

Honeypots e Honeynets: Aprenda a detectar e enganar os invasores

Prefácio	3
Conceitos básicos	4
Definição.....	5
Princípios de Engenharia Social	6
Sistemas de Detecção de Intrusos	7
Network Intrusion Detection System	8
Host Intrusion Detection System	10
Segurança por obscuridade	12
Honeypots	13
Riscos e aspectos legais dos Honeypots	13
Localização dos Honeypots	14
Tipos de Honeypots	15
Serviços de alta interação	15
Serviços de baixa interação	15
Honeypots de pesquisa.....	16
Honeypots de produção	16
Honeynets.....	17
Honeynets GEN I e GEN II.....	17
Honeynet real.....	18
Honeynet virtual	23
Centralização de logs.....	33
Honeytokens.....	34
Honeywall	35
Tipos de Firewalls	35
Zona Desmilitarizada.....	36
Instalando o IPTables.....	37
Regras do iptables	38
Utilizando o NIDS Snort	45
Configuração do Snort	47
Serviços	48
DHCP	49
DNS.....	51
VPN.....	57
Outros serviços	63
Softwares	65
Deception Toolkit.....	65
Honeyd	66
KFSensor.....	71
Visualizando informações de log.....	73
Visualizando e alterando assinaturas de ataque.....	76
Specter	79
Configuração de usuários	80
Visualizando incidentes.....	81
Valhala Honeypot	83
Menu Opções.....	85
Menu Configurar	89
Servidor WEB.....	90

Servidor FTP	93
Servidor Finger.....	97
Servidor POP3	99
Servidor SMTP	102
Servidor telnet.....	103
Servidor TFTP	107
Servidor Proxy.....	109
Modo Console	111
PatriotBox	111
Ativando os serviços	113
Configuração dos serviços	116
Utilizando scripts personalizados	118

Prefácio

Este material é a versão digital do livro Honeypots e Honeynets, lançado em 2009. Com exceção do prefácio, seu conteúdo original permanece inalterado. Este material visa demonstrar as possibilidades da detecção de intrusos baseado em um conceito relativamente recente no meio, o de “pote de mel”. Os honeypots chegaram para ficar, não são mais apenas experimentos feitos em laboratórios por estudantes de ciências da computação. Já existem muitos produtos comerciais que comprovam que essa modalidade de IDS é uma realidade muito bem aceita pelo mercado, e seu crescimento aumenta cada vez mais.

Portanto, o objetivo é demonstrar o conceito de um honeypot, as formas de criar e organizar um, entender terminologias como honeynet e honeytoken, e também conhecer softwares que permitem criar um pote de mel na sua rede com grande facilidade. Não pretendo me prender em uma abordagem muito científica, e sim mais prática, demonstrando táticas e dicas de como obter sucesso ao capturar tentativas de invasão ao seu sistema. Inclusive irei demonstrar a utilização de algumas ferramentas que auxiliam na criação de um honeypot, tanto aquelas que criam o pote de mel em si, quanto ferramentas auxiliares como um firewall.

Para comentários, opiniões e críticas, favor enviar para um dos e-mails abaixo:

mflavio@defhack.com
mflavio2k@yahoo.com.br
mflavioaa@gmail.com

Ou visite alguns desses endereços:

Página do Facebook: www.facebook.com/marcosflavioassuncao
Perfil do Linkedin: br.linkedin.com/in/mflavio2k
Canal no youtube: www.youtube.com/defhack

Marcos Flávio Araújo Assunção

Conceitos básicos

Deixe-me perguntar uma coisa: seu computador já foi invadido? Sei que é meio “de rosca”, a pergunta, mas responda com sinceridade. Se já, meus pêsames, torço para que o estrago não tenha sido grande. Se não foi ainda, poderia aproveitar a oportunidade e me responder a outra pergunta: como você sabe disso?

Normalmente, quando um ladrão invade a sua casa, leva algo de valor, de que logo você dá falta. Acontece que as invasões de computadores não são bem assim. Alguém pode roubar a sua senha, mas, com certeza, você nunca se queixará à polícia de que ela “desapareceu”.

A informação, ao contrário das coisas “físicas”, pode ser duplicada, ou simplesmente copiada. Justamente por isso, um “arrombamento digital” pode ser muito difícil (ou quase impossível) de ser percebido, caso o invasor tenha um mínimo de conhecimento de como apagar seus rastros.

Isso, porque estamos falando do seu computador residencial. Em uma empresa ou outro sistema de informática mais complexo, fica ainda mais difícil descobrir uma quebra de segurança. Isso se deve à quantidade absurda de tráfego que existe ali.

Para um entendimento mais fácil, poderíamos comparar isso com um evento real. Imagine que a prefeitura de Salvador instalou câmeras de segurança nas principais avenidas durante o Carnaval. O objetivo seria detectar furtos e pequenos incidentes com os turistas. Mas o tráfego de pessoas é tão grande, que podem acontecer dois problemas:

Primeiro, o falso positivo: um folião se anima e começa a flertar com uma moça, mas quem está analisando os dados das câmeras acredita ser um assalto e toma as medidas necessárias.

Segundo, o falso negativo: um ladrão passa perto de uma vítima e delicadamente fura o celular de dentro do bolso dela, sem que a pessoa e as câmeras percebam o delito, por causa do movimento excessivo.

Isso também acontece no mundo digital. Utilizamos um sistema chamado IDS (Intrusion Detection System, ou Sistema de Detecção de Intrusos, no bom e velho português). Devido ao grande volume de informações, ele pode deixar de identificar ataques ou acreditar que uma ação totalmente legal seja uma tentativa de invasão.

Surgiu, então, uma abordagem diferente... Baseada na abelha e no pote de mel. Se você deixar um pote desses sobre a mesa, criará uma armadilha, que,

uma hora ou outra, irá atrair uma abelha desavisada. Chamamos essa tática de honeypot.

É bem simples a utilização de um honeypot, na realidade. Imagine que você liga um computador em uma rede qualquer. Ninguém sabe da existência daquele computador, exceto você. E, dentro dele, você instala alguns serviços, como servidor de páginas de Internet, servidor de arquivos etc.

Comparando novamente com o mundo real: é como se você comprasse uma casa e a mobiliasse parcialmente, com alguns poucos móveis. A casa teria normalmente duas portas e duas janelas, que servem como entrada e saída (o que seriam os serviços no honeypot de computador). Acontece que você não disse a ninguém que adquiriu o imóvel. Então, a não ser você, qualquer pessoa que entrar e sair da casa é um invasor, com cem por cento de certeza.

O fator de segurança aqui é a obscuridade. Não contando a ninguém sobre o sistema, você filtra e exclui todos os possíveis acessos “benéficos”. Se alguém se conectar ao seu computador, sem dúvida será um invasor tentando ganhar acesso.

Essa é uma metodologia interessante a ser implementada, quando precisamos de um nível de proteção maior em uma rede de grande porte. Além do que, como os crackers normalmente buscam a forma mais fácil de acesso, serão facilmente “enganados” pela armadilha e deixarão de atacar outros locais.

Vamos conhecer em detalhes então, o que exatamente é um honeypot:

Definição



O que é um honeypot? Bom, se considerarmos apenas a tradução dessa palavra estamos nos referindo a um “pote de mel”. Normalmente considera-se que um pote de mel aberto irá atrair não só abelhas, mas ursos, formigas, e outros animais que gostam do adocicado sabor desse alimento. Se pudermos definir o conceito de honeypot em apenas uma palavra, ela seria “armadilha”.

Criar uma armadilha em um computador ou rede pode parecer como um conceito novo para muitos, mas não é. Para se ter idéia essa idéia já foi abordada no livro “Cuckoo’s Egg”, de Cliff Stoll, que foi lançado em 1990. Nesta obra, o autor conta como conseguiu rastrear um hacker alemão que havia invadido os computadores do Lawrence Berkeley Labs. Esse é

considerado o primeiro caso de utilização do conceito de “armadilha” (honeypot) na caça de um cybercriminoso. A partir daí esse conceito já foi largamente utilizado.

Outro exemplo interessante é o filme *Takedown* (chamado no Brasil de “Caçada Virtual”), que conta a história do hacker Kevin Mitnick. No final da história, Tsutomu Shimomura consegue enganar Mitnick ao fazê-lo pensar que estava realizando transferência de seus dados para um local seguro quando na realidade o fazia para o computador de seu rival.

Qual a necessidade utilização de um Honeypot na segurança de redes hoje?. Para responder a essa pergunta devemos primeiro entender os sistemas de detecção de intrusos tradicionais e suas características.

Princípios de Engenharia Social

Antes mesmo de começar a utilizar um honeypot, é necessário entender um pouco de Engenharia Social, pois esse tipo de ferramenta é totalmente baseada neste conceito. Se olharmos pela ótica da lei, poderíamos chamar as pessoas que praticam esses princípios de estelionatários. Acontece que é algo muito mais profundo.

Nós convivemos com engenheiros sociais a nossa vida inteira. Seja aquela tia que tem um jeitinho carinhoso de lhe pedir as coisas, fazendo com que você nunca consiga dizer não. Ou uma namorada/namorado que se utiliza de chantagem emocional para tentar evitar que você saia com seus amigos.

Poderíamos definir então a Engenharia Social como um conjunto de técnicas específicas que permitem manipular os sentimentos e aspirações de um ser humano para benefício próprio. Essas técnicas já são conhecidas e utilizadas a muito tempo por pessoas que aplicam golpes, e muitas vezes passam despercebidas pela maioria.

Normalmente um engenheiro social irá manipular algum dos seguintes pontos: **Curiosidade, Confiança, Orgulho, Simpatia, Culpa ou Medo**. Cada um possui um grupo específico de pessoas no qual causa mais efeito. Por exemplo:

Curiosidade funciona com qualquer tipo de pessoa, pois todos tem um instinto natural de serem curiosas.

Confiança funciona com pessoas mais desconfiadas, que não entregaráo seus “tesouros” em uma primeira tentativa

Simpatia funciona com pessoas bem-humoradas, que gostam de fazer amizades

Orgulho funciona com pessoas que acreditam serem o último biscoito do pacote.

Culpa funciona com pessoas que se abalam emocionalmente com facilidade.

Medo funciona com pessoas que tem fobia de chefes e superiores

Com base em alguns desses pontos, dá para se desenvolver centenas de táticas que podem ser utilizadas.

