterados. Deve-se restringir acesso a pastas e diretórios críticos especialmente em servidores. Os usuários devem ser alertados dos riscos que correm ao instalar programas suspeitos ou não autorizados em suas estações de trabalho.

São vírus que fazem cópias do seu próprio código e as enviam para outros computadores, seja por e-mail ou via programas de bate-papo, dentre outras formas de propagação pela rede. Eles têm se tornado cada vez mais comuns e perigosos porque o seu poder de propagação é muito grande.

Do lado dos servidores, os *worms,* que são uma das pragas mais perigosas atualmente, eles unem o conceito de vírus e trojan utilizando a internet para se propagarem automaticamente*,* os mais recentes exploram vulnerabilidades dos serviços ou programas instalados no servidor para se infiltrar e fornecer acesso ao atacante. Além disso, uma vez instalados eles começam a procurar novos endereços vulneráveis para atacar.

Já do lado das estações, os *worms* mais comuns exploram vulnerabilidades

dos programas de recebimento de e-mail para se infiltrarem e se propagarem para todas os endereços cadastrados no cliente de e-mail, além de se anexarem automaticamente em todas as mensagens enviadas.

7.2 CAVALO DE TRÓIA & *BACKDOORS*

Os cavalos de *tróia* são programas instalados de forma indevida pelo próprio usuário, ou diretamente pelo *Hacker*. Uma vez instalado, o cavalo de tróia retorna informações para o *Hacker* dá acesso diretamente ao seu computador (Strebe e Perkins, 2002).

Os tipos mais úteis de Cavalos de tróia, do ponto de vista dos *Hackers*, são as chamadas “porta dos fundos” (*Backdoors*). São programas que uma vez instalados abrem brechas invisíveis ao usuário no sistema, geralmente são difíceis para o usuário notar esse tipo de interação com o sistema, somente é notado quando o sistema operacional fica comprometido de alguma forma, como lentidão e mensagens estranhas.

Há programas projetados como *NetBus*, *BackOrifice, BO2K*, dentre outros, que são programas benignos e que podem ser aproveitados para se obter controle de um sistema como o *NetCat, VNC, PcAnywhere*. Esse tipo de ameaça devem ser pequenas e de rápida instalação, além de terem que desempenhar o principal objetivo, executar invisivelmente.

7.3 *BUFFER OVERFLOW*

Representam uma classe de ataques que exploram uma fraqueza comum a todo software, ou seja, se aproveitam do fato de que a maior parte dos programas reservam blocos de memória de tamanho fixo para criar um área de armazenamento temporário chamada *buffer*, na qual os programas processam essas informações

oriundas da rede, esses *buffers* são programados com tamanhos fixos (STREBE & PERKINS, 2002).

Esse tipo de falha acontece no dado momento em que uma mensagem mente sobre seu tamanho, ou é criada maior que o tamanho máximo permitido. Esclarecendo o entendimento, pode-se imaginar uma mensagem que informa o tamanho de 240 *bytes*, mas que na verdade possui 256 *bytes*, assim o serviço que a recebe poderá alocar um *buffer* somente de 240 *bytes*, mas depois tentar copiar 256 nesse *buffer*. Aí que surge o problema, justamente nesse 16 *bytes* restantes, onde eles serão sobrescritos com o que a mensagem contiver.

Os *Hackers* aproveitam desse problema incluindo código na linguagem de máquina nesses 16 *bytes*, sendo ainda mais perturbador o fato do software ser escrito frequentemente de tal modo que a execução do código começa exatamente depois do final do *buffer*, permitindo assim que os *Hackers* executem códigos no contexto de segurança do serviço sendo executado (Strebe & Perkins, 2002).

7.4 *FINGER*

O protocolo *finger* pode oferecer informações suficientes sobre os usuários que permitam aos *Hackers* adivinhar nomes e senhas, geralmente era adotado nos primórdios da Internet como uma forma de examinar endereços de e-mail.

Geralmente os Hackers usam esse comando para sondar um servidor *finger,* procurando informações sobre os usuários de um sistema, dentre nomes de contas e mesmo dicas de senhas podem ser obtidas com frequência (STREBE & PERKINS, 2002).

7.5 *BRUTE FORCE*

Uma vez identificado o *host* alvo, o *Hacker* irá tentar levantar informações de

serviços que estão em execução, como por exemplo, *telnet* e *ftp*. A maior parte dos serviços de um sistema são protegidos por senha, dessa forma o que o atacante vai tentar é uma sequência de supostas adivinhações para tentar burlar aquele serviço, geralmente essa sequência já está pré-definida, onde ele possui uma listagem de assinaturas com usuários mais comuns e senhas, daí faz-se jus ao nome da ameaça, onde o atacante irá tentar todas suas forças e informações de usuários e senhas disponíveis para ter acesso aquele sistema. É uma das ameaças mais comuns em sistemas UNIX (STREBE & PERKINS, 2002).

7.6 *SPYWARES*, *ADWARES* E *MALWARE*

Segundo informações de um dos consagrados sites da comunidade Linux, http://epidemiclinux.org, o *spyware* consiste em um programa automático de computador, que recolhe informações sobre o usuário, sobre os seus costumes na Internet e transmite essa informação a uma entidade externa na Internet, sem o seu conhecimento nem o seu consentimento. É um software espião, diferem dos trojans por não terem como objetivo que o sistema do usuário seja dominado, seja manipulado, por uma entidade externa, por um *cracker*.

Adware é um programa indesejável mas nem sempre malicioso. Costuma-se incluir os *adwares* no estudo dos *spywares*, pois assemelham-se na sua forma de infecção e na sua forma de desinstalação. Seriam como se fossem um subgrupo dos *spywares*. Os *adwares* são conhecidos por trazerem para a tela do usuário algum tipo de propaganda embutida, geralmente *banners* ou *pop-ups*, para gerar alguma renda aos autores.

O termo *malware* é proveniente do inglês *malicious software*, é um software destinado a se infiltrar em um sistema de computador alheio de forma ilícita, com o intuito de causar algum dano ou roubo de informações. Vírus de computador, cavalos de tróia e *spywares* são considerados *malware*. Também pode ser considerada *malware* uma aplicação legal que por uma falha de programação execute funções que se enquadrem na definição citada.

8. ESTATÍSTICAS DE ATAQUES

8.1 A ATUAÇÃO DO CERT.br NO CENÁRIO DE SEGURANÇA BRASILEIRO

É um grupo de resposta a incidentes na Internet brasileira, mantido pelo NIC.br, do comitê gestor da internet no Brasil. Sua responsabilidade é tratar incidentes de segurança em computadores que envolvam redes conectadas a Internet brasileira. CERT.br atua como um ponto central para notificações de incidentes de segurança no Brasil, provendo a coordenação e o apoio no processo de resposta a incidentes e quando necessário colocando as partes envolvidas em contato (http://www.cert.br/sobre).

Além de todo o trabalho realizado pelo CERT.br no processo de tratamento de incidentes, talvez o mais importante ou melhor, mais próximo ao desenvolvimento desses trabalho é o estabelecimento e crescimento dos CSIRTs no Brasil, com o objetivo estratégico de aumentar os níveis de segurança e da capacidade de tratamento de incidentes das redes conectadas a Internet no Brasil.

Toda essa coleta de informações realizadas pelos CSIRTs espalhados por boa parte do Brasil é realizada utilizando *Honeypots* de baixa interatividade, ou seja, os mesmos fundamentos adotados para esse trabalho.

8.2 PRINCIPAIS ATIVIDADES DESENVOLVIDAS PELO CERT.br

8.2.1 Tratamento de incidentes

- Dar suporte ao processo de recuperação e análise de ataques e de sistemas comprometidos.

- Estabelecer um trabalho colaborativo com outras entidades, como outros CSIRTs, empresas, universidades, provedoras de acesso e serviços de Internet e *backbones*.
- Manter estatísticas públicas dos incidentes tratados e das reclamações de *spam* recebidas.

8.2.2 Treinamento e conscientização

- Oferecer treinamentos na área de tratamento de incidentes de segurança, especialmente para membros de CSIRTs e para instituições que estejam criando seu próprio grupo.
- Desenvolver documentação de apoio para administradores de redes Internet e usuários.
- Realizar reuniões com setores diversos da Internet no Brasil, de modo a articular a cooperação e implantação de boas práticas de segurança.

8.2.3 Análise de tendência de ataques

- Aumentar a capacidade de detecção de incidentes, correlação de eventos e determinação de tendências de ataques no espaço da Internet brasileira, através da manutenção de uma rede de *Honeypots* distribuídos em diversas redes do país;
- Obter, através de *Honeypots* de baixa interatividade, dados sobre o abuso da infraestrutura de redes conectadas à Internet para envio de spam.

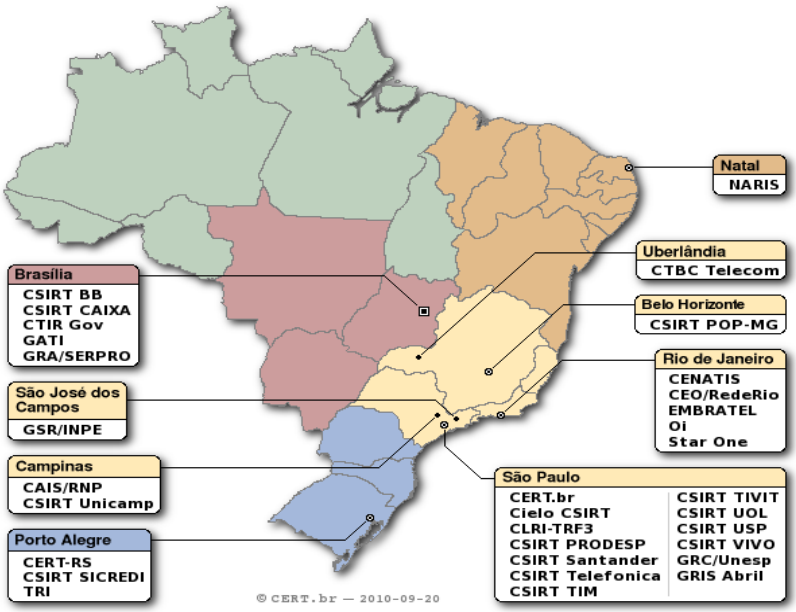

Ilustração 6: Atuação dos CSIRTs no Brasil.
Obtido em (CERT.br, 2010)

Na Ilustração 6 tem-se o nível de expansão dos CSIRTs no Brasil, com informações atualizadas ate o mês de setembro de 2010. Fica claro como exposto, que diversas empresas e órgãos contribuem com o levantamento dessas informações. A região Sudeste lidera no ranking.

É interessante que organizações como as empresas de telefonia e instituições bancárias se interessam em colaborar com os CSIRTs. Um dos motivos a despertarem tal interesse deve estar ligado a quantidade de ameaças que sofrem no decorrer dos meses, dessa forma encontram um meio para contribuírem para melhorias futuras na redução de incidentes.

O CERT.br possui uma abrangência maior ainda pelo Sudeste, devido a sua base central estar localizada na cidade de São Paulo. Mais pelo excelente trabalho que tem desenvolvido até o momento, a expansão para outras regiões do país será de forma mais rápida, devido a participação de empresas de destaque estarem envolvidas no projeto reforça o propósito do CERT.br mais uma vez como um órgão sério e que existe realmente com objetivo de somar forças para a melhoria de segurança em redes de computadores no nosso país.

8.2.4 Estatísticas reportadas ao CERT.br

Além de desmembrar toda informação antes de ser exposta a consultas, o CERT.br levanta várias estatísticas de tendências de ataques, os de maior ocorrência, de forma a esclarecer como tem sido a atuação das ameaças na Internet como um todo e pode-se levantar quais as tendências que ainda devem-se manter nos próximos anos.

A ilustração 7 expõe uma tendência quase que natural, onde com ao passar dos anos o número de computadores vem aumentando, consequentemente tem-se um número maior de usuários conectados a Internet, fazendo aumentar as ameaças.