Um caso que eu particularmente achei interessante foi a de um rapaz que entrou em uma loja de eletrônicos em Belo Horizonte, pegou um aparelho de DVD de última geração que estava em exposição, levou-o ao caixa da loja e perguntou:

- É aqui que vocês consertam aparelhos de DVD com defeito?

A mulher respondeu:

- Não senhor, você se enganou. Aqui só vendemos.

E ele:

- Ok, obrigado.

Logo depois saiu da loja com o DVD player. Só descobriram o roubo no outro dia ao contarem o estoque.

Mas o que nos interessa é a Engenharia Social relacionada aos honeypots. Nesse caso, a curiosidade é o melhor ponto a ser explorado pois funciona com todos os grupos de indivíduos. Você pode utilizar-se de arquivos com nomes curiosos, de e-mail falsos, e todos os tipos de artifícios para atrair um intruso. Consulte a seção sobre honeytokens para maiores detalhes.

Sistemas de Detecção de Intrusos

Tipicamente, um sistema de detecção de intrusos é o programa que é utilizado como “retaguarda”. Após todo o resto ter falhado, é ele que vai conseguir detectar alguma possível brecha que alguém possa ter utilizado para tentar um ataque. Sistemas desse tipo são utilizados hoje em todos os tipos de empresa pois as ferramentas de segurança tradicionais não conseguem mais segurar a barra sozinhas.

Um ótimo exemplo é o caso do firewall. A maioria das ferramentas de firewall apenas seguem cegamente regras que permitem que certo tráfego entre ou saia da rede, mas sem se preocupar com o conteúdo desses pacotes passantes.

Existe um exemplo que explica bem essa situação : imagine que em uma certa loja de roupas, o responsável pela segurança foi treinado para permitir que apenas as mulheres entrem na loja, e não os homens. Ele consegue fazer isso muito bem. Mas se alguma das mulheres entrar com uma metralhadora para assaltar a loja ao invés de uma bolsa, ele vai deixar sem problemas. Afinal, se a regra padrão foi seguida, ele não se importa com os detalhes.

Um IDS pode ser utilizado da duas formas distintas. Para cada uma dessas formas existem programas e metodologias completamente diferentes.

Network Intrusion Detection System

O Network Intrusion Detection System, ou simplesmente NIDS, é o tipo de IDS que possui a sua atuação baseado na rede como um todo. Ao invés de ter que se posicionar individualmente em cada máquina que deve monitorar (como é o caso do HIDS, que veremos pouco à frente), ele irá monitorar todo o tráfego da rede para identificar ameaças e tentativas de ataque.

Essa abordagem possui algumas vantagens e desvantagens: a maior vantagem óbvia é o fato de conseguir detectar ataques de modo muito mais abrangente do que os IDS baseado em host. Mas também há um ponto fraco: ele não consegue monitorar o uso de aplicações nos computadores, assim como a utilização maliciosa da memória ou consumo intenso de CPU. Para chegar a esse nível, deve ser utilizado um HIDS.

Dentro do conceito de IDS, podemos utilizar duas metodologias para a detecção de ataques.

- *Detecção de comportamento malicioso baseada em assinatura*
- *Detecção baseada em anomalia*

Detecção baseada em assinatura

Esse tipo de detecção visada identificar ataques conhecidos com base em assinaturas baseadas no tipo de comportamento malicioso. Uma assinatura é na realidade um trecho de informação que um pacote deve conter. Suponha que um pacote se destinando a um servidor HTTP / WEB contenha o pedido “/admin”. Isso pode ser uma tentativa de identificar um possível diretório de administração do site. Portanto, se um NIDS baseado em assinatura estivesse monitorando o tráfego desse servidor, ele provavelmente acusaria um ataque.

Entretanto é preciso frisar que a assinatura dever baseada em características normalmente “invariáveis” dos ataques. Por exemplo, podemos citar o fato da maioria dos exploits tentarem lançar um shell como payload. Detectando o Shell, você pega a maioria dessas tentativas de exploração (com exceção daquelas com o código do shell alterado propositalmente para evitar detecção.)

As vantagens dessa forma de detecção é a velocidade e identificação da maioria dos ataques. Mas existem desvantagens:

Necessidade de atualização freqüente do banco de dados de assinaturas: Sem novas assinaturas um NIDS desse tipo se torna inútil pois a maioria dos ataques recentes ele não vai ser capaz de reconhecer.

Só consegue detectar ataques conhecidos: Esse também é um grande problema. Afinal, o NIDS só irá detectar ataques que já forem conhecidos do público. Certos ataques novos, diferentes, provavelmente não serão capturados pelas regras.

Alto número de falsos positivos: Esse é um problema que praticamente não existe nos honeypots, como explicarei mais à frente. Um falso positivo é um alerta gerado pelo NIDS dizendo que um ataque ocorreu, quando na realidade era apenas um usuário legítimo tentando acessar alguma coisa. Ou seja, um falso ataque que foi acusado como verdadeiro. Esse tipo de coisa causa transtorno por gerar muito lixo nos logs.

Um software popular que age como um NIDS baseado em assinaturas, é o **Snort**:

Detecção baseada em anomalia

Esse segundo conceito de detecção não precisa de assinaturas ou regras específicas. Ele irá tentar detectar o que chamamos de uma anomalia. Imagine que foi definido um perfil normal do tráfego da rede, criado após um

ano de análise sistemática. Sabe-se com isso por exemplo que o horário de menor atividade é o horário do almoço, que o maior período de pico é no final da tarde quando os funcionários estão realizando o seu backup online, etc. Esses dados definem o que podemos chamar de um modelo padrão ou normal do sistema.

Se alguma coisa aparecer que fugir muito à esse modelo o NIDS irá soar como uma anomalia. Por exemplo, imagine que um invasor esperto ganhou acesso às máquinas da rede instalando um cavalo de tróia. Mas ele só utiliza esse acesso durante o horário de almoço dos funcionários para evitar ser descoberto. E nesse intervalo transfere grandes quantidades de dados. Esse é um comportamento completamente anormal do sistema para aquele horário, e portanto, o NIDS acusará um erro.

Assim como o outro tipo, um NIDS baseado em anomalia também não é perfeito. Ele pode por exemplo sofrer também de falsos positivos. E se um funcionário resolveu durante uma semana permanecer durante o horário de almoço na empresa para baixar músicas pela internet? Não é necessariamente uma ataque.

Host Intrusion Detection System

O Host Intrusion Detection System, ou HIDS, é um tipo de IDS baseado em host, ou seja, ele deve ser instalado individualmente nas máquinas. Em um primeiro momento pode parecer que não é uma situação muito prática, já que o NIDS consegue monitorar o tráfego de todas as máquinas ao mesmo tempo. Mas existem motivos que fazem com que um HIDS seja uma ferramenta indispensável. Entre alguns fatores que ele consegue detectar, podemos citar:

- *Utilização indevida e excessiva da memória*
- *Processos estranhos e de comportamento suspeito*
- *Conexões de rede suspeitas*
- *Utilização da CPU*
- *Utilização das chamadas de sistema (System Calls)*
- *Utilização do disco (leitura, gravação, criação e exclusão de arquivos e pastas)*

Esse é um nível bem mais profundo do qual um NIDS trabalha. Por isso mesmo, certos ataques só podem ser detectados por um HIDS (como por exemplo, cavalos de tróia que criptografam a sua comunicação, inutilizando o sistema de assinaturas do NIDS).

Mas um problema ainda persiste. Poderia continuar havendo falsos positivos? Definitivamente sim. Por exemplo: uma utilização anormal da CPU poderia não ser devido à um software malicioso, e sim a um problema físico, como superaquecimento. Existem diversas variáveis para identificar um

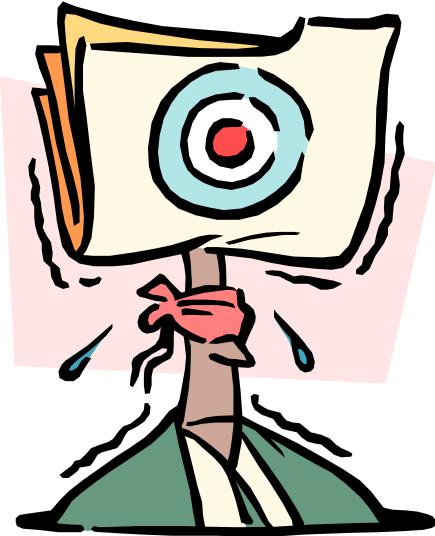
comportamento como anormal e é comum que muitas das vezes os alarmes sejam falsos.

Como exemplo de um host based detection system conhecido, podemos citar OSSEC HIDS:

The screenshot shows the OSSEC Web Interface (BETA2) running in Mozilla Firefox. The interface has a search bar at the top with the URL <http://xxx.ossec.net/oswui/index.php?f=s>. Below the search bar is a navigation menu with links to Main, Search, Stats, OSSEC Site, and About. The main content area is titled "Search options:" and includes fields for "From" (2007-01-01 05:53), "To" (2007-01-04 09:53), "Minimum level" (1), "Category" (Multiple auth failures), "Pattern" (empty), "Rule id" (empty), "Srcip" (empty), "User" (empty), and "Max Alerts" (20000). A "Search" button is located below these fields. Below the search options is a section titled "Results:" with the sub-section "Total alerts found: 6". Under "Total alerts found:", there are three hyperlinks: "+Severity breakdown", "+Rules breakdown", and "+Src IP breakdown". The "Alert list" section displays a list of six alerts, all of which are "Multiple auth failures". The first alert is expanded to show its details:
2007 Jan 03 16:28:28 Rule id: 5720 level: 10
Location: (jul) 192.168.2.0->/var/log/messages
Src IP: 192.168.21.56
Multiple SSHD authentication failures.
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31288 ssh2
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31299 ssh2
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31298 ssh2
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31297 ssh2
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31296 ssh2
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31295 ssh2
Jan 3 15:09:07 slackerr sshd[3077]: Failed password for root from 192.168.21.56 port 31294 ssh2

A grande maioria dos softwares que permitem a criação de um Honeypot, como o KFSensor, Honeyd, Specter e Valhala atuam como um sistema de detecção de intrusos do tipo HIDS. Mas eles ainda se diferenciam dos HIDS tradicionais por se basearam no princípio da Segurança por obscuridade.

Segurança por obscuridade



Deve ter ficado claro ao estudar sobre os sistemas de detecção de intrusos (IDS) que um de seus maiores problemas é a questão dos falsos positivos. Mas esse é um problema que nunca poderá ser resolvido por essas ferramentas tradicionais, simplesmente por um simples fato: ao monitorar o tráfego de uma rede legítima, eles tem o árduo trabalho de separar o que é um ataque e o que é um acesso inofensivo. Mesmo o sistema mais inteligente e bem desenvolvido, pode falhar.

Agora podemos responder à pergunta que foi feita no início do capítulo: qual a principal finalidade de um honeypot? Claro, a resposta também não é tão simples.

Primeiramente, um honeypot não possui falsos positivos como os IDS normais. Mas como isso é possível? Simplesmente porque ele se baseia em um conceito que gosto de chamar de *Segurança por obscuridade*.

Entender esse conceito é mais fácil que parece. Imagine que você pegou o seu notebook, levou pela primeira vez para a sua empresa e o conectou na rede. De repente o seu firewall pessoal acusa uma tentativa de acesso às suas pastas de compartilhamento.

Isso pode ou não ser considerado um ataque? Com certeza pode. Afinal, como você levou o notebook ali pela primeira vez, *ninguém tinha conhecimento da*

existência dele, então qualquer conexão ou acesso é 100% garantida uma tentativa de invasão ou exploração do sistema.

Essa é a idéia da segurança por obscuridade. Por ninguém conhecer o sistema, todo o tráfego virá de ataques.

Sabendo disso, podemos dizer que o honeypot possui como finalidade ajudar as ferramentas de detecção de intrusos adicionais, ajudando-as a melhorar suas assinaturas através do descobrimento de novos tipos de ataque. Também pode-se utilizar um pote de mel como um sistema de IDS totalmente independente. Tudo depende do objetivo. Veremos mais sobre os modos de se utilizar um honeypot no próximo capítulo.

Honeypots

As maneiras de se montar e utilizar um pote de mel são muito variadas. Esse próprio termo em si pode significar várias coisas, dependendo do contexto. O objetivo desse capítulo é apresentar alguns dos riscos que podem existir em se utilizar um honeypot, as diferenças entre os tipos de pote de mel existentes e quando deve-se cada um deles. Também veremos o que é Honeynet, componentes necessários para montar uma, seja real ou virtual. E terminaremos falando sobre Honeytokens, que também são essenciais para o funcionamento dos Honeypots.

Riscos e aspectos legais dos Honeypots

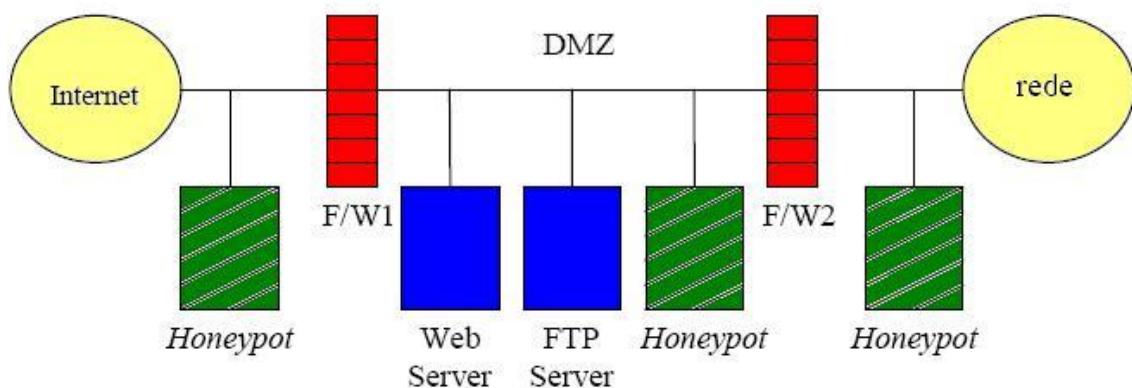
Utilizar um honeypot é muito interessante do ponto de vista da facilidade de se detectar invasões e poder assim melhorar os sistemas existentes de detecção de intrusos. Mas é importante ressaltar que também é uma faca de dois gumes: caso um honeypot seu seja comprometido e o invasor utilizá-lo para atacar outras redes, isso pode lhe causar um grande problema. Portanto é importante pesar e balancear quando utilizar os serviços de baixa interatividade, que não oferecem praticamente risco algum com aqueles de alta interatividade que, apesar de ajudarem a recolher mais informações interessantes, podem fazer o feitiço se virar contra o feiticeiro.

Outra questão muito comentada sobre os honeypots é em relação ao aspecto jurídico de sua utilização. Algumas pessoas alegam que um honeypot induz alguém a fazer algo errado. Isso de forma alguma é verdade. Um pote de mel não está induzindo ninguém a realizar nada de errado, até porque muitas vezes ele é um computador como qualquer outro da rede. Apenas a finalidade de colocá-lo ali é que foi diferente. O honeypot não está sendo exibido para ninguém, o invasor entrou porque quis.

É o mesmo que você colocar um alarme em uma porta da sua casa, já imaginando que alguém possa tentar arrombar. Você não está incentivando ninguém, apenas se preavendo de algo que pode acontecer ali.

Em relação aos aspectos de privacidade também, não há nada de errado em usar um honeypot. Afinal, ao monitorar o seu próprio sistema, você não está quebrando a privacidade de ninguém. O invasor entrou ali porque ele quis, e já estava mal intencionado a fazer algo errado. Pense que seria a mesma coisa que instalar câmeras de segurança na sua casa. Um ladrão reclamaria para um juiz que você o fez?

Localização dos Honeypots



Um honeypot pode ser localizado em três locais distintos, considerando que uma instituição possui uma DMZ (zona desmilitarizada):

1 – Antes do primeiro Firewall (F/W1): Nesse caso, o Honeypot vai ficar propositalmente exposto ao máximo, sem ter nenhum tipo de proteção. É uma situação que provavelmente o invasor conseguirá causar mais danos, mas justamente por isso é o mais interessante do ponto de vista de capturar ações maliciosas, que é o objetivo de um honeypot de pesquisa, como veremos mais à frente.

2 – Na zona desmilitarizada (DMZ): Nessa segunda situação, o Honeypot estará no mesmo nível dos principais servidores reais da rede. Ele não ficará tão exposto quanto o primeiro pote de mel, mas ainda assim passível de receber ataques. Por estar dentro da mesma faixa, um invasor pode tentar fazer uma varredura de portas, encontrar o honeypot e acreditar que se trata de um servidor de banco de dados ou de correio. Eu considera esse o melhor local para posicionar o honeypot.

3 – Junto à rede interna, após o segundo firewall (F/W2): É um local interessante, pois onde está posicionado esse terceiro honeypot dificilmente receberá ataques vindos da internet. Claro, pode acontecer mas a freqüência será bem menor. Para que serve então posicionar nossa armadilha aqui? Inverta o pensamento: e se em vez de detectar invasores da internet quisermos detectar funcionários que possam estar acessando dados indevidos dentro da

rede? Esse honeypot então é o ideal para encontrar atacantes que venham da rede interna.

Tipos de Honeypots

Antes de entender como diferenciar os tipos de honeypots existentes, o que é feito com base no objetivo deles, devemos entender os serviços que um honeypot utiliza. Basicamente, dois tipos de serviços existem:

Serviços de alta interação

Ao criar um pote de mel, você irá adicionar serviços nele para que possam atrair invasores. Esses serviços podem ser um programa que funcione como servidor de correio ou de transferência de arquivos. Ou mesmo, você pode querer fornecer acesso ao shell de comandos do seu sistema operacional, dando a ele controle total. Então, se você entregar ao atacante um sistema real com serviços que funcionem de verdade, esses serviços são de alta interação.

Mas isso não poderia ser perigoso? Sim, definitivamente. Ao entregar uma máquina com serviços reais para um invasor existe a chance dele conseguir comprometer esse computador e conseguir acesso a outro da rede. Afinal, ele terá um sistema operacional real à sua disposição para fazer o que quiser.

Porém, há também grande vantagem: dificilmente um honeypot que use serviços de alta interação é detectado pelo invasor. Pois é quase impossível diferenciá-lo de qualquer outro servidor ou computador desktop da rede, já que ele possui todos os serviços.

Um exemplo: você compra um computador para utilizar como honeypot. Instala o Windows XP com SP2 e desabilita as atualizações automáticas, deixando-o com mais buracos do que um queijo suíço. Compartilha algumas pastas para serem acessadas por todos, instala alguns programas como tocador de mp3 para parecer que o computador é realmente utilizado (falaremos mais sobre isso em honeytokens). Enfim: se alguém atacar e invadir essa máquina, o que provavelmente vai acontecer, essa pessoa vai simplesmente achar que você é desleixado e não que caiu em uma armadilha.

Serviços de baixa interação

Os honeypots que tem serviços de baixa interação, são o extremo oposto dos de alta interação. Todos os serviços, seja um Shell do sistema ou um servidor de correio, são simulados. O invasor nunca terá acesso ao sistema real, apenas à versões simuladas dos mesmos. Isso é muito vantajoso do ponto de vista de segurança, pois se o que o atacante está vendo é apenas um jogo de “faz de conta”, ou seja, uma simulação, ele não conseguirá em

hipótese alguma comprometer a segurança do sistema e ganhar acesso à outra máquina da rede.

Nem tudo é perfeito, entretanto: qualquer invasor com um nível mínimo de habilidade conseguirá detectar com grande rapidez um serviço de baixa interação. Provavelmente após três ou quatro comandos, no máximo. Mas olhando por outro lado, mesmo ele descobrindo a armadilha, já vai ter sido detectado e seu ataque registrado.

Agora sim, após entender os dois tipos de serviços existentes, conseguimos entender os objetivos de cada honeypot:

Honeypots de pesquisa

Um honeypot de pesquisa não tem como objetivo primário ser utilizado como uma ferramenta de IDS. O que ele pretende é realmente ser atacado várias vezes, e com isso estudar todos os detalhes de cada ataque. Cada arquivo que o invasor acessar, cada senha que ele digitar, cada comando, absolutamente tudo será salvo e estudado. Quando se deve montar um honeypot de pesquisa? Depende do caso. Alguns estudantes podem montar um honeypot desses para realizar um trabalho acadêmico sobre os tipos de ataques mais freqüentes... ou mesmo uma empresa de segurança que fabrique antivírus ou outras ferramentas de proteção podem criar um ambiente assim para estudar novos ataques e assim atualizar seus produtos com o que há de mais recente.