O mais curioso é a diferença do ano de 2009 para o ano de 2010, mesmo levando em consideração o fato da coluna no gráfico de 2010 não levar em consideração o último trimestre do ano, mas se a proporção se manter, temos resultados positivos, ou seja, um índice menor de incidência de ataques no ano de 2010.

Nota-se ainda três momentos interessantes, onde no ano de 2002 o índice de incidentes reportados se destacou muito em relação ao ano de 2001, praticamente dobrou. Um outro pico mais que notável é no ano de 2006, que praticamente triplicou em relação ao ano de 2005.

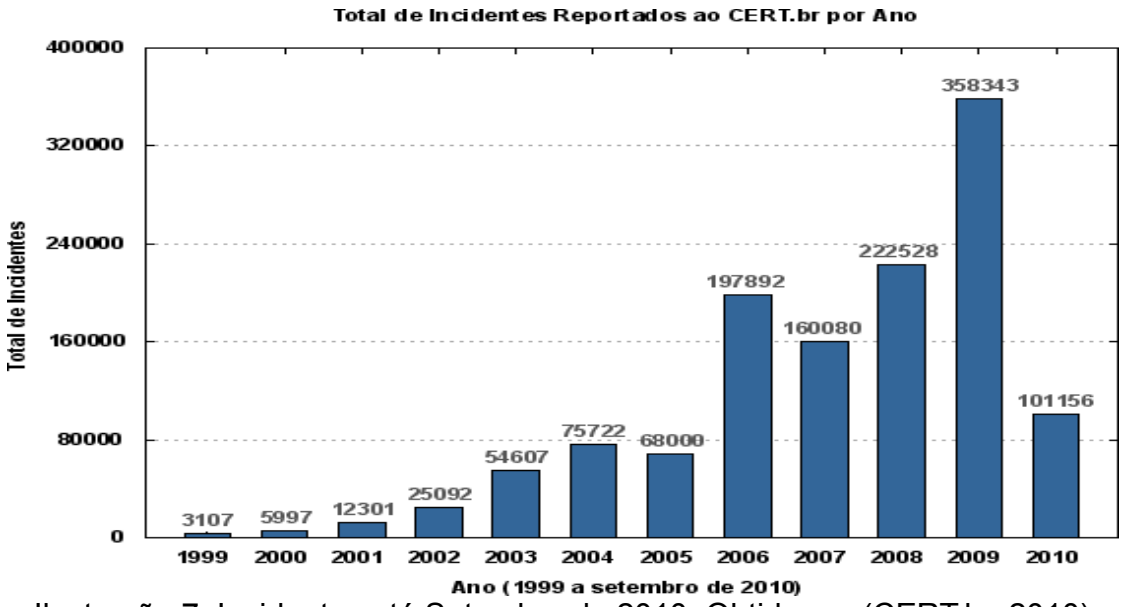

Ilustração 7: Incidentes até Setembro de 2010. Obtido em (CERT.br, 2010)

Segundo dados estatísticos levantados pelo CERT.br através da ilustração 8, a incidência maior de tipo de ataques no período avaliado de Julho a Setembro de 2010, acontece por meio de *scans*, fraudes, seguido de ataque Web, e outros também descritos, mais com nível de incidência menor.

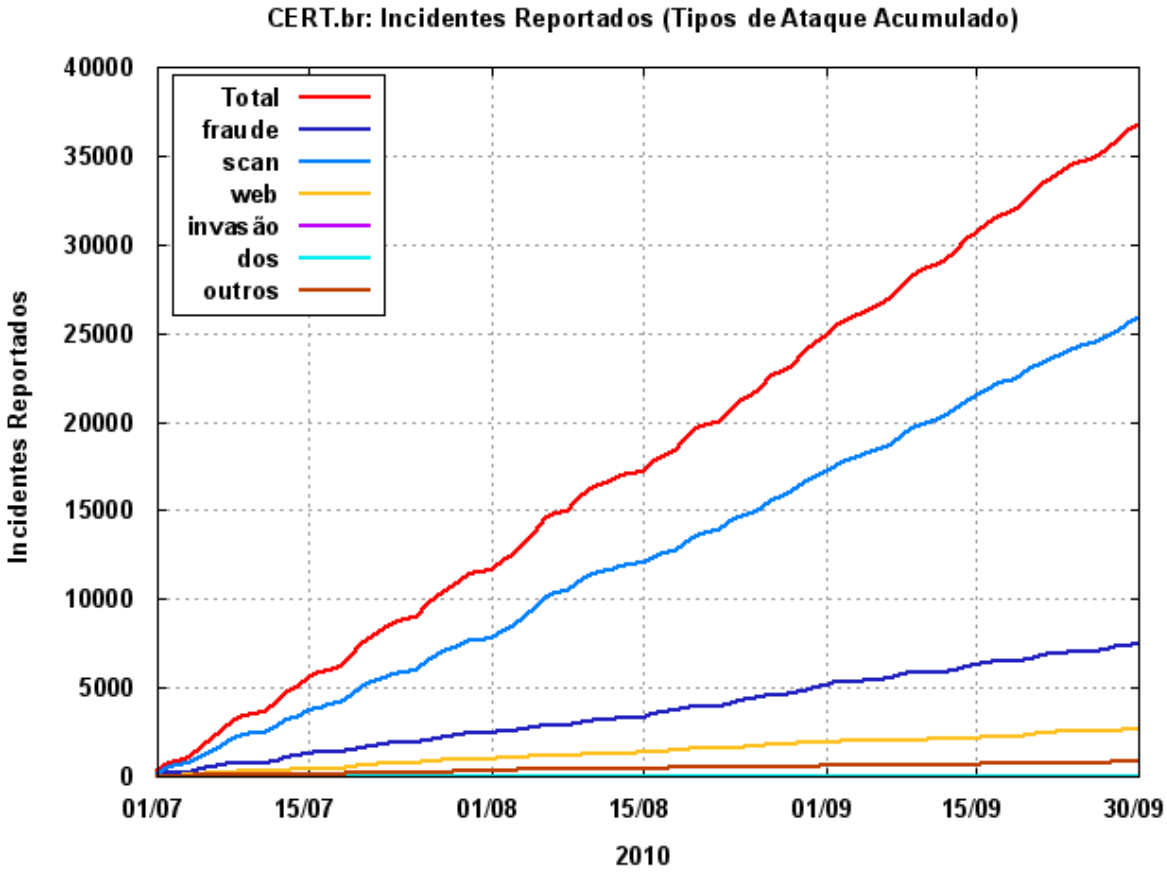

Ilustração 8: Tipos de ataques acumulativos de Julho a Setembro. Obtido em (CERT.br, 2010)

Como mencionado no capítulo 7, sobre os tipos de ameaças, pode-se perceber a tendência das ameaças que mais crescem, e a partir daí mobilizar grupos de pesquisas até mesmo nas academias para desenvolver formas mais eficazes na busca constante de redução desses tipos de ataques. O ataque Web por exemplo, ocupa o terceiro lugar no ranking, e está se tornando cada vez mais comum, sendo que suas consequências são muito ruins e podem gerar diversas perdas a organização ou entidade em questão que se passa por vítima de um desses ataques.

Nos dados coletados pelo CERT.br, mostra que os índices de incidentes

reportados no segundo e terceiro trimestre desse ano, servem para traçar um comparativo, como de acordo com as ilustrações 9 e 10.

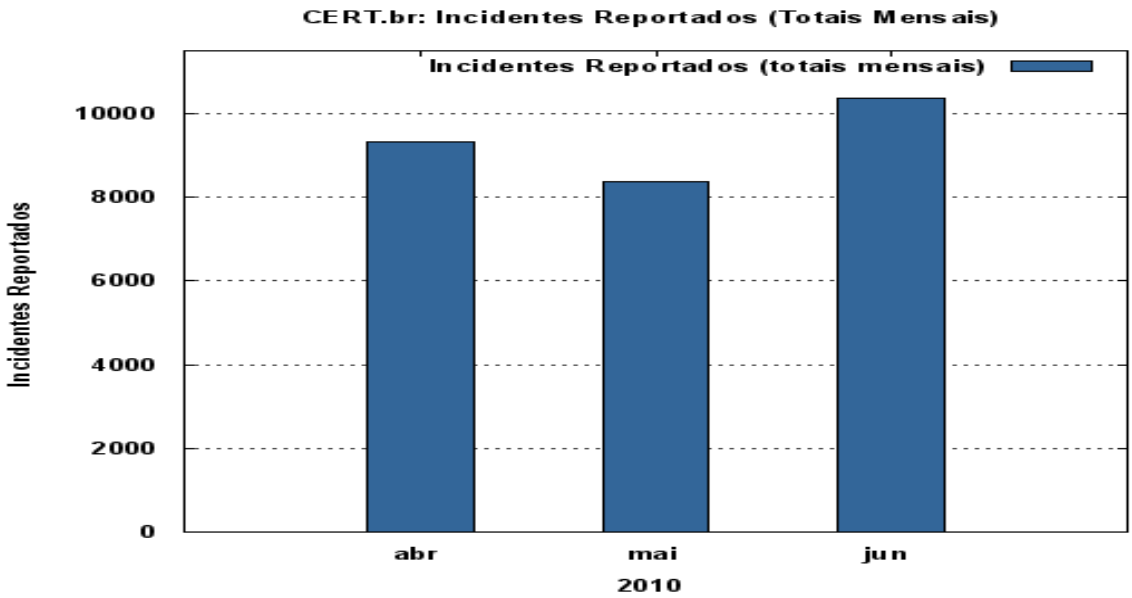

Ilustração 9: Incidentes reportados segundo trimestre 2010.
Obtido em (CERT.br, 2010)

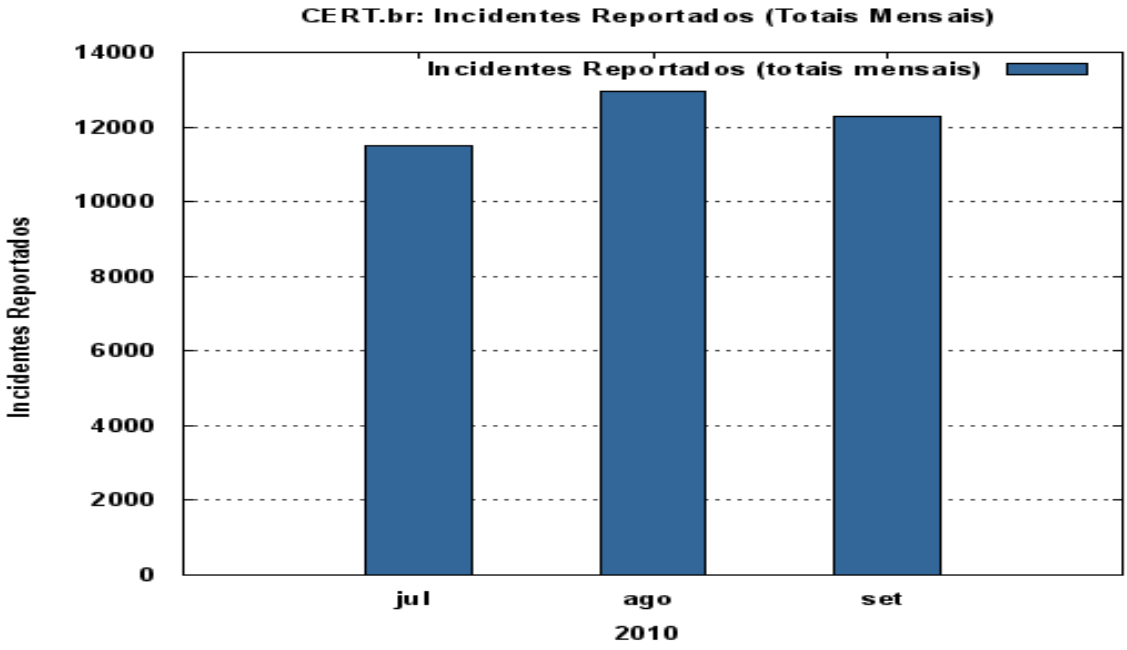

Ilustração 10: Incidentes reportados no terceiro trimestre 2010.
Obtido em (CERT.br, 2010)

Um índice muito alto de incidentes se comparado a ilustração 9 com o

trimestre anterior de 2010, onde o pico mais alto passou de 10.000 incidentes ainda no mês de junho, o mês final da etapa avaliada.

A ilustração 10 apresenta realmente a continuidade aos índices de crescimento nos ataques, onde no mês de julho já reflete um crescimento se comparado ao mês de Junho, e na sequência dos dois próximos meses o aumento é notório, com um destaque maior em agosto e uma incidência um pouco menor em setembro. De uma forma geral pode-se avaliar que a expansão dos incidentes está sendo infelizmente gradativa, baseando-se nesses gráficos, tem-se um indicativo que em 2011 teremos um índice ainda maior de incidentes reportados no Brasil.

De acordo com a Ilustração 11, percebe-se que a incidência dos ataques sofre variações no decorrer dos dias da semana, onde nos finais de semana, compreendendo os dias de Sábado e Domingo geralmente o nível de incidência é menor, de acordo com o gráfico ilustrativo fornecido pelo CERT.br.

O que esclarecer o porque do índice nos finais de semana serem menores, pode estar ligado diretamente a ausência dos usuários diante dos computadores nesse período. Geralmente possuem outros atrativos além do computador.

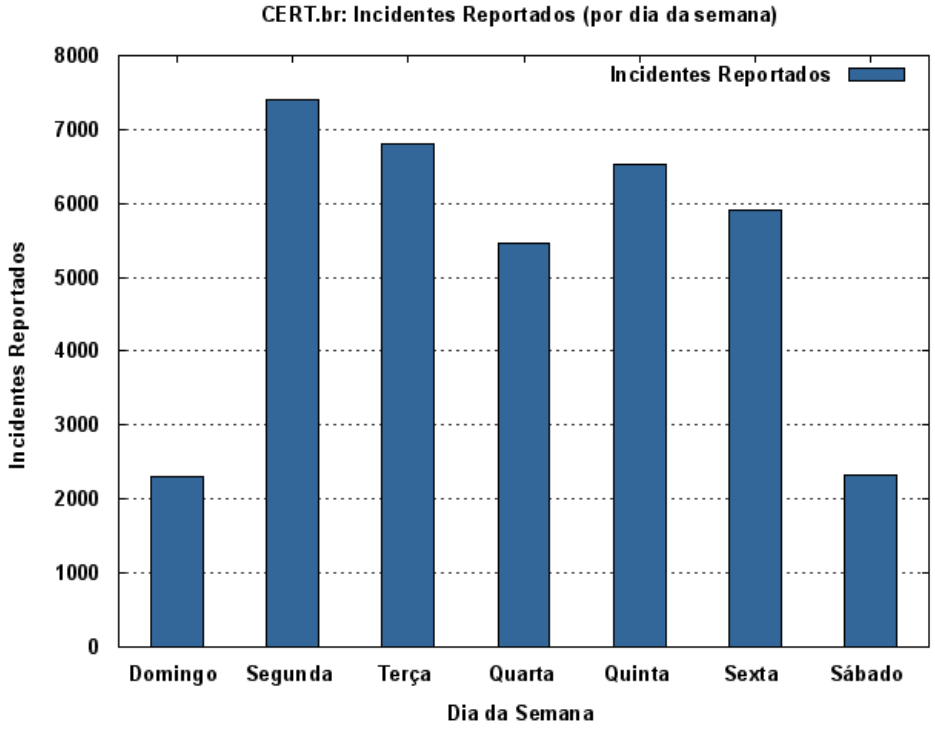

Ilustração 11: Estimativa de ataques pelos dias da semana.
Obtido em (CERT.br, 2010)

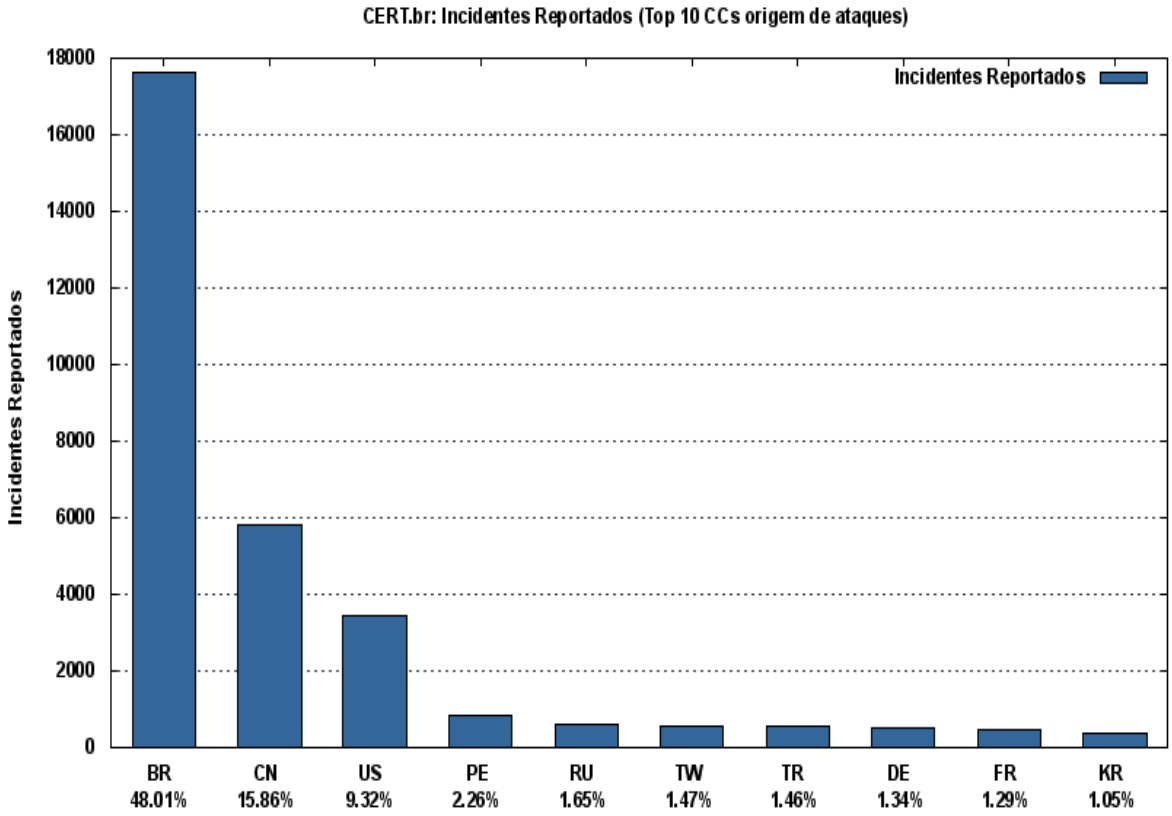

Ilustração 12: Dez países que mais originam ataques por todo mundo.
Obtido em (CERT.br, 2010)

Nota-se na Ilustração 12 o índice de ataques reportados originado pelos 10 países que mais causam transtornos de alguma forma na rede mundial de computadores, onde o Brasil mais uma vez se destaca na liderança, acompanhado pelo Canadá e Estados Unidos.

O índice que traz o Brasil como primeiro colocado está na casa de 48,01% que representa um total de quase 18.000 ataques originados. Isso reforça o fato de o Brasil ser o país que mais possui problemas com segurança. Os demais países se mantém em uma média um pouco abaixo de 1.000 ataques.

Outro fato interessante ressaltado pela ilustração 12, é que pode-se traçar um comparativo com a ilustração 15, que é componente do capítulo 11, onde é demonstrado alguns resultados. Dentre os resultados mostrados pelo ilustração 15, dos 5 países citados, 4 deles que são: Brasil, USA, China e Taiwan estão presentes dentre os 10 citados pelo CERT.br na ilustração 12, isso comprova a vericidade dos resultados obtidos.

9. METODOLOGIA

Honeypots tem a tradução de “pote de mel”, com o objetivo de atrair *Hackers*, de forma que seus rastros sejam estudados e/ou rastreados para se ter a informação de quem está tendo contato com a rede sem prévia autorização. Esse trabalho trata-se de *Honeypot* de baixa interação, onde os serviços são falsos, tudo é emulado com grande similaridade a um serviço real.

Foi realizado uma análise sobre as invasões a esse *Honeypot*, de forma a levantar informações importantes como: usuários e senhas mais utilizados em determinados serviços; IP de origem, horário onde ocorre maior incidência de ataques; comandos mais executados; serviços mais temidos; número de tentativas e serviços explorados, oriundos de um mesmo IP em relação a data. Sobre essas informações sendo feito uma análise de forma estatística para demonstrar as informações.

A instalação do servidor *Honeypot* foi realizada utilizando-se da ferramenta *Honeyperl*, que se dará após a configuração básica do servidor, de maneira a conhecer primeiramente os arquivos de configurações e todas as parametrizações para um correto funcionamento desde a escolha do *hardware* até as análises finais.

Utiliza-se para esse experimento um microcomputador Intel Pentium D 3.2GHZ, com 2 GB de memória ram DDR2 dispondo de 1 interface de rede 10/100 Mbps e um *Hard Disk* de 250GB com as seguintes estruturas de sistema:

- 300 Mb de espaço para partição /boot;
- 20 GB de espaço para a partição /;
- 227.7 GB de espaço para partição /system;
- 2 GB de espaço para partição de troca.

9. 1 ESTUDO DE CASO UTILIZANDO A FERRAMENTA HONEYPERL

Segundo BORGES E BENTO(2006), o *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 adicionar funcionalidades aos que já foram lançados e testados.

O *Honeyperl* é um *Honeypot* de baixa interação que simula diversos serviços como:

HTTP: Servidor web, é o responsável por atender as requisições de páginas solicitadas pelos clientes retornando o conteúdo requisitado, está sendo emulado com apache na versão 2.2.14, o apache que atualmente é o servidor Web mais popular na Internet, com aprovação em 60% de todos servidores Web. Segundo HATCH & KURTZ (2002), essa popularidade é devido a vários fatores como:

- o apache é configurável;
- é extensível;
- o apache é de fonte aberto;
- é gratuito.

Ainda ressalta que ele é relativamente seguro, possui um histórico de comprometimentos a segurança, mas quando essas falhas são descobertas, as correções são disponibilizadas na Internet quase que imediatamente. Isso o torna diferente dos demais servidores Web, principalmente os proprietários, onde os comprometimentos na segurança demoram para serem corrigidos.

Como o servidor Web é um dos principais alvos de ataques, nesse trabalho a implementação utilizará o servidor apache, por se tratar de diversas vantagens se comparado com os outros servidores como relatado acima.

FTP: Protocolo utilizado para transferência de arquivos de um hospedeiro a outro.

O servidor emulado foi o *wuftp*.

SMTP: Protocolo utilizado para envio de mensagem, utiliza a porta 25. O trabalho foi realizado utilizando o servidor de envio de mensagens virtualizando *sendmail*. Outros servidores como *qmail*, *postifix* e *exchange* poderiam ser utilizados.

POP3: Protocolo utilizado para recebimento de mensagem, utiliza a porta 110. O servidor virtualizado utilizado foi o *qpopper*, podendo variar com o *teapop*.

ECHO: Serviço que permite escrita na tela, diversas mensagens de erros e

avisos são apresentadas por esse comando, utiliza a porta 7.

TELNET: Serviço de acesso remoto, geralmente escuta na porta 23. Utilizado o servidor *telnetd* virtualizado.

SQUID: Serviço utilizado para *proxy/cache* da rede, funciona por padrão na porta 3128.