É importante ressaltar que devido ao objetivo de um honeypot de pesquisa, devem ser utilizados apenas serviços de alta interação. Afinal, se o que se quer é capturar o máximo de ações de um atacante não dá para entregar a ele um ambiente simulado. Ele irá simplesmente descobrir a farsa, e nunca mais se conectar ali. E com isso o trabalho de pesquisa vai por água abaixo.

Honeypots de produção

Honeypots de produção são o oposto dos de pesquisa. Sua intenção é praticamente apenas detectar intrusos na rede e tomar as providências contra esses invasores o mais rápido possível. É o que normalmente seria utilizado em alguma empresa ou instituição que deseja proteger a sua rede.

Nesse caso, de forma alguma queremos fornecer acesso real ao sistema pro invasor. Afinal, queremos que ele saia da rede o mais rápido possível e não que permaneça passeando à vontade. Então, em um honeypot de produção devem ser utilizados apenas serviços de baixa interação, pois são simulados e

não oferecem nenhum risco ao sistema real. Não há problema nesse caso o atacante descobrir a armadilha.

Honeynets

Para entender uma honeynet é necessário levar o conceito de honeypot a um sentido mais amplo. Se ao invés de você utilizar um único computador como armadilha, que tal uma sala cheia deles formando uma rede com o único objetivo de servir de armadilha? Isso é uma Honeynet.

A Honeynet então basicamente é uma rede na qual todo o tráfego que entra e sai do gateway/roteador é malicioso. Lembre do conceito de Segurança por Obscuridade: ninguém conhece os computadores dessa rede, portanto qualquer pacote que se destine a qualquer um deles é um ataque em potencial.

Dentro do contexto de uma Honeynet, cada computador que faz parte dela é considerado um Honeypot.

Existe na internet um projeto chamado Honeynet Project (www.honeynet.org) que possui inclusive uma versão brasileira (www.honeynet.org.br). Nesse site eles demonstram diversos exemplos e dicas de como configurar corretamente a sua “rede armadilha”. Visite que é bem interessante.

Montar uma rede dessas não é tarefa simples, pois dependendo da forma que escolher pode dar um grande trabalho configurá-la. Existem duas maneiras de se criar Honeynets: uma real e outra virtual.

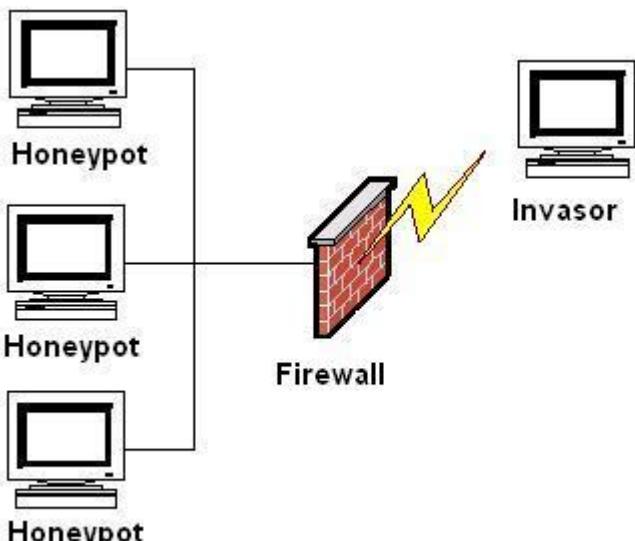
Honeynets GEN I e GEN II

Existe atualmente duas “gerações” de honeynets. A GenI que é a primeira geração usa tecnologias básicas para capturar e controlar as atividades de um atacante. Normalmente nesse caso, quando se montava uma honeynet o gateway atuava apenas no papel de firewall não utilizando ferramentas mais avançadas de detecção de intrusos, como um Network Intrusion Detection System (NIDS). Ou o IDS era rodado em uma máquina a parte, ou simplesmente não utilizado. No caso da GenI em muitos casos dependia-se da utilização de algum software extra para a obtenção e gerência dos logs as invasões (como muitos programas de se criar honeypots, vistos no último capítulo).

Já a Gen II surgiu com a concepção de usar o “honeywall”, um computador gateway que irá agir como Firewall e IDS ao mesmo tempo. Além disso, a Gen II utiliza tecnologias bem mais avançadas que podem incluir especificar uma bridge (dispositivo de da camada de enlace do modelo osi) que permite bloquear ou modificar ataques que entram e saem da Honeynet. A GenII usa ferramentas bem mais avançadas do que as usadas na primeira geração.

Fala-se na utilização da GEN III, já incluindo recursos de virtualização mais complexos, os quais falaremos de alguns em Honeynet virtual.

Honeynet real



Uma honeynet real é aquela que todos os computadores e componentes são físicos. Ou seja, cada computador honeypot existe fisicamente dentro da rede. É o tipo rede que mais próximo chega de uma situação verdadeira.

Porém, há um problema: custo. Montar uma honeynet real requer muito dinheiro pois é necessário comprar muitos equipamentos para conseguir fazer essa rede funcionar. E nem todos tem orçamento para realizar esse tipo de investimento.

A descentralização dos computadores honeypots é uma vantagem em relação à segurança. Provavelmente nenhum deles ficará muito sobrecarregado, já que os ataques provavelmente serão distribuídos entre todos.

Principais componentes de uma honeynet real

A seguir listo alguns itens que devem ser adquiridos para montar uma honeynet de forma bem sucedida. Lembrando que é como montar qualquer outra rede, apenas o objetivo desta é diferente das demais.

Servidores / Computadores



O que seria de uma honeynet sem os computadores honeypot? Apesar de que pode existir apenas um único computador na rede, um investimento para se comprar cinco ou mais computadores é justificado se você pretende montar uma rede permanente para servir como um “honeypot de pesquisa” (ou seja, quer estudar os invasores).

Switch / Access Point



Caso monte uma honeynet cabeada, você irá precisar de um switch para interligar os computadores. Pode optar por uma Access point no caso de uma rede Wireless:



Cabos



Necessários caso você não resolva trabalhar com uma rede wireless (sem-fio). Prepare-se para adquirir uma grande quantidade de cabos. Prefira os UTP Cat5 já que a rede não necessita de um investimento de velocidade ou segurança que exige um STP ou UTP Cat6.

Periféricos



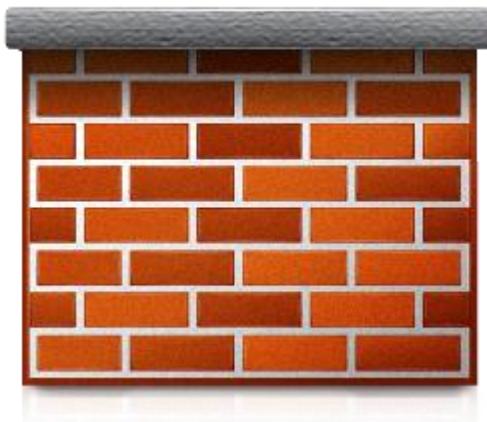
Algumas pessoas não consideram a utilização de periféricos em uma honeynet, como webcams, impressoras, scanners, etc. Eu particularmente acho interessante por dois motivos: primeiro para passar a impressão de uma rede “verdadeira”. E segundo, um invasor pode detectar que existe uma webcam ligada a um computador e tentar visualizar as imagens, por exemplo. Imagine a confiança que isso passaria de que se trata de uma rede real.

Roteador



O roteador é um componente essencial, afinal é ele que vai servir de gateway entre a sua rede e outra rede interna ou a internet. Se você tiver um roteador e apenas um único computador honeypot, já tem a sua própria Honeynet. Não precisa ser um roteador muito sofisticado, como os da Cisco. A menos que você deseja algumas funções de roteamento mais avançadas, como as existentes no Cisco IOS, opte por um dlink ou linksys.

Firewall



É essencial ter um firewall entre o roteador e a rede honeypot. Mas não seria melhor deixar a rede completamente escancarada? Não necessariamente. Grande parte das redes hoje possuem um nível mínimo de proteção, portanto, deixar a rede completamente sem um firewall pode levar à desconfiança de alguns invasores mais experientes. Isso, sem falar nos logs que o firewall nos fornecerá do tráfego que está sendo filtrado por ele. Na realidade é o Firewall e o IDS que vão fazer todo o trabalho de capturar e gravar os pacotes que entra e saem da rede.

Um bom firewall que pode ser utilizado, é o IPTTables, para Linux e similares.

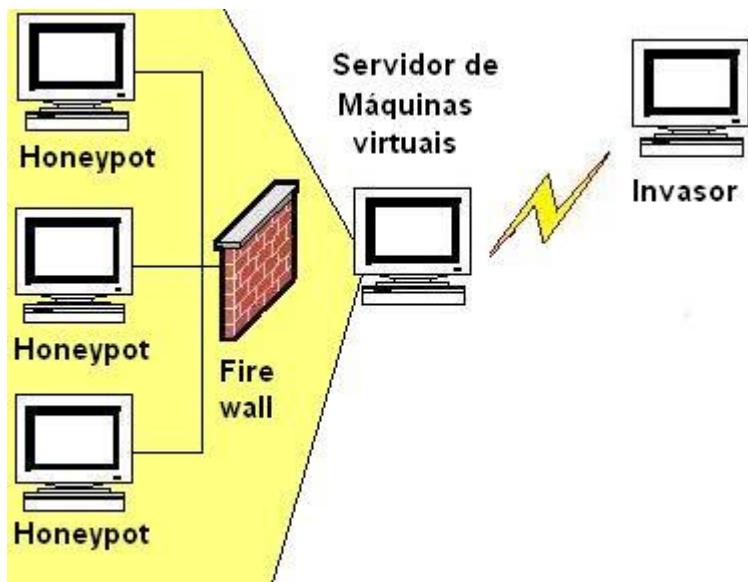
Esse é um tópico tão importante em uma honeynet, que iremos abordá-lo em seu próprio capítulo mais à frente no livro.