O *Honeyperl* proporciona alternar os servidores entre os *fakes*, como relatado, porém os serviços que foram escolhidos para serem emulados são os mais utilizados em todas as implementações da ferramenta, dessa forma pode-se obter resultados mais rápidos e precisos, devido a demanda desses servidores. Assim com um conjunto de servidores falsos (*fake servers*), e totalmente vulneráveis com o objetivo de atrair, enganar e obter dos *Hackers* todas as atividades não “permitidas”, ou seja, tentativas de intrusão ou outras atividades em seus fakes servers. A ferramenta é ideal para sistemas Linux, devido ao fato de estarem presentes na maioria dos servidores de produção, o fazem ser desejados no mundo virtual.

O *Honeyperl* é uma ferramenta *Open Source*, escrita sob a linguagem *Perl*, que é hoje uma das linguagens que merece destaque se tratando dos quesitos, desenvolvimento web e linguagem multiplataforma, características fortes que aumentam ainda mais sua credibilidade no mundo dos entusiastas por software livre, prova disso é que permite ao administrador desenvolver novos parâmetros para melhoria de sua estrutura e/ou novas funcionalidades em relação a detecção de ataques.

Dentre os serviços que serão avaliados, se tratam basicamente de serviços essenciais em um servidor, pois estão presentes na maioria das instalações reais, ou seja, esse projeto não irá tratar de serviços desconhecidos, ou serviços quase nunca utilizados, pois isso poderia gerar também um insucesso na pesquisa. Os serviços são realmente emulados, todos os serviços reais estão desativos. Todas as portas padrão no sistema descritas pelas suas funcionalidades citadas acima são mantidas pelo *Honeyperl*, mas ele não depende dos serviços reais para seu funcionamento, onde por exemplo no caso do serviço *telnet* dentre outros, não há necessidade de ter servidor *telnetd* instalado no sistema, o *Honeyperl* consegue emular os serviços mesmo que o serviço real estando ausente.

O *Honeyperl* possui uma integração direta com o *firewall* netfilter, que é manipulado pelo comando *Iptables*, podendo ser parametrizado a critério do administrador. Nesse estudo optou-se por não utilizar essa funcionalidade, mantendo o foco principal, que é manter o ambiente o mais vulnerável possível, uma vez ativado o *firewall*, pode-se sofrer influências sobre bloqueios aos ataques que serão direcionados ao *Honeypot* em questão.

Como o *Honeyperl* trabalha com *fakes servers*, seu princípio de funcionamento em relação a captura de informações é algo bem transparente de ser entendido. Assim quando o atacante de partes mais remotas do mundo iniciar qualquer tipo de ataque sobre os serviços emulados, o *Honeyperl* se encarrega da interação com o atacante, gerando todo log através de seus *fake servers*, e melhor, sem que o atacante perceba que esteja realmente conectado ou inteirando com um ambiente que possui a finalidade de representar apenas uma armadilha, assim estando bem distante de um ambiente real. Os logs coletados são compostos basicamente de:

- Serviço ao qual o atacante tentou acesso;
- IP de origem do ataque;
- data e hora do ataque;
- usuário e senha utilizados, quando cabe ao comando essa informação;
- comando executado, esse sendo mostrado quando é reconhecido pela base de assinatura do *Honeyperl*.

De posse a essas informações conseguimos levantar a coleta dos logs gerados pela ferramenta *Honeyperl* em questão, que emula o ambiente *Honeypot* de baixa interatividade, assim consegue-se obter informações importantes como:

- Usuário e senha mais utilizados em determinados serviços;
- horário onde ocorre maior incidência de ataques;
- comandos mais executados;
- serviços mais temidos;
- número de tentativas e serviços explorados, oriundos de um mesmo IP em relação a data, dentre outras.

Toda e qualquer tentativa de conexão gerada nos logs do sistema foi

contabilizada como registro na tentativa de ataque para o ambiente, sobre esses logs, inicia-se a fase de levantamento de informações afim de definir os objetivos propostos nesse trabalho.

As pesquisas por diversas literaturas mostraram que o *Honeyperl* se destaca sobre a ferramenta *Honeyd*, que é sua concorrente direta em sistemas Linux, devido a dificuldade de configuração e a carência de tutoriais explicativos de forma prática para instalação e manutenção dentre todos os dados capturados. A filtragem para obtenção de um ambiente que forneça bons resultados. *Honeyperl* além de ser uma excelente ferramenta como dito anteriormente, conta com a facilidade de configuração e tutoriais funcionais dispostos por toda comunidade Linux, dentre vários portais sobre sistemas Linux e comunidades de segurança, e ainda recebe o suporte por blogs de vários desenvolvedores que participaram ou ainda participam do desenvolvimento dos *fakes servers* para enriquecimento da ferramenta.

10. ESTATÍSTICAS DOS DADOS COLETADOS

O servidor esteve disponível por 44 dias, no período compreendendo as datas de 12 de outubro de 2010 até 25 de novembro de 2010 sem interrupções nos serviços de conexão com internet ou qualquer outro problema que poderia comprometer os resultados esperados nesse trabalho. Nesses 44 dias o arquivo de registro principal do sistema, o qual era responsável pelo armazenamento dos registros, atingiu suas mais precisas 40.416 linhas de registros de tentativas de ataque ao sistema, que prova total sucesso de funcionalidade da ferramenta *Honeyperl* na captura das informações necessárias em *Honeypots* de baixa interatividade.

10.1 FORMATO DOS LOGS *HONEYPERL*

A análise dos resultados sobre a captura dos dados através dos *fakes servers* segue formatos diferenciados para cada tipo de serviço *fake*. Vale ressaltar que os serviços tratados nesse trabalho seguem traduzidos pelo seu sucessor falso: squid, foi emulado pelo servidor falso:*fakesquid.* O servidor smtp foi emulado pelo falso serviço: *fakesmtp.* O servidor httpd foi emulado pelo serviço falso: *fakehttpd.* O servidor pop3 foi emulado pelo falso servidor*: fakepop3.* O serviço echo foi emulado pelo falso*: fakeecho.* O servidor ftp, foi emulado pelo falso servidor: *fakeftp,* e por último o servidor telnet, que foi emulado pelo falso*:faketelnet*.

Os dados levantados sobre os logs serão de caráter genérico, ou seja, não será descrito nos dados estatísticos o que ocorreu no sistema a cada hora de cada dia nesses 44 dias, e sim informações em comum como já mencionado anteriormente nesse trabalho, pois dessa forma a caracterização sobre as informações em cima dos resultados apresentam mais ênfase.

10.1.1 *Fakesquid*

Na tentativa de ataque sobre o *fakesquid*, o arquivo no formato html abaixo será exibido no navegador do atacante:

```
<!DOCTYPE HTML PUBLIC - //IETF//DTD HTML 2.0//EN>
<HTML><HEAD>
<TITLE>ERROR: The requested URL could not be retrieved</TITLE>
</HEAD><BODY>
<H1>ERROR</H1>
<H2>The requested URL could not be retrieved</H2>
<HR>
<P>
While trying to process the request:
<PRE>
get
</PRE>
<P>
The following error was encountered:
<UL>
<LI>
<P>
Some aspect of the HTTP Request is invalid.  Possible problems:
<UL>
<LI>Missing or unknown request method
<LI>Missing URL
<LI>Missing HTTP Identifier (HTTP/1.0)
<LI>Request is too large
<LI>Content-Length missing for POST or PUT requests
<LI>Illegal character in hostname; underscores are not allowed
</UL>
<P>Your cache administrator is <A
$mail.
<hr noshade size=1>
Generated Wed, 28 May 2003 17:22:19 GMT by proxy.$serverurl
($bugsquid)</BODY></HTML>
";
```

10.1.2 *Fakesmtp*

Abaixo segue uma forma de ataque onde o atacante tentou enviar uma mensagem para o e-mail: poi@mail2000.com.tw. Esse formato caracteriza um ataque com tentativa de utilização do serviço.

```
Thu Oct 14 07:54:46 2010  fakesmtp log - Connection from 118.168.142.195:2110
HELO 189.13.11.156 :
MAIL FROM: <hi7188s.pp5975@msa.hinet.net> :
```

RCPT TO: <poi@mail2000.com.tw> :
DATA :
QUIT :

A outra forma de registro que exemplifica outro tipo de ataque, é onde o atacante conectou-se ao servidor mas não interagiu realmente com o *fake*. Ataques desse tipo são característicos de *Hackers* principiantes.

Sat Oct 16 15:44:52 2010 fakesmtp log - Connection from 46.4.211.102:53386

10.1.3 *Fakehttpd*

Segue um formato mais simples de um ataque realizado via web, onde apenas ocorre a tentativa de conexão a um dos domínios criados e explicados no Anexo1 desse trabalho.

Mon Nov 22 23:08:45 2010 fakehttpd log - Connection from 173.13.139.133:54047

Uma outra forma de registro utilizado pelo *fakehttp* que segue abaixo, onde o atacante além de efetuar a conexão utiliza-se do comando GET na tentativa de visualizar algum conteúdo do servidor.

Tue Nov 23 13:59:56 2010 fakehttpd log - Connection from 109.230.213.27:7916
GET http://proxyjudge1.proxyfire.net/fastenv HTTP/1.1 : Ataque WEB ! Tentativa de execucao de comando

10.1.4 *Fakepop3*

Nas tentativas de conexão realizadas ao *fakepop3*, tem-se mais de um padrão identificado pelo sistema, onde em um primeiro estágio o atacante apenas tenta se conectar, assim como nos demais serviços. Abaixo segue o primeiro estágio

de tentativa:

Fri Oct 15 02:33:39 2010 fakepop3 log - Connection from 96.0.243.210:3520

No segundo estágio o atacante realiza a tentativa de conexão, onde tenta perceber o ambiente e digita algum comando, no exemplo abaixo o comando para encerrar a conexão:

Fri Oct 15 02:33:47 2010 fakepop3 log - Connection from 96.0.243.210:3739
QUIT :

No terceiro e último estágio de formato de log gerado pelo *Honeyperl* o atacante vai mais além, na intenção de realmente usufruir dos recursos do sistema.

Fri Oct 15 02:33:45 2010 fakepop3 log - Connection from 96.0.243.210:3740
USER tester :
PASS temp1234 :
QUIT :

10.1.5 *Fakeecho*

Utilizando-se do *fakeecho* o atacante pode tentar redirecionar algum texto, ou algum outro tipo de comunicação, tanto para um terminal em execução quanto para um arquivo de configuração de extrema importância para o sistema. Segue um exemplo ilustrativo abaixo:

Fri Nov 12 21:38:02 2010 fakeecho log - Connection from 123.121.206.185:62496
^E^A :

Um outro estágio de tentativa de ataque é apresentado abaixo do *fakeecho*:

Fri Nov 12 21:38:01 2010 fakeecho log - Connection from 123.121.206.185:62414
CONNECT us.battle.net:443 HTTP/1.1 :
Accept: */* :
Content-Type: text/html :
Proxy-Connection: Keep-Alive :
Content-length: 0 :

10.1.6 *Fakeftp*

As tentativas de acesso a esse *fake* foram constantes, a sua maioria caracteriza por tentativas de ataques força bruta, onde o atacante de posse a uma lista contendo usuário e senhas padrões de muitas instalações nos sistemas são testadas incansavelmente.

Pelo fato de emular um serviço largamente utilizado no mundo real, o *ftp*, o *fakeftp* recebeu centenas de ataques, por se tratar de um protocolo de transferência de arquivos. Segue dois exemplos distintos de tipos de ataques:

```
Thu Oct 14 19:22:43 2010  fakeftp log - Connection from 189.76.222.6:43389
USER anonymous :
PASS mozilla@example.com :
SYST :
PWD :
TYPE I :
PASV :
Aviso Fake-FTP : Fake FTP abriu uma conexao no modo passivo na porta 32309
SIZE / :
MDTM / :
RETR / :
PASV :
Aviso Fake-FTP : Fake FTP abriu uma conexao no modo passivo na porta 58756
CWD / :
```

10.1.7 *Faketelnet*

O *faketelnet* não compõe por padrão os *fakes* da versão do *Honeyperl* utilizada, ele foi incorporado ao ambiente, assim seu formato de log é apenas no terminal em execução, como segue abaixo:

```
Trying 183.83.19.245...
Connected to srv-netone.
Escape character is '^]'.
Debian GNU/Linux 5.0
srv-netone login: firtewin
Password: linuxsrv

Login incorrect
srv-netone login: root
Password: saganet

Login incorrect
```

Como notado acima o terminal fica paralisado na espera de uma conexão

com o *fake*. O atacante tentou efetuar uma conexão com o servidor, porém não obteve sucesso, pois toda e qualquer informação que o mesmo repassar ao sistema não será aceita.

11. RESULTADOS

11.1 RESUMO QUANTITATIVO DE ATAQUES

No decorrer dos dias que o ambiente ficou sob monitoramento, ocorreu um grande número de ataques, como já era esperado. Desses ataques, vários ocorreram apenas uma vez relacionado a um mesmo IP e país de origem, como da mesma forma ocorreu diversas situações em que houve reincidência.

É muito interessante como apresenta a tabela 1 informações importantes sobre os ataques como: IP, países de origem dos ataques, número de ataques, data e hora do ataques. O Chile é apresentado na tabela 1, linha 19, com apenas uma incidência nesse período e se mantém como o país de maior ataque, resultado é que em apenas uma incidência o índice de tentativas de conexão em vários serviços foi enorme, o que fez seu índice quantitativo aumentar de forma significante.

Pode se observar claramente pela tabela 1, organizada pela data do incidente, que o horário de maior incidência é no período da noite, onde geralmente os estudantes, e demais entusiastas ao mundo *Hacker* estão de retorno aos lares.

DATA/HORA	IP	PAÍS DE ORIGEM	NÚMERO DE ATAQUES
11/10/2010 16:00 às 17:00	**114.45.50.191**	República da Coréia	1
12/10/2010 a 13/10/2010 11:00 as 17:00	**189.76.222.254**	Brasil	26
13/10/2010 21:00 às 22:00	**189.127.159.154**	Brasil	14
13/10/2010 23:00:00	**63.230.183.150**	Estados Unidos	1
14/10/2010 07:00 às 08:00	**118.168.142.195**	Taiwan	1
15/10/2010 2:00 às 3:00	**96.0.243.210**	Estados Unidos	859
15/10/2010 a 20/10/2010 11:00 às 02:00	**112.105.142.231**	Taiwan	2
15/10/2010 00:00 às 01:00	**187.63.198.185**	Brasil	5
16/10/2010 a 17/10/2010 07:00 às 19:00	**75.127.194.3**	Estados Unidos	6
16/10/2010 15:00 às 16:00	**46.4.211.102**	Germânia	2
20/10/2010 16:00:00	**124.160.94.218**	China	1
23/10/2010 16:00 às 17:00	**77.221.156.210**	Rússia	10
23/10/2010 22:00 às 23:00	**88.80.10.1**	Países Baixos	1
23/10/2010 21:00 às 22:00	**218.87.16.140**	China	2
31/10/2010 20:00 as 21:00	**208.109.91.114**	Países Baixos	1
31/10/2010 21:00 às 22:00	**206.71.179.51**	Azerbaijão	2
31/10/2010 a 01/11/2010 22:00 às 02:00	**212.154.153.85**	Cazaquistão	2
01/11/2010 17:00 as 18:00	**189.83.29.56**	Brasil	5
25/11/2010 05:00 às 08:00	**200.111.183.252**	Chile	4773
01/11/2010 19:00 às 18:00	**118.175.66.120**	Tailândia	157
01/11/2010 20:00 às 21:00	**62.233.197.190**	Estados Unidos	1
04/11/2010 03:00 às 04:00	**118.166.212.178**	Taiwan	1
04/11/2010 09:00 às 11:00	**118.167.97.111**	Taiwan	5
06/11/10 00:00 às 01:00	**114.45.55.182**	Taiwan	1
07/11/2010 16:00 às 17:00	**94.23.159.248**	França	2
10/11/2010 a 12/11/2010 08:00 às 23:00	**216.240.147.78**	Estados Unidos	12
10/11/2010 21:00 às 22:00	**208.109.217.92**	Azerbaijão	2
11/11/2010 15:00 às 16:00	**93.186.192.105**	Germânia	2
12/11/2010 06:00 às 07:00	**207.218.218.58**	Estados Unidos	2
12/11/2010 a 12/11/2010 21:00 às 17:00	**123.121.206.185**	China	4
13/11/2010 13:00 às 14:00	**195.243.243.4**	Germânia	1
15/11/2010 00:00 às 01:00	**128.59.14.115**	Estados Unidos	2
16/11/2010 13:00 às 14:00	**85.214.49.14**	Germânia	2
17/11/2010 17:00 as 18:00	**82.113.149.196**	Reino Unido	4
18/11/2010 17:00:00	**189.13.139.133**	Brasil	8
21/11/2010 23:00 às 00:00	**201.50.148.134**	Brasil	1

23/11/2010 12:00 às 13:00	**113.111.52.227**	China	1
24/11/2010 09:00 às 18:00	**62.215.242.203**	Kwait	179
25/11/2010 04:00 às 05:00	**184.154.62.41**	Estados Unidos	1
25/11/2010 10:00 às 11:00	**72.55.165.45**	Canadá	2
25/11/2010 18:00 às 08:00	**65.207.113.162**	Estados Unidos	135
25/11/2010 11:00 às 12:00	**118.160.213.112**	Taiwan	1

Tabela 1: Resumo quantitativo de ataques baseados no país de origem, data e hora.

Como demonstrado pela tabela1, diversos ataques oriundos de diversos países contribuíram para os resultados desse trabalho. Pode-se levantar a partir dos dados da tabela acima que o horário de maior incidência é no período noturno, estendendo-se pela madrugada, pois geralmente é o momento que os usuários/Hackers estão de fato em suas residências e aproveitam para atravessar as madrugadas a troca de um ataque de sucesso, onde buscam realmente como resultado um nível gratificante de comprometimento com o sistema. Mas como o presente trabalho trata-se de um ambiente *Honeypot,* esse nível de satisfação desejado pelo *Hacker* nunca é atingido.

11.2 NÍVEIS DE INTERAÇÃO E FAKES COMPROMETIDOS

A tabela 2 esclarece todos os IPS relacionados na tabela 1, com o diferencial de apresentar os *fakes* comprometidos, usuários e senhas mais explorados pelos atacantes. Uma versão mais completa da tabela, segue no anexo II, com o complemento dos comandos executados.

A tabela 2 demonstra os fakes que tiveram maior comprometimento, seguido de usuários e senhas mais utilizados. Os *Hackers,* utilizam uma base de assinatura ou uma listagem pré-definida contendo diversos usuários e senhas. Esse usuários são mesclados entre as senhas para que sejam formadas combinações diferentes no intuito de aumentar as chances de um ataque de sucesso.

Em muitos dos ataques não era registrado usuário e senha, devido ao fato de se tratar de apenas conexões e não tentativa de acesso através de usuários, nesse caso apenas a tentativa em questão era logada.

IP	FAKE	USUÁRIOS	SENHAS
96.0.243.210	POP3	stanley,roob,milton,minim,tester,neal,solar,root,todd,stell,neil,pics,portal,timothy,victor,wear,rosimare,tomy,sasha,sanchez	test1234,passwd,qweasd.
118.175.66.120	POP3	root,admin,webmaster,user,web,www,administrator,oracle,sybase,informix,oracle8,lizdy,data,account,access,pwrchute,	1234
189.76.222.254	POP3, SMTP, HTTPD, FTP	root, linux	root, saganet
200.111.183.252	POP3	ortega,igor,sarah,travis,romeo,steve,superman,public,reagan,test2,peterson,pub,website,susan,sonny,screen,training,siteadmin	eagle,qwerty,super,123,network,guest,123456,network,abc123
189.127.159.154	HTTPD		
63.230.183.150	HTTPD		
75.127.194.3	HTTPD		
124.160.94.218	HTTPD		
77.221.156.210	HTTPD		
88.80.10.1	HTTPD		
188.165.64.240	HTTPD		
208.109.91.114	HTTPD		
62.233.197.190	HTTPD		
94.23.159.248	HTTPD		
216.240.147.78	HTTPD		
195.243.243.4	HTTPD		
128.59.14.115	HTTP		
82113149196	HTTPD		
189.13.139.133	HTTPD		
184.154.62.41	HTTPD		
187.63.198.185	FTP,POP3,TELNET	root,	saganettttt
218.87.16.140	FTP		
206.71.179.51	FTP		
189.83.29.56	FTP, TELNET		
208.109.217.92	FTP		
207.218.218.58	FTP		
85.214.49.14	FTP		
72.55.165.45	FTP		
118.168.142.195	SMTP		

112.105.142.231	SMTP		
46.4.211.102	SMTP,POP3		
118.166.212.178	SMTP		
118.167.97.111	SMTP		
114.45.55.182	SMTP		
113.111.52.227	SMTP		
201.50.148.134	SMTP		
114.45.50.191	SMTP		
65.207.113.162	SMTP		
118.160.213.112	SMTP		
123.121.206.185	ECHO		

Tabela 2: Comprometimento, serviços mais temidos, usuários e senhas mais comuns utilizados pelos Hackers

Interessante também ressaltar que um único IP de origem efetuava tentativa de conexão a vários serviços, isso prova que o ambiente estava respondendo bem aos estímulos, pois conseguia de forma simples prender a atenção do *Hacker*, por um período um pouco mais extenso, aumentando a base de informações para análise.

11.3 RESULTADOS ESTATÍSTICOS

Segue a Ilustração 13 mostrando o ranking dos 10 países onde mais se originaram ataques com base na coleta de informações sob *Honeypots* de baixa interatividade utilizando a ferramenta *Honeyperl*.

O Chile se destacou com grande diferença dos demais países. Analisando as tabelas de logs, pode-se ver que tal diferença se explica devido aos ataques serem do tipo força bruta, onde uma rajada de informações era disparada sob o *Honeypot*, fazendo com que seus registros aumentassem de forma rápida, fazendo-o ocupar o primeiro lugar dentro os demais, com características de um ataque automatizado, revelando ações de um *script kiddie*.

Canadá, Países Baixos e Cazaquistão mantiveram empatados com 2 registros cada, os demais seguem cada um com seus valores.

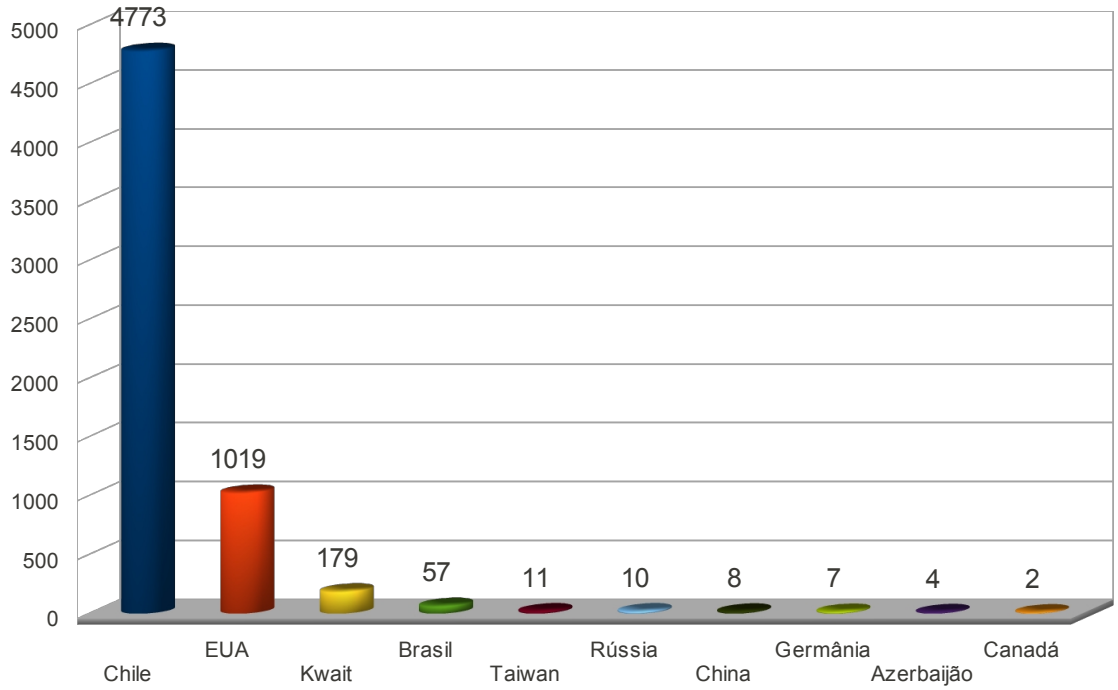

Ilustração 13: Ranking dos 10 países onde mais se originam ataques

A ilustração 13 demonstra acima de tudo nesse trabalho, que mesmo com todos os problemas referentes a segurança em relação ao Brasil, diversos países contribuem para o aumento do número de Hackers além do Brasil, fato que comprova o Brasil estar no 4º lugar e não em 1º lugar no trabalho. Isso muito preocupa, pois do lado da segurança de redes as ferramentas e sistemas tem que ser melhorados em uma proporção bem superior as ameaças para que um dia esses ataques sejam de fato reduzidos a uma parcela bem menor.