IDS



No capítulo anterior, expliquei um pouco sobre os tipos de sistemas de detecção de intrusos existentes. Em uma rede honeynet, é essencial utilizar um NIDS baseado em assinatura, como o Snort. Assim, além de capturar todo os pacotes você ainda pode ter alertas gerados sobre os ataques mais perigosos cometidos contra a rede. Utilize o IDS preferencialmente no mesmo computador que será instalado o firewall. Esse computador se ligará com o roteador e o resto da rede, se tornando o gateway. Lembre-se: o tráfego de uma honeynet só pode ser corretamente capturado e monitorado ao se passar pelo gateway. O snort pode ser obtido em www.snort.org.

Honeynet virtual



Um dos maiores problemas da Honeynet real é o alto custo de montagem da mesma. Claro, como disse antes, não são todos que podem gastar alto com a montagem de equipamentos. Existe entretanto, uma maneira mais simples de se montar uma honeynet: utilizando máquinas virtuais.

A virtualização é um recurso que vem ganhando mais e mais espaço a cada dia, seja em ambientes acadêmicos ou mesmo em grandes corporações.

Mas o que é exatamente uma máquina virtual? É um programa que você pode utilizar para rodar um outro sistema operacional dentro do seu. O software de virtualização utiliza-se de seu acesso privilegiado ao hardware da máquina para conseguir tal feito. Então, eu posso utilizar Windows Vista e rodar o Linux Ubuntu em uma máquina virtual sem problemas. De fato, você pode rodar várias máquinas virtuais em um mesmo servidor, tudo depende da quantidade de processamento, disco e memória RAM que se tem disponível.

Isso irá reduzir de forma drástica os custos da honeynet, já que um único computador físico conseguirá rodar todas os honeypots, que serão computadores virtuais.

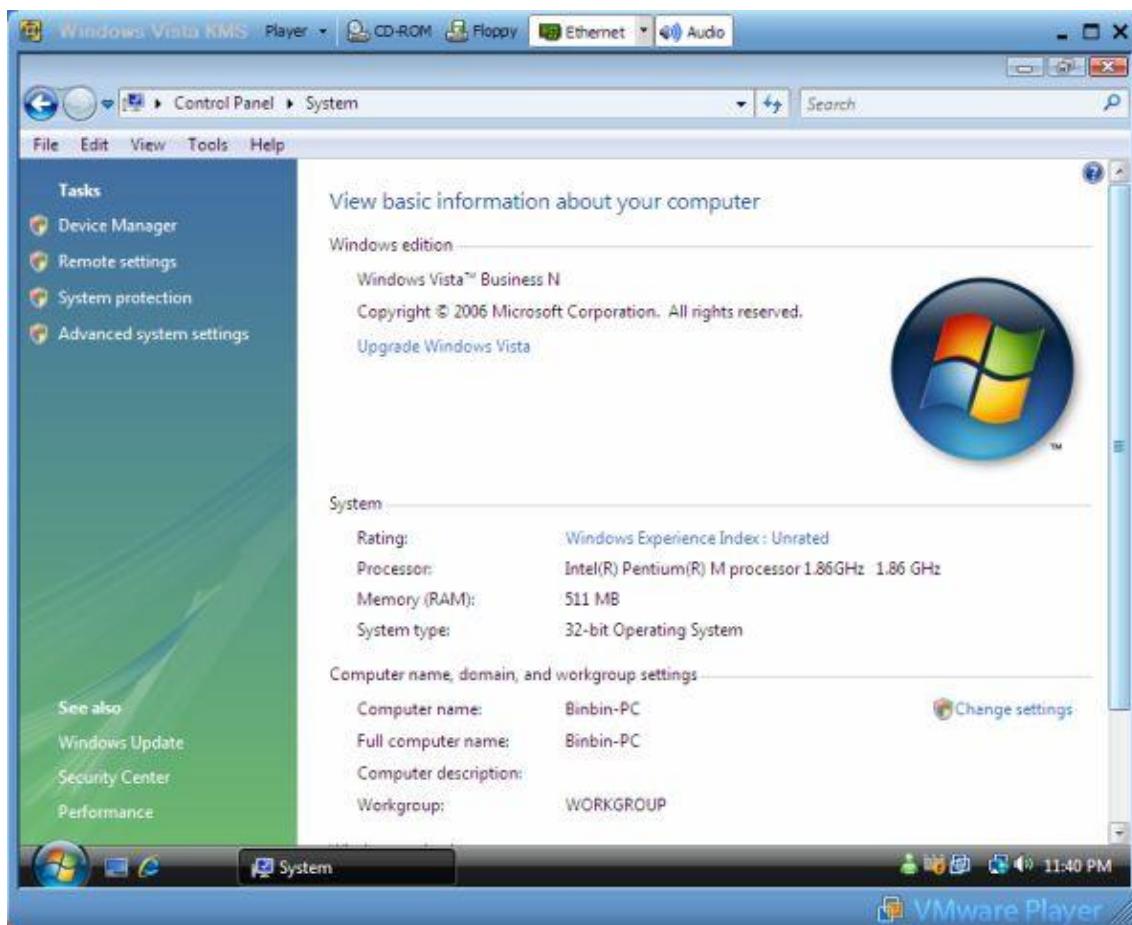
Vamos conhecer alguns dos programas de virtualização disponíveis no mercado:

Vmware

O Vmware foi o pioneiro em virtualização. A empresa oferece essa tecnologia já a bastante tempo, e com recursos bem avançados como a capacidade de se utilizar dispositivos USB nas máquinas virtuais e também o uso de redes wireless.

Existem várias versões, como o Workstation, Server, e o Player. Este último serve simplesmente para rodar máquinas virtuais já criadas em alguma das outras versões, que são comerciais

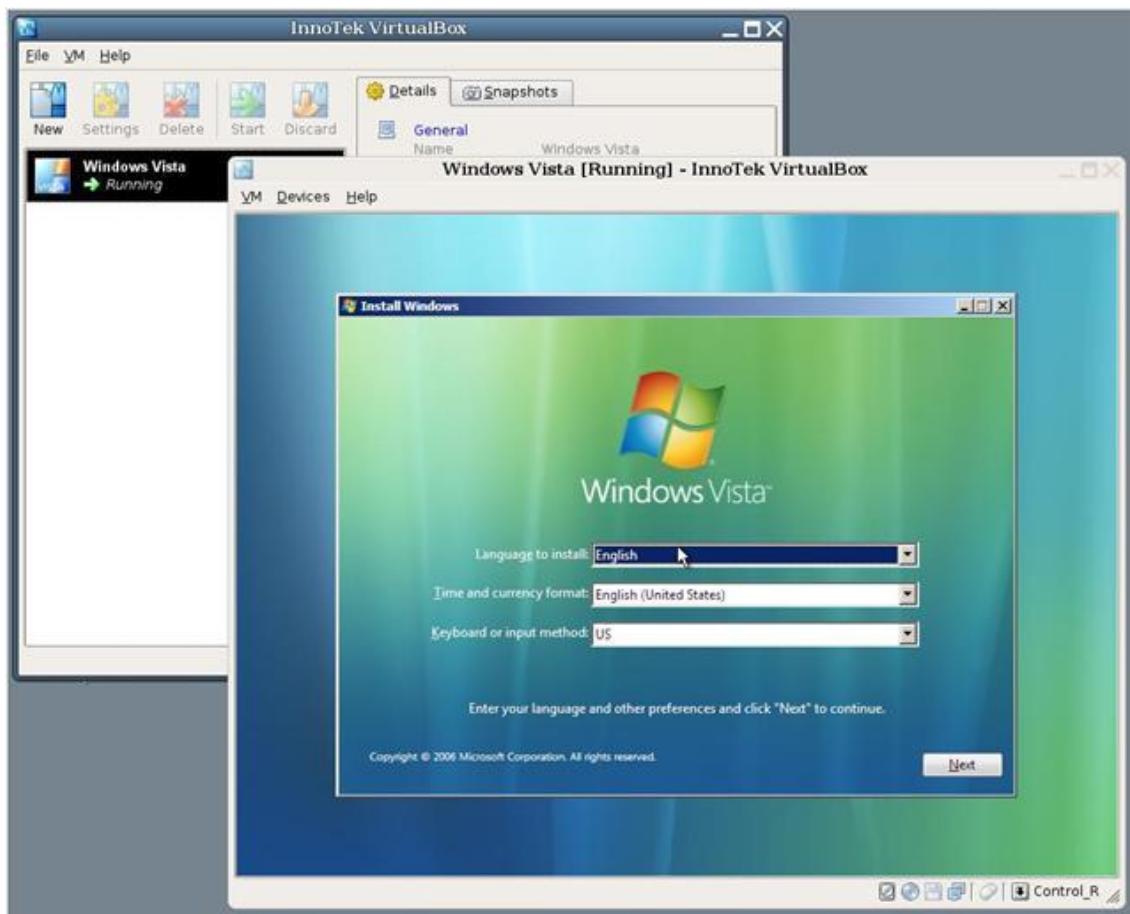
Mais informações em www.vmware.com . Abaixo uma máquina virtual vmware rodando o Windows Vista.