11.4 *FAKESERVER* COM MAIS OCORRÊNCIAS DE ATAQUES

A Ilustração 14 demonstra os serviços mais procurados pelos *Hackers* da atualidade. Como já mencionado nesse trabalho, o *fakehttpd* sobressaiu sob os outros devido ao fato de estar emulando um servidor Web, pois na atualidade são

mais cobiçados VALLE e ULBRICH(2007). A ilustração 14 comprova a afirmação:

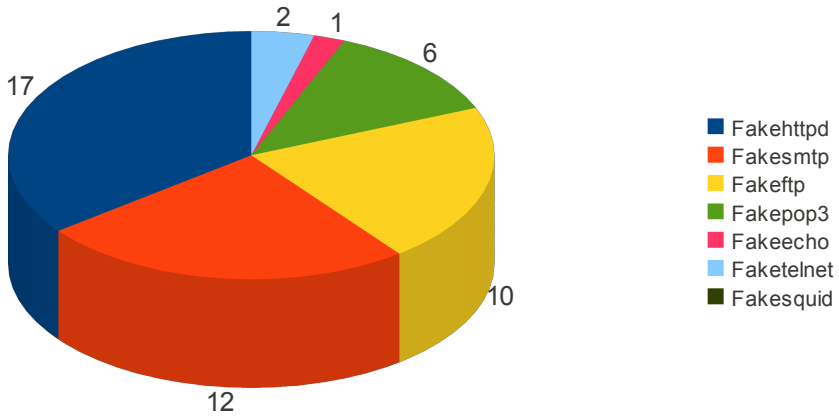

Ilustração 14: Serviços mais desejados pelos atacantes

11.5 PAÍSES MAIS REINCIDENTES

A seguir na ilustração 15, é definido os países principais em reincidência a ataques, ou seja, aqueles países que realizaram tentativas de acesso ao servidor diversas vezes. Esse é um tipo de informação muito importante de ser levantado, pois como ocorreram diversos ataques e muitos deles uma única vez, é geralmente característico de ataque automatizado. Sendo assim quanto maior o grau de reincidência de ataques é prova que o atacante está monitorando o servidor em questão.

Como mencionado no decorrer do trabalho, o Chile se destacou no número de ataques, porém no quesito reincidência ele não aparece, pois trata-se de um caso único. Como podemos perceber, Estados Unidos lidera o primeiro lugar, com maior índice de reincidência.

Nesse trabalho não está sendo levado em consideração a identificação da ferramenta utilizada para a realização do ataque, por não ser realmente o foco de estudos utilizando *Honeypots* de baixa interatividade.

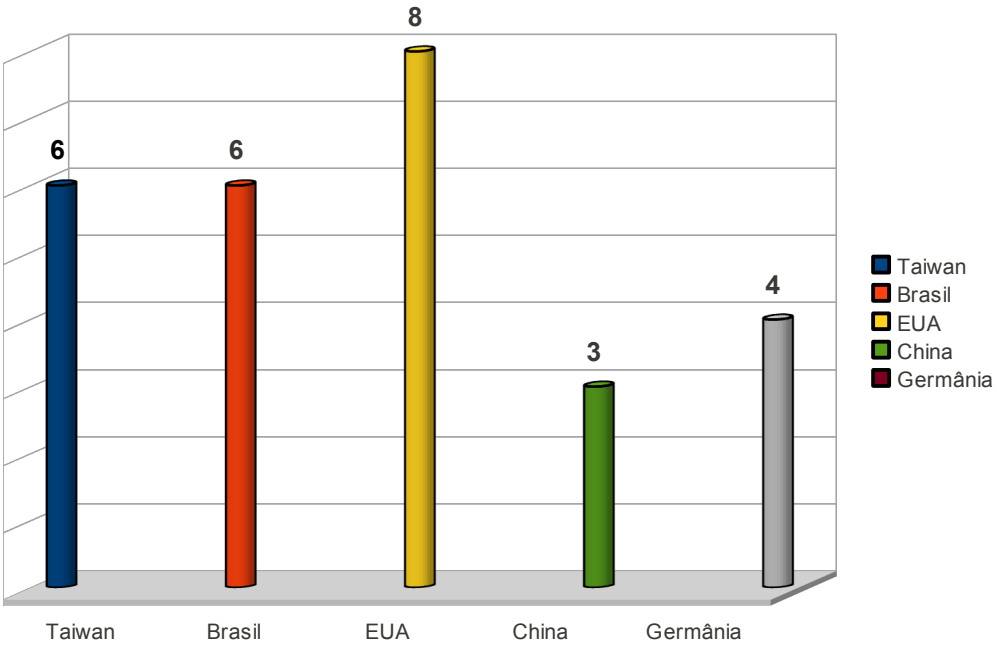

Ilustração 15: Cinco países de maiores reincidência em ataques

11.6 IMPORTÂNCIA DOS HONEYPOTS PARA SEGURANÇA NAS EMPRESAS

Os dados levantados por meio dos gráficos e tabelas citadas acima demonstram claramente que as empresas devem investir o máximo possível em segurança, e ainda quando possível implantarem um *Honeypot*, para conseguir extrair as informações levantadas nesse trabalho que ajudará a zelar pela segurança da organização em questão.

A segurança como tratada no decorrer desse trabalho, é assunto que merece respaldo, e ainda ser tratado com seriedade. Hoje muitas vítimas de ataques poderiam ter se prevenido a algum tempo atrás com a instalação de *Honeypots* por exemplo, implementando soluções mais robustas no ambiente real de produção sobre os resultados obtidos com *Honeypots*. Como ressaltado no trabalho, o *Honeypot* não é um *firewall*, mais consegue alertar ao administrador de uma rede por exemplo, sobre os pontos mais vulneráveis existentes em sua organização.

As empresas hoje, são o foco principal dos ataques, prova disso é ter utilizado

DNS`s que facilitaram as pesquisas por parte dos *Hackers*, e que possuem similaridade com as nomenclaturas de domínios reais, representando organizações de suma importância a humanidade, como por exemplo: google.com e googlenetwork.sytes.net.

Se o trabalho fosse realizado como uma tarefa do dia a dia em um ambiente real de produção, ao invés de ser um estudo científico, o administrador de redes de posse aos resultados estaria esclarecido sobre os pontos falhos em relação a segurança em seus servidores, e que alguns serviços quando não utilizados podem ser removidos para evitar transtornos e comprometimento por pessoas não autorizadas a estrutura do sistema.

Honeypots traduzem de forma muito clara sua importância no crescimento e amadurecimento de uma organização e justifica seus valores, e devem ser utilizados e explorados cada vez mais, para que assim consiga-se informações relevantes para tomada de decisão, nas questões que envolvem segurança, se tratando de tecnologia da informação em um ambiente empresarial.

12. CONCLUSÃO

Com o desenvolvimento desse trabalho, notou-se a crescente demanda sobre a necessidade de monitoramento de redes e de sistemas envolvidos. Demonstrou-se que as organizações devem ser monitoradas com ferramentas que realmente tragam resultados precisos e que os administradores de rede sempre devem atualizar seu conhecimento sobre correções de falhas e ameaças que rodeiam seus servidores.

O *Honeypot* não substitui nenhuma técnica de segurança desenvolvida ou aplicada por administradores no dia a dia, ele somente agrega valores de importância, trazendo ricos resultados para melhorias sobre as ferramentas e metodologias já utilizadas. Isso permite um contato mais próximo com os *Hackers*, deixando-os mais próximos aos serviços oferecidos, fornecendo aos administradores uma melhor compreensão das ações dos invasores, de suas intenções e dos danos projetados às redes de computadores e aos sistemas envolvidos.

A captura de informações como: comandos, países de origem com seus respectivos IPS, usuários e senhas, serviços mais temidos, data e hora da ocorrência de ataques, reafirmam o sucesso do propósito desse trabalho. Ainda transparece a importância da utilização do *HoneyPerl* como *Honeypot* de baixa interação, pelo fornecimento de seus logs com formatos mais esclarecedores, ao contrário de outras ferramentas, como o *Honeyd*.

Assim, entende-se que os valores agregados a organização através de um *Honeypot*, estão diretamente ligados ao profissional envolvido e com a ferramenta em questão, pois seu sucesso demanda um acompanhamento diário dos registros para que as decisões possam ser mais rápidas, trazendo estabilidade a segurança da rede da organização envolvida.

12.1 TRABALHOS FUTUROS

Dentre várias propostas de trabalhos a futuros concluentes do curso de Ciência da Computação dessa, e de outras instituições, envolvendo o assunto sobre segurança em redes de computadores através de *Honeypots,* destaca-se algumas das mais importantes tirando proveito do presente trabalho realizado.

Segue como primeira proposta a exposição do servidor *Honeypot* durante um período bem superior a 44 dias*,* o qual foi exposto*,* para que mais informações sejam levantadas, podendo se ter uma base de informações mais completa capaz de contribuir para levantamento de informações estatísticas com maior exatidão.

A segunda proposta, trata-se da implantação de um *Honeypot* na Infraestrutura de TI dessa instituição, seja na forma de grupo de estudos, ou um projeto real de conclusão. Consequentemente esse projeto de *Honeypot* inicial pode se estender ao projeto de uma *Honeynet*, destacando pelo grau de abrangência e complexidade envolvida, ideal para se tratar em um grupo de pesquisa.

Na terceira proposta, que é muito abrangente, seria um grupo de alunos, interessados em estudar *Honeypots* orientados pelo coordenador do curso, em nome da instituição FIC (Faculdades Integradas de Caratinga) representar um CSIRT, ou seja representando mais um sensor agregado ao CERT.br, para reportar os incidentes ocorridos. Para que isso seja possível a instituição teria que comprovar por meios burocráticos o seu interesse em realmente de integração ao CERT.br. Todos esses procedimentos são descritos no portal do CERT.br.

Assim a instituição FIC abre automaticamente uma nova linha de pesquisa e desafios aos egressos e concluintes do curso de Ciência da Computação, podendo se expandir a outros campi da instituição, assim a instituição alcança um grande status, pois passará a fazer parte de um componente do CERT.br que é um órgão sério e muito respeitado mundialmente.

Vale ressaltar, que no nosso estado só existem dois pontos de presença através de CSIRT, segundo dados atualizados do CERT.br, que estão localizados nas cidades de Uberlândia, e Belo Horizonte, isso é prova de que a instituição representar um CSIRT é uma proposta de trabalho totalmente viável.

REFERÊNCIAS

ANDRUCIOLI, Alexandre Pinaffi . **Proposta e Avaliação de um Modelo Alternativo Baseado em Honeynets para Identificação de Ataques e Classificação de Atacantes na Internet**. COPPE/UFRJ [Rio de Janeiro] 2005.

ASSUNÇÃO, Flávio Marcos Araújo. **Honeypots e Honeynets: Aprenda a detectar e enganar os invasores**. Florianópolis: Visual Books, 2009.

BELCHIOR, Francisco de Oliveira; SOUSA, Iara Moura; ARAÚJO, Lêda Brito. **Projeto: Utilizando Honeynets como ferramenta auxiliar na verificação de vulnerabilidades em sistemas operacionais.** Faculdade AD1, Brasília, novembro 2004.

BORGES, Ciro Fernando Preto; BENTO, Paulo Diego Nogueira. **Segurança de redes utilizando Honeypots**. Instituto de Estudos Superiores da Amazônia - IESAM [Belém - PA] 2006.

CERT.br, **Centro de Estudos, Respostas e Tratamento de Incidentes de Segurança no Brasil**. Disponível em http://www.cert.br, acessado em 07 de setembro de 2010.

EPIDEMICLINUX, **Uma experiência contagiante**, disponível em http://epidemiclinux.org, acessado em 19 de novembro de 2010.

HATCH, Brian; LEE, James; KURTZ, George. **Hackers Expostos – Linux**. São Paulo: Makron Books 2002.

KUROSE, James F. ; ROSS, Keith W. **REDES DE COMPUTADORES E A INTERNET**, Uma abordagem Top-Down. São Paulo: Pearson .

MARCELO, Antônio; ALVES, Marcos José Pitanga. ***Honeypots*: A arte de iludir hackers**. Rio de Janeiro: Brasport, 2003.

MORIMOTO, Carlos Eduardo. **Servidores Linux, Guia prático**. Porto Alegre: Sul Editores, 2006.

NETO, Urubatan. **Dominando Linux Firewalls Iptables**. 1ªed. Ciência Moderna Ltda, Rio de Janeiro – Brasil, 2004.

SOUZA, Tiago. **Honeypots - A segurança através do disfarce**. Ravel – COPPE/UFRJ, GRIS – DCC/UFRJ 9 de Agosto de 2005

SPITZNER, Lance. **Honeypots: Tracking-Hackers.** 1ªed Addison-Wesley Professional 2002

STREBE, Matthew; PERKINS, Charles. **Firewalls**. São Paulo: MAKRON Books, 2002.

TANENBAUM, Andrew S. **Redes de Computadores**, 3ª Edição. Rio de Janeiro: Campus, 1997.

SPITZNER, Lance. THE *Honeynet* PROJECT. "**Conheça seu inimigo – O Projeto Honeynet**". Editora Makron Books, São Paulo, 2002.

THE *Honeynet* PROJECT. Disponível em http://www.project.*Honeynet*.org, acessado em 17 de outubro de 2010.

THE *Honeynet* PROJECT. Disponível em Know http://www.project.*Honeynet*.org/book/, acessado em 17 de outubro de 2010.

ULBRICH, Henrique César; VALLE, James Della. **Universidade H4CK3R**. 5ª Ed. São Paulo: Digerati Books 2007.

WALKERTALKS, disponível em http://walkertalks.com, acessado em 25 de outubro de 2010.

ANEXO I - INSTALAÇÃO HONEYPERL

Para se instalar a ferramenta, basicamente aconselha-se uma distribuição estável para que seja garantido que as falhas ocorreram apenas pela ferramenta Honeyperl, em consequência no projeto optou-se pela utilização da distribuição Debian, por ser uma distribuição já consagrada no mundo de servidores Linux.

A instalação e configuração da ferramenta pode ser realizada a partir de pequenos passos, segundo BORGES & BENTO(2006), a única dependência é que o *Perl* instalado no servidor esteja na versão 5.60 ou superior. A distribuição Debian pode ser obtida através do portal oficial da Debian (www.debian.org). O *Honeyperl* pode ser obtido pelo portal oficial (www.*Honeypot*.com.br/honeyperl) ou (http://sourceforge.net/projects/honeyperl/).

CONFIGURAÇÃO E EXECUÇÃO

Segue o conjunto de passos para instalação e configuração da *Honeyperl*:

1. Efetue o download no site oficial http://www.*Honeypot*.com.br/honeyperl ou no site alternativo http://sourceforge.net/projects/honeyperl/

2. Descompacte o pacote com o comando tar -xzvf honeyperl-005.tar.gz.

3. Acesse o diretório criado honeyperl-005, em sequência execute o comando perl verify.pl, para verificar se todos os módulos *Perl* necessários estão presentes. A não existência de algum dos módulos impossibilita o funcionamento da ferramenta, nesse caso deve-se prosseguir com o download e instalação dos módulos através do painel CPAN, através do comando com usuário root :
    1. perl -MCPAN -e shell

2. install *nomedomodulo*
3. exit.

4. Aconselha-se que tenha instalado no sistema o aplicativo NMAP, não é de forma alguma uma dependência para *Honeyperl* funcionar, mais facilita a administração do ambiente para visualização das portas que foram abertas em tempo real pela ferramenta. Com a ferramenta a possibilidade na falha de inicialização de algum *fake* é logo detectada, sem muitos artifícios por parte do administrador no sistema.

5. Ainda no diretório utilize o editor de preferência para edição do arquivo de configuração principal *honeyperl.conf*. Nesse arquivo concentra-se todos os parâmetros que devem ser alterados para o ambiente em questão. Teremos o arquivo como descrito abaixo:

#Dominio que será utilizado pelos fakes
dominio=Honeypot.com.br
#Email utilizado nos fakes
email=admin@googlenetwork.sytes.net

#Usuario utilizado (nao rode como root)
usuario=root

#Arquivo de log
logfile=logs/honeyperl.log

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

#Deseja ativar firewall
#opcoes:(sim/yes)/(nao/no)

firewall=sim

#Os sistemas disponíveis para utilização de firewall:
#pode-se ter linux22, linu24 ou openbsd
#openbsd : trabalha com PF
#linux24 : IPTables
#linux22 : ipchains
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
faketelnet:telnet:conf/telnet.conf:23:telnet emul

A configuração dos fakes pode ser feita modificando seus arquivos 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 emulado. As opções validas são: exchange, sendmail, qmail e postfix. Já $logdir=”logs/smtp”; indica diretoriode log 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 *faketelnet*, foi incorporado a estrutura do sistema, não está incluso por padrão na instalação do *Honeyperl*, isso prova a flexibilidade da ferramenta, onde diversos *fakes* podem ser inseridos e/ou desenvolvidos para fazerem parte do sistema. Os *fakes, squid* e *ftp*, trabalham com uma base de assinaturas para reconhecimento dos ataques, estão localizados no mesmo diretório, sendo eles: web-regras.txt e ftp-signatures.txt.

web-regras.txt:

/bin/ps=Ataque WEB ! Tentativa de execucao de comando
get=Ataque WEB ! Tentativa de execucao de comando
http /=Ataque WEB ! Tentativa de execucao de comando
http 1.1=Ataque WEB ! Tentativa de execucao de comando
head=Ataque WEB ! Tentativa de execucao de comando mal formado
/phf=Ataque WEB ! Tentativa de exploracao de bug phf
phf=Ataque WEB ! Tentativa de exploracao de bug phf

ps=Ataque WEB ! Tentativa de execucao de comando
wget=Ataque WEB ! Tentativa de execucao de comando
/usr/bin/id=Ataque WEB ! Tentativa de execucao de comando
/bin/echo=Ataque WEB ! Tentativa de execucao de comando
/bin/kill=Ataque WEB ! Tentativa de execucao de comando
/bin/chmod=Ataque WEB ! Tentativa de execucao de comando
/chgrp=Ataque WEB ! Tentativa de execucao de comando
/chown=Ataque WEB ! Tentativa de execucao de comando
/usr/bin/chsh=Ataque WEB ! Tentativa de execucao de comando
tftp =Ataque WEB ! Tentativa de execucao de comando
/usr/bin/gcc=Ataque WEB ! Tentativa de execucao de comando
gcc -o=Ataque WEB ! Tentativa de execucao de comando
/usr/bin/cc=Ataque WEB ! Tentativa de execucao de comando
cc =Ataque WEB ! Tentativa de execucao de comando
/usr/bin/cpp=Ataque WEB ! Tentativa de execucao de comando
cpp =Ataque WEB ! Tentativa de execucao de comando
bin/python=Ataque WEB ! Tentativa de execucao de comando
python =Ataque WEB ! Tentativa de execucao de comando
bin/tclsh=Ataque WEB ! Tentativa de execucao de comando
tclsh8 =Ataque WEB ! Tentativa de execucao de comando
bin/nasm=Ataque WEB ! Tentativa de execucao de comando
nasm =Ataque WEB ! Tentativa de execucao de comando
/usr/bin/perl=Ataque WEB ! Tentativa de execucao de comando
perl =Ataque WEB ! Tentativa de execucao de comando
net localgroup administrators /add=Ataque WEB ! Tentativa de execucao de comando
traceroute =Ataque WEB ! Tentativa de execucao de comando
/bin/ping=Ataque WEB ! Tentativa de execucao de comando
nc =Ataque WEB ! Tentativa de execucao de comando
nmap =Ataque WEB ! Tentativa de execucao de comando
/usr/X11R6/bin/xterm=Ataque WEB ! Tentativa de execucao de comando
-display =Ataque WEB ! Tentativa de execucao de comando

lsof =Ataque WEB ! Tentativa de execucao de comando
rm =Ataque WEB ! Tentativa de execucao de comando
/bin/mail=Ataque WEB ! Tentativa de execucao de comando
mail =Ataque WEB ! Tentativa de execucao de comando
/bin/ls\|=Ataque WEB ! Tentativa de execucao de comando
/bin/ls=Ataque WEB ! Tentativa de execucao de comando
/etc/inetd.conf=Ataque WEB ! Tentativa de execucao de comando
/etc/motd=Ataque WEB ! Tentativa de execucao de comando
/etc/shadow=Ataque WEB ! Tentativa de execucao de comando
conf/httpd.conf=Ataque WEB ! Tentativa de execucao de comando
.htgroup=Ataque WEB ! Tentativa de execucao de comando

ftp-signatures.txt:

CEL=FTP CEL overflow attempt
CMD=FTP CMD overflow attempt
STAT=FTP STAT overflow attempt
SITE CHOWN=FTP SITE CHOWN overflow attempt
SITE NEWER=FTP SITE NEWER overflow attempt
SITE CPWD=FTP SITE CPWD overflow attempt
SITE EXEC=FTP SITE EXEC format string attempt
SITE=FTP SITE overflow attempt
RMDIR=FTP RMDIR overflow attempt
MKD=FTP MKD overflow attempt
REST=FTP REST overflow attempt
DELE=FTP DELE overflow attempt
RMD=FTP RMD overflow attempt
MODE=FTP invalid MODE
SITE ZIPCHK=FTP SITE ZIPCHK attempt
SITE NEWER=FTP SITE NEWER attempt
SITE EXEC=FTP site exec
CWD ~root=FTP CWD ~root attempt

CWD ...=FTP CWD ...
CWD ....=FTP CWD .... attempt
.%20.=FTP serv-u directory transversal
%p=FTP format string attempt
RNFR ././=FTP RNFR ././ attempt
LIST ..=FTP LIST directory traversal attempt
.forward=FTP .forward
.rhosts=FTP .rhosts
authorized_keys=FTP authorized_keys
RETR passwd=FTP passwd retreval attempt
RETR shadow=FTP shadow retreval attempt
pass -iss@iss=FTP iss scan
pass wh00t=FTP pass wh00t
pass -cklaus=FTP piss scan
pass -saint=FTP saint scan
pass -satan=FTP satan scan

Toda e qualquer tentativa de execução de comandos que encaixe nessa base de assinaturas serão logados da forma em que estão descritos. Essa base de assinatura pode ser personalizada a critério do administrador, de forma a poder ficar mais explícito nos logs um ataque web do tipo: nmap =Ataque WEB ! Tentativa de execução de comando, por nmap = Scan de porta realizado pelo nmap ! Execução de comando sniffer.