VirtualBox

O VirtualBox é hoje o software de virtualização que mais cresce em popularidade. Criado pela Sun Microsystems, a mesma que desenvolveu a linguagem Java e o sistema operacional Solaris, ele começou como um completo desconhecido e se tornou o principal rival do Vmware. Existem repositórios na internet com dezenas de máquinas virtuais do VirtualBox disponíveis para serem baixadas. Esse software é gratuito para uso pessoal. Mais informações em : www.virtualbox.org

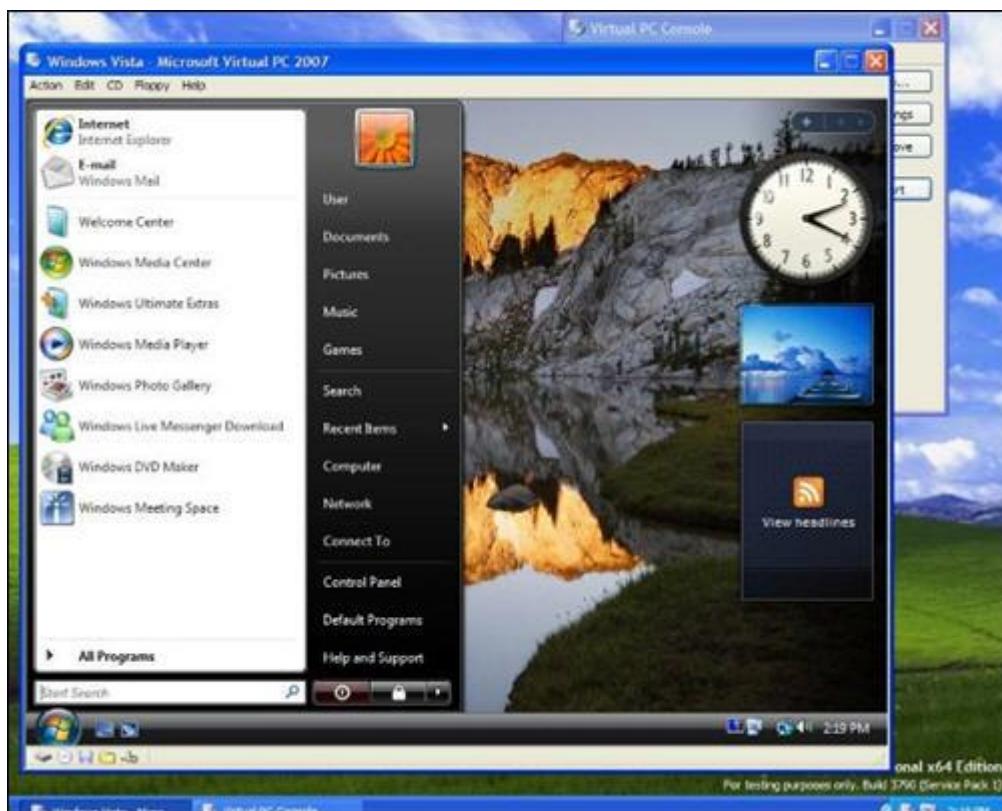
Abaixo, imagem de uma máquina virtual do VirtualBox rodando o Windows Vista



Virtualpc

Essa é a resposta inicial da Microsoft ao fenômeno da virtualização. Muito antes de incluir recurso virtualizados no Windows Server 2008, a empresa começou a comercializar o virtualpc, que logo se tornou um sucesso. Através de uma decisão acertada, a Microsoft liberou o programa para uso gratuito. É atualmente uma das máquinas virtuais de mais fácil utilização, apesar de ter alguns problemas de compatibilidade com certos sistemas operacionais e de não possuir muitos recursos. Você pode obter esse programa no site da Microsoft, www.microsoft.com.br ou em sites de download como o www.superdownloads.com.br

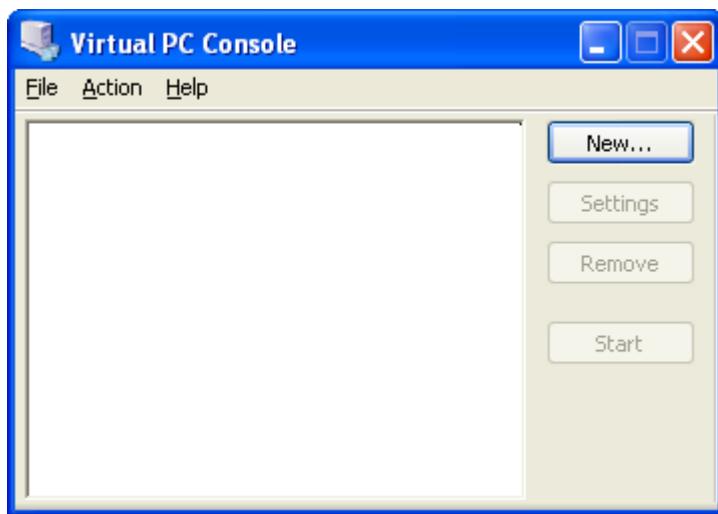
Veja abaixo, uma máquina virtual do virtualpc rodando o Windows Vista:



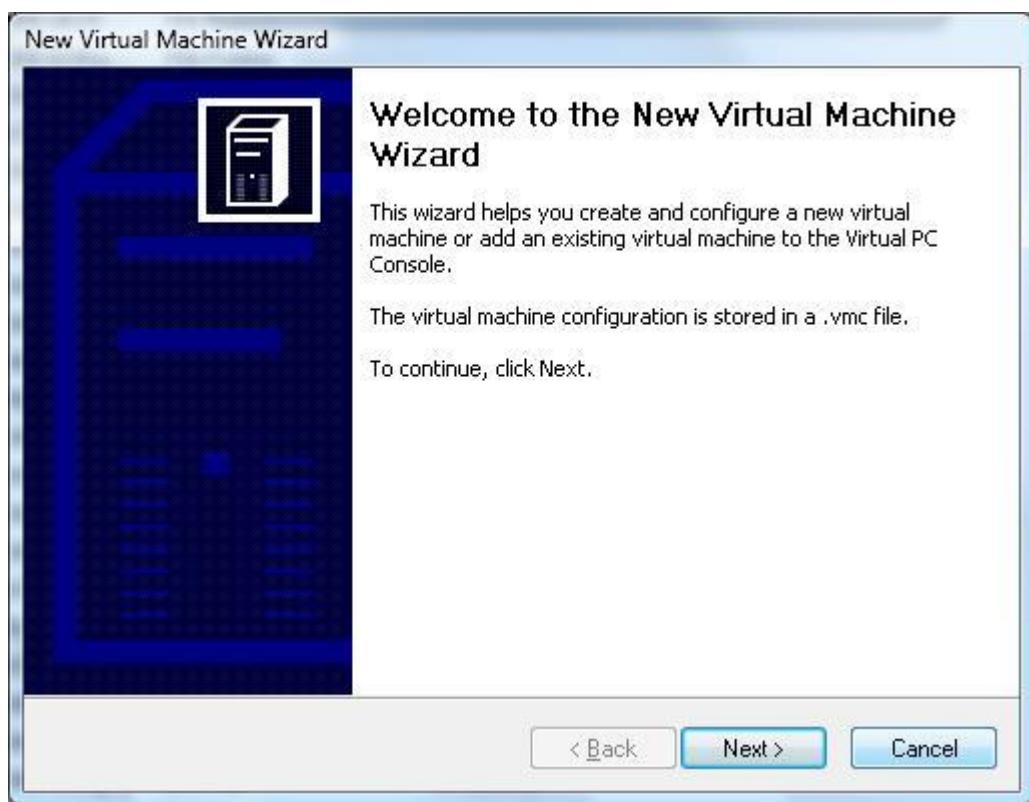
Irei utilizar o virtualpc para exemplificar o processo de criação de uma nova máquina virtual para servir de computador honeypot. Não há grande diferença entre esse processo no programa da Microsoft e o VirtualBox ou o Vmware.

Criando uma nova máquina virtual no Virtual PC

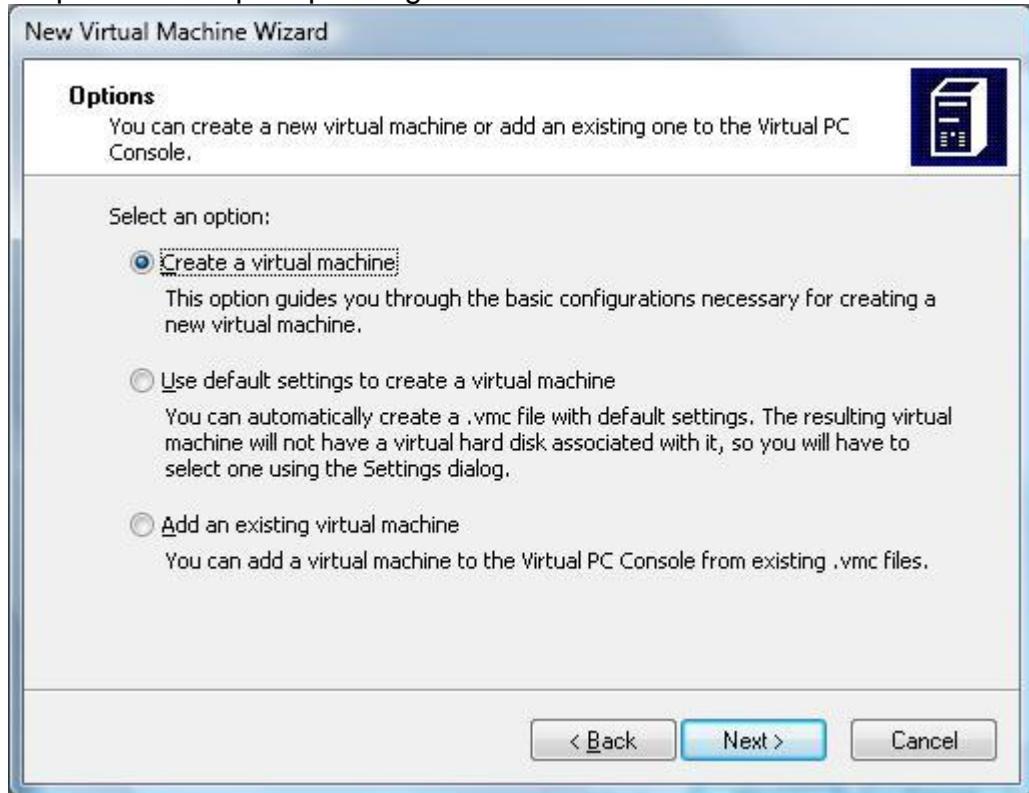
Primeiramente baixe o Virtualpc 2007 ou versão mais recente e instale no seu computador. Abra o ícone “**Microsoft Virtual PC**”, e a seguinte telá aparecerá:



Clique no botão “New”. Isso iniciará o processo de criação de uma nova máquina virtual. Observe:



Clique em Next para prosseguir.