Para facilitar a vida do atacante ainda mais, foram criados diversos DNS`s dinâmicos contando com a ajuda da ferramenta NO-IP. Através dela o ambiente responde por cinco domínios, o que facilita muito um scan oriundo de um *Hacker* ir de encontro ao *Honeypot*.

O SERVIÇO NO-IP

NO-IP é um serviço de DNS dinâmico, ou seja ao invés de lembramos o IP de um determinado domínio é muito mais simples lembramos o domínio yourname.no-ip.org por exemplo. É um serviço largamente utilizado nas plataformas nas diversas plataformas de sistemas atuais, devido ao fato de renovar um endereço automaticamente quando trabalha com conexões dinâmicas a Internet. Para que isso seja possível pode-se utilizar uma gama de subdomínios livres, entre eles o trabalho contou com: *sytes.net, no-ip.org, no-ip.info, no-ip.biz, servehttp.com* Fonte: http://www.no-ip.com.

Trata-se de um serviço gratuito, quando está sendo utilizado para fins pessoais e pago para fins comerciais, onde exige uma grande demanda de tráfego. O que o NO-IP faz é simplesmente direcionar todos os domínios cadastrados na conta do cliente em seu portal Web para um determinado IP. Dessa forma consegue-se cinco domínios diferentes tratando-se de uma conta gratuita, que é o máximo permitido. Assim o servidor utilizado no trabalho consegue responder pelos seguintes domínios:

- **bancodobrasil.no-ip.org;**
- **googlenetwork.sytes.net;**
- **jupiternet.servehttp.com;**
- **emachines.no-ip.info;**
- **sciencecomputer.no-ip.biz.**

Nesse ponto já tem-se um ambiente pronto e funcional, com os principais parâmetros para um melhor funcionamento. Aconselha-se retirar todos os serviços reais da inicialização do sistema, os quais serão emulados pela ferramenta. Caso contrário por funcionar na mesma porta dos serviços reais os *fakes* não conseguirão executar e retornará uma mensagem semelhante a : “Endereço já em uso at telnet.pl line 17”, no caso do serviço telnet.

Para certificar-se de que os serviços inicializaram, pode-se inicializar o Honeyperl com o seguinte comando e acompanhar as mensagens mostradas no terminal de acordo com a ilustração 16.

 # perl honeyperl.pl

Abaixo segue a tela inicial do sistema com os serviços inicializados.

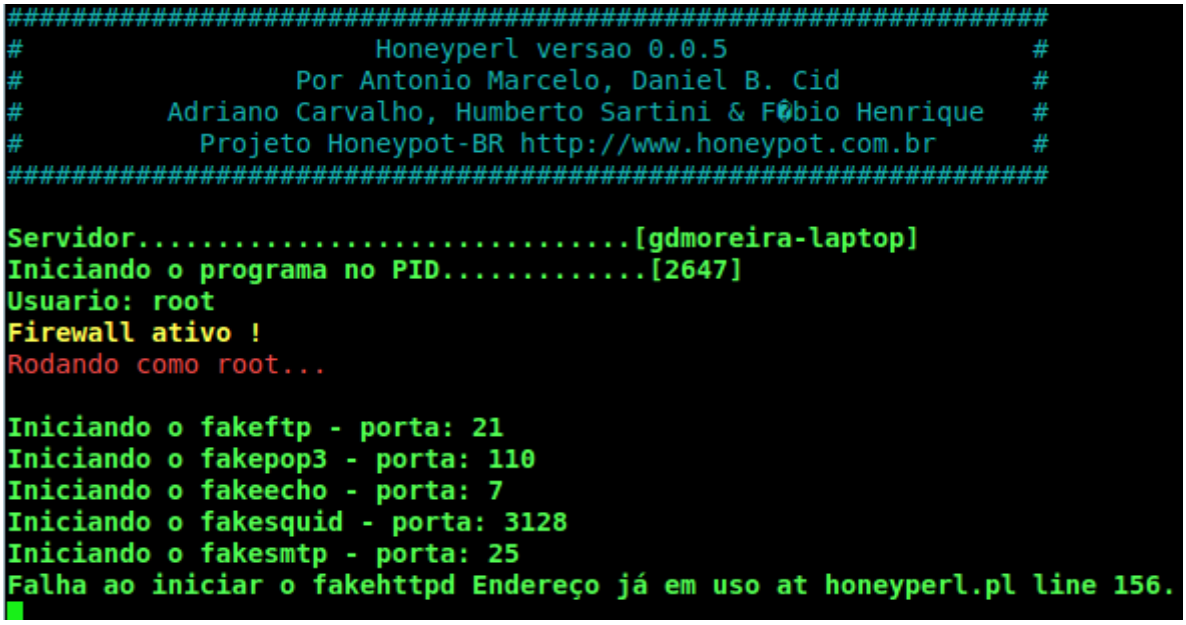

Ilustração 16: Tela inicial do sistema com serviço real em execução, em decorrência falha ao inicializar fake

ANEXO II – ANÁLISE GERAL DE COMPROMETIMENTO DE SERVIÇOS E INTERAÇÕES MAIS COMUNS

IP	FAKE	COMANDO	USUÁRIOS	SENHAS
96.0.243.210	POP3	Conexão efetuada	stanley,roob,milton,minim,tester,neal,solar,root,todd,stell,neil,pics,portal,timothy,victor,wear,rosimare,tomy,sasha,sanchez	test1234,passwd,qweasd.
118.175.66.120	POP3	Conexão efetuada	root,admin,webmaster,user,web,www,administrator,oracle,sybase,informix,oracle8,lizdy,data,account,access,pwrchute,	1234
189.76.222.254	POP3, SMTP, HTTPD, FTP	Conexão efetuada GET / HTTP/1.1 GET /favicon.ico HTTP/1.1	root, linux	root, saganet
200.111.183.252	POP3		ortega,igor,sarah,travis,romeo,steve,superman,public,reagan,test2,peterson,pub,website,susan,sonny,screen,training,sitead min,sofia,usr,sys,rosalie,trunk,spencer,ralph,todd, webteste,set,videochat,webuser,secret,sebastian,tiffany,stevens,system,welcome,security,ruth,ronald,torey,records,wesley,valerie,site3,warren,subscribe,richard,zachary,shop,universal,root2,spider,sue,trinity,troy,splinter,seller,wanson,ray,rossi,services,ron,tob,turbo,sara,raymman,terry,rick,vince,visitor,trainer,sportz,tyson,tst	eagle,qwerty,super,123,network,guest,123456,network,abc123,super,internet,test,12345,1q2w3e4r,temp,qazwsx,security,water, *firewall*,guest, microsoft,
189.127.159.154	HTTPD	GET / HTTP/1.1 : GET /favicon.ico HTTP/1.1		
63.230.183.150	HTTPD	Conexão efetuada		
75.127.194.3	HTTPD	CONNECT 205.188.251.26:443 HTTP/1.0 CONNECT 205.188.251.36:443 HTTP/1.0 CONNECT 205.188.251.43:443		

		HTTP/1.0		
124.160.94.218	HTTPD	conexão efetuada		
77.221.156.210	HTTPD	GET //phpmyadmin/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //pma/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //admin/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //dbadmin/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //mysql/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //php-my-admin/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //myadmin/config/config.inc.php?p=phpinfo(); HTTP/1.1 GET //PHPMYADMIN/config/config.inc.php?p=phpinfo(); HTTP/1.1		
88.80.10.1	HTTPD	GET http://88.80.10.1/pp/anp.php?a=U%5C%5DH%5EU_AJZCRF&b=1155&c=22f9 HTTP/1		
188.165.64.240	HTTPD	GET http://proxyjudge1.proxyfire.net/fastenv HTTP/1.1 CONNECT www.google.com:443 HTTP/1.0		
208.109.91.114	HTTPD	GET /user/soapCaller.bs HTTP/1.1		
62.233.197.190	HTTPD	GET //user/soapCaller.bs HTTP/1.1		
94.23.159.248	HTTPD	GET http://proxyjudge1.proxyfire.net/fastenv HTTP/1.1		
		GET		

216.240.147.78	HTTPD	/phpmyadmin//scripts/setup.php HTTP/1.1 GET /phpmyadmin//setup/config.php?type=post HTTP/1.1		
195.243.243.4	HTTPD	GET /w00tw00t.at.ISC.SANS.DFind:) HTTP/1.1		
128.59.14.115	HTTP	GET / HTTP/1.1		
82.113.149.196	HTTPD	CONNECT 205.188.251.43:443 HTTP/1.0 CONNECT 64.12.202.116:443 HTTP/1.0		
189.13.139.133	HTTPD	GET //var/www/index.html HTTP/1.1		
184.154.62.41	HTTPD	GET http://proxyjudge3.proxyfire.net/fastenv HTTP/1.1		
187.63.198.185	FTP,POP3, TELNET	Tentaiva de conexão nas portas: 187,63,198,185,248,223	root,	saganettttt
218.87.16.140	FTP	Conexão efetuada		
206.71.179.51	FTP	Conexão efetuada		
189.83.29.56	FTP, TELNET	HELO 189.83.29.56 :		
208.109.217.92	FTP	Conexão efetuada		
93.186.192.105	FTP	GET /quit HTTP/1.1 : Accept: */* : Accept-Language: en-us : Accept-Encoding: gzip, deflate : User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows 98) : Host: 189.83.72.82 : Connection: Close :		
207.218.218.58	FTP	Conexão efetuada		
85.214.49.14	FTP	Conexão efetuada		
72.55.165.45	FTP	Conexão efetuada		
118.168.142.195	SMTP	Conexão efetuada		
112.105.142.231	SMTP	HELO 189.13.11.156 : MAIL FROM: <baby@gmail.com> :		

		RCPT TO: <vbibiorm@gmail.com> : DATA : QUIT : HELO 189.83.86.237 : MAIL FROM: <baby@gmail.com> : RCPT TO: <vbibiorm@gmail.com> : DATA : QUIT :		
46.4.211.102	SMTP,POP3	Conexão efetuada		
118.166.212.178	SMTP	HELO 189.83.29.56 : MAIL FROM: <z2007tw@yahoo.com.tw> :		
118.167.97.111	SMTP	HELO 189.83.29.56 : MAIL FROM: <b52.b52@msa.hinet.net> : RCPT TO: <k8899@kiss99.com> : DATA : QUIT : HELO 189.83.29.56 : MAIL FROM: <b52.b52@msa.hinet.net> : RCPT TO: <k8899@kiss99.com> : DATA : QUIT :		
114.45.55.182	SMTP	HELO 189.83.86.41 : MAIL FROM: <z2007tw@yahoo.com.tw> RCPT TO: <gk49fawn@yahoo.com.tw> : DATA : QUIT :		
113.111.52.227	SMTP	MAIL FROM:<yhrpvs@veloxzone.com.br> : RCPT TO:<smtp2315@163.com> : DATA : RSET :		
201.50.148.134	SMTP	ehlo [10.100.10.49] : AUTH LOGIN c3RhZmY= :		

114.45.50.191	SMTP	HELO 189.83.72.82 : MAIL FROM: <z2007tw@yahoo.com.tw> RCPT TO: <vkihwpdh@yahoo.com.tw> : DATA : QUIT :		
65.207.113.162	SMTP	Conexão efetuada		
118.160.213.112	SMTP	HELO 189.83.16.182 : MAIL FROM: <hi7188s.pp5975@msa.hinet.net> : RCPT TO: <zz@mail2000.com.tw> : DATA : QUIT :		
123.121.206.185	ECHO	CONNECT secure.ncsoft.com:443 HTTP/1.1 : Accept: */* : Content-Type: text/html : Proxy-Connection: Keep-Alive : Content-length: 0 :		

Tabela 3: Comprometimento, serviços mais temidos, comandos mais utilizados, usuários e senhas mais comuns utilizados pelos Hackers