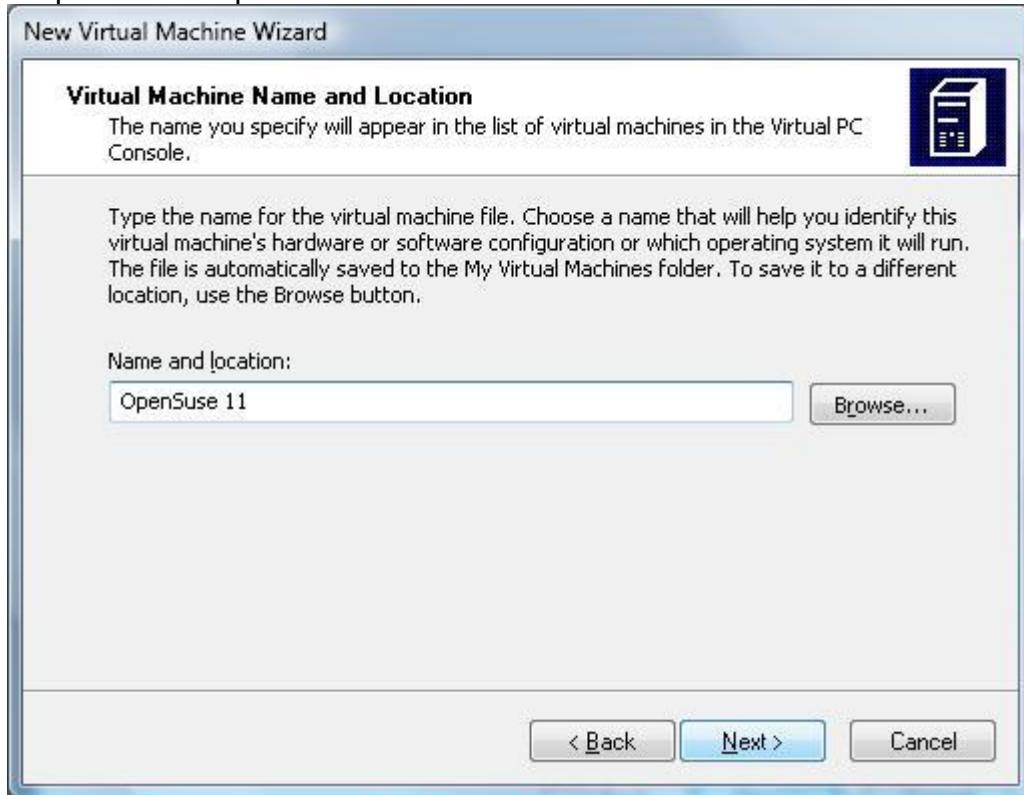


Escolha a primeira opção: "Create a virtual machine". Ela vai permitir criar uma máquina virtual completamente nova.

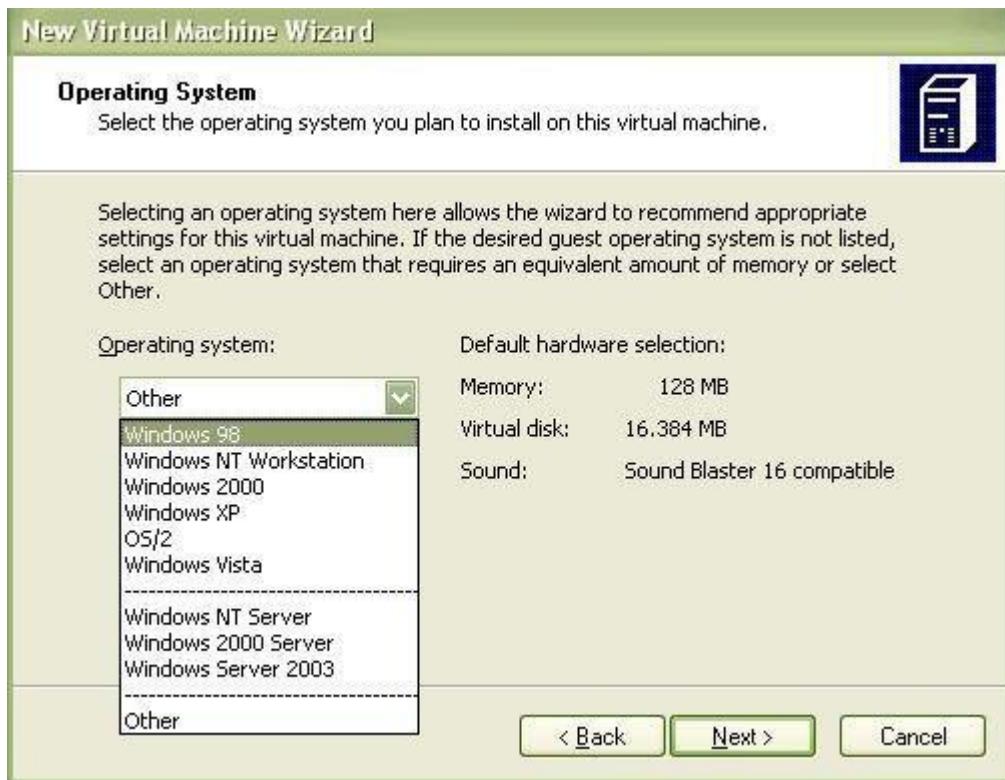
A segunda opção cria apenas uma configuração padrão para uma possível nova máquina virtual.

A terceira opção adiciona uma máquina virtual já existente. Pode ser uma que você tenha copiada no seu pendrive, por exemplo.

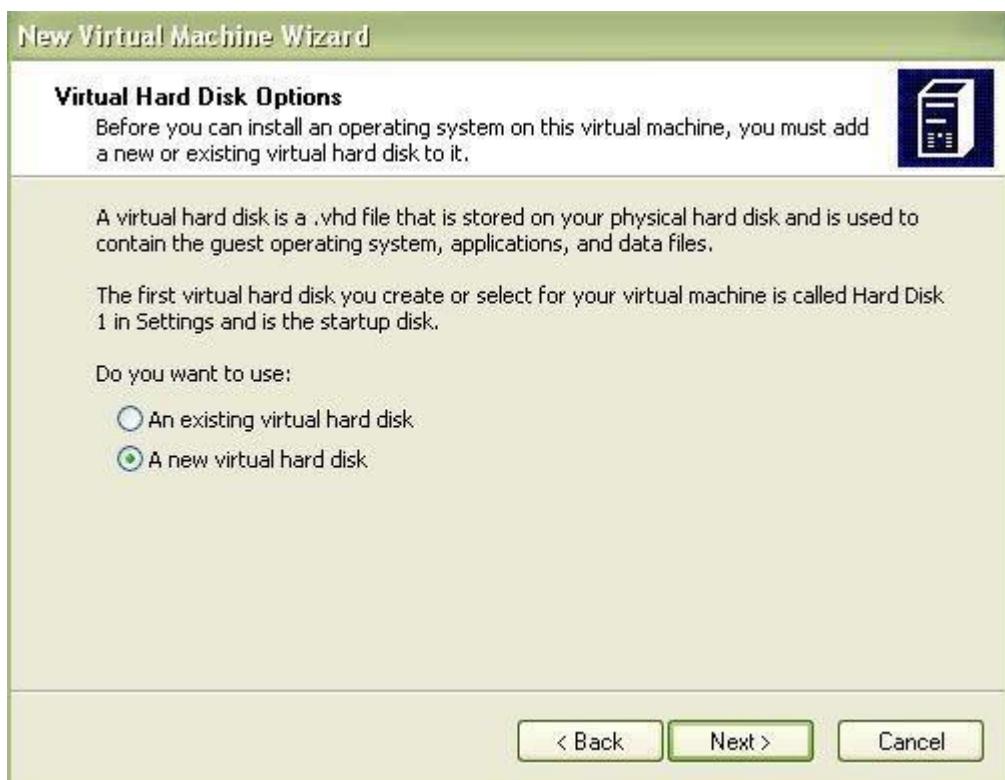
Clique em Next para continuar.



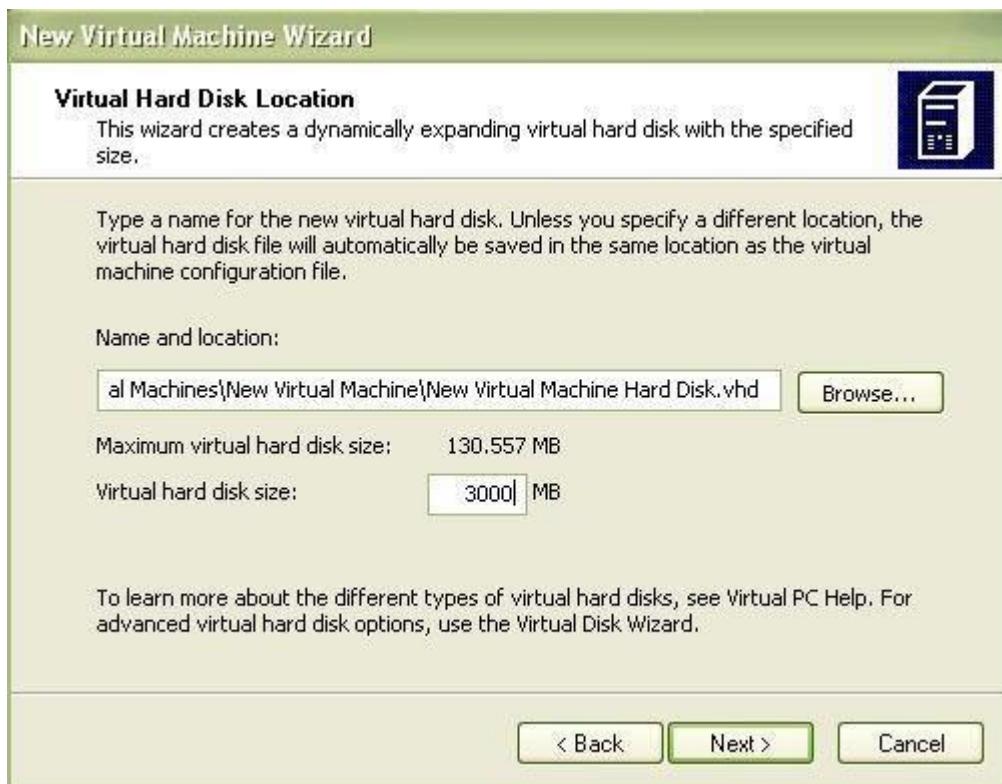
Aqui você colocará o nome da máquina virtual. No exemplo, vamos criar uma máquina OpenSuse 11. Se fosse Windows XP, Red Hat Linux, enfim, é só colocar o nome aqui. Clique em Next:



Uma opção “Operation System” irá aparecer para que você selecione o sistema operacional. Na realidade essa escolha não influenciará muito porque ela apenas determina a quantidade de memória RAM e disco que a máquina virtual terá, algo que você também pode configurar manualmente depois. Clique em next:



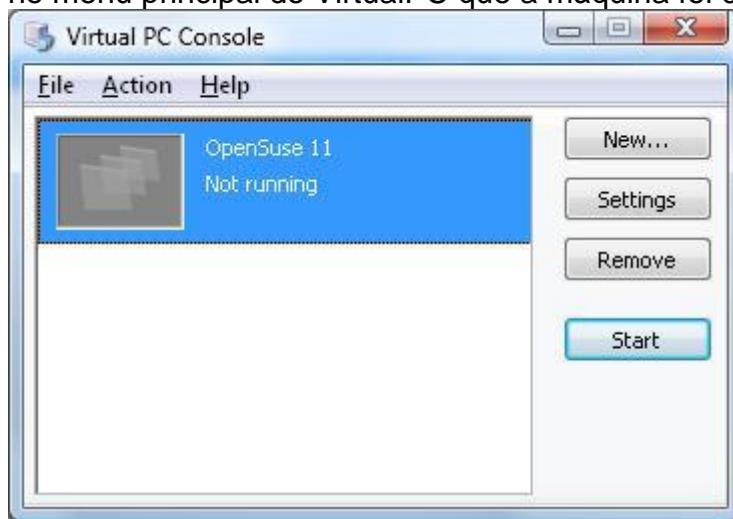
Agora você tem duas opções. A primeira, é para utilizar um disco virtual já existente. Mas quando você cria uma máquina nova, vai querer criar um novo disco virtual. Use a opção “A new virtual hard disk”. Clique em Next:



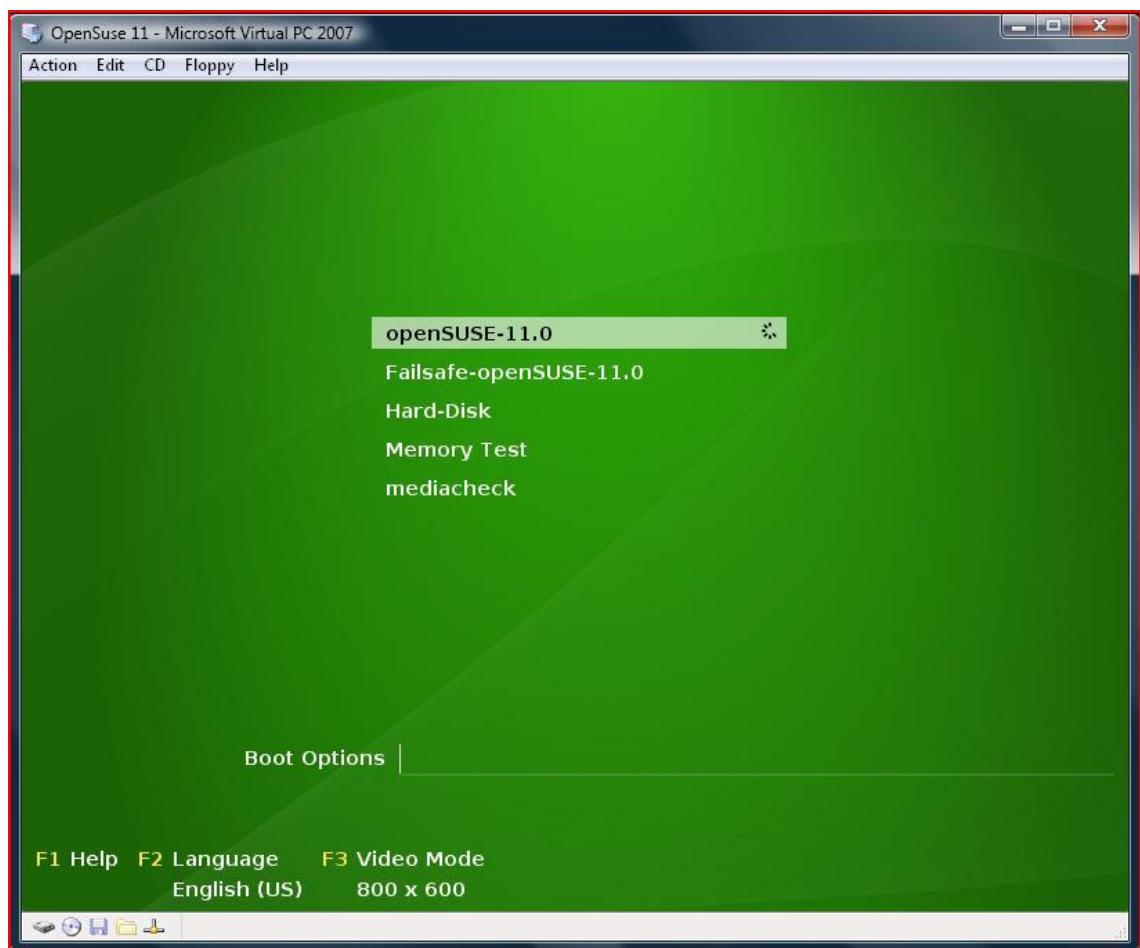
Aqui você irá definir onde o disco virtual será criado, assim como o tamanho total que ele terá (em Megabytes). Clique em next:



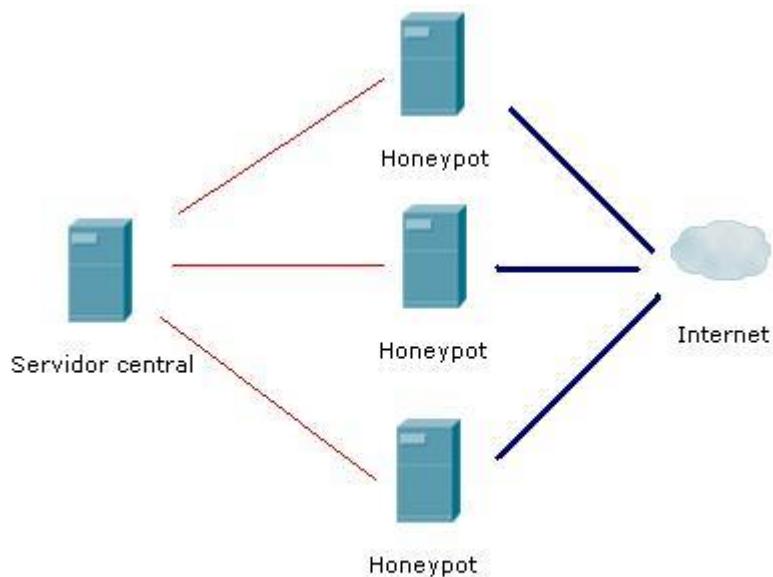
Prontinho. A criação da sua máquina virtual já está completa. Observe no menu principal do VirtualPC que a máquina foi criada com sucesso.



O processo a partir de agora é bem simples, praticamente idêntico ao que você faria na máquina real. O que você fez aqui foi criar o computador virtual, ainda não instalou o sistema operacional. A máquina virtual está completamente vazia. Coloque o CD de instalação do SO no drive (no caso deste exemplo, o OpenSuse 11) e clique no botão “Start” para inicializar a máquina virtual. A instalação irá ocorrer normalmente como seria no computador real. Veja:



Centralização de logs



Centralização dos Logs dos Honeypots em um servidor

Uma das importantes tarefas em uma honeynet é manter os logs guardados com segurança para evitar problemas com possíveis brechas e ataques. Imagine se um invasor consegue acesso a seu gateway, apaga os logs e você não tem um backup? Assim como poderia ocorrer em todas as máquinas. É interessante então investir em um servidor central de logs, que receberia os dados de todas as máquinas (no caso da utilização de honeypots do tipo HIDS, baseados em software) e do gateway.

Honeytokens



Os honeytokens são essenciais para ser utilizados em honeypots comuns ou honeynets, sejam elas reais ou virtuais. Um honeytoken é um trecho de informação que tem um nome ou algum outro atributo que irá atrair a curiosidade de um possível invasor. Você pode criar uma regra no IDS da Honeynet para monitorar todos os acessos aos honeytokens que foram criados.

Difícil de imaginar o que seria isso? Imagine que um atacante encontre os seguintes arquivos dentro de um computador honeypot:



São quatro itens de nomes muito interessantes:

Senhas.txt seria o primeiro óbvio alvo. Muitas pessoas ainda tem o péssimo hábito de gravar as suas senhas em arquivos de texto ou documentos do Word. Esse definitivamente faria o invasor dar pelo menos uma “abridinha”.

Relação de salários.xls também desperta curiosidade pois o atacante pode imaginar que vai conhecer mais sobre os funcionários e pessoas que trabalham na rede que ele está invadindo. Normalmente em planilhas deste tipo estão o nome e o cargo do funcionário.

Escuta oculta.wav faria o invasor pensar que a pessoa que teve o seu computador comprometido por ele estava realizando algum tipo de investigação. Seria contra seu chefe? Seus colegas? Contra sua esposa que ele acreditava ser infiel? É um clássico em atiçar a curiosidade.

Fotos privadas é outro caso que seria acessado com certeza. Afinal, qualquer atacante tem a curiosidade de ver a cara daqueles que tiveram seu sistema invadido pela sua pessoa. Só não vai saber que a loira de olhos azuis que possui foto ali é uma holandesa capturada no Google.

Não pense que os honeytokens são apenas arquivos, eles podem ser também uma conta de usuário com o nome provocativo, como por exemplo: **administrador, diretor, presidente, etc.**

Honeywall

O principal gateway para o controle do tráfego dentro da honeynet (especialmente a Gen II), é chamado de *Honeywall*. Ele mistura um firewall filtro de pacotes com uma ferramenta de detecção de intrusos (NIDS). Sua parte firewall atua como qualquer outro, sendo responsável por criar as regras principais para a entrada e a saída do tráfego malicioso, gravando tudo que você julgar necessário nos logs. Vendo a grande importância disto, irei abordar nessa seção alguns conceitos como os tipos de firewalls mais comuns assim como fornecer algumas dicas de configuração básica do Firewall iptables.

Tipos de Firewalls

Existem basicamente três tipos de firewalls:

Firewall Filtro de Pacotes – É o tipo de firewall mais comum. Ele age na camada de rede do modelo OSI, filtrando os pacotes que entram e saem da rede, baseando-se nas listas de controle de acesso (ACLs).

Firewall Proxy – É utilizado para fornecer acesso filtrado à internet com maior segurança para as máquinas de uma rede interna, fazendo-as passar por um único ponto. Muitas empresas empregam firewalls do tipo Proxy pois isso é útil para realizar um maior controle de segurança, bloqueando o acesso a determinados websites, etc.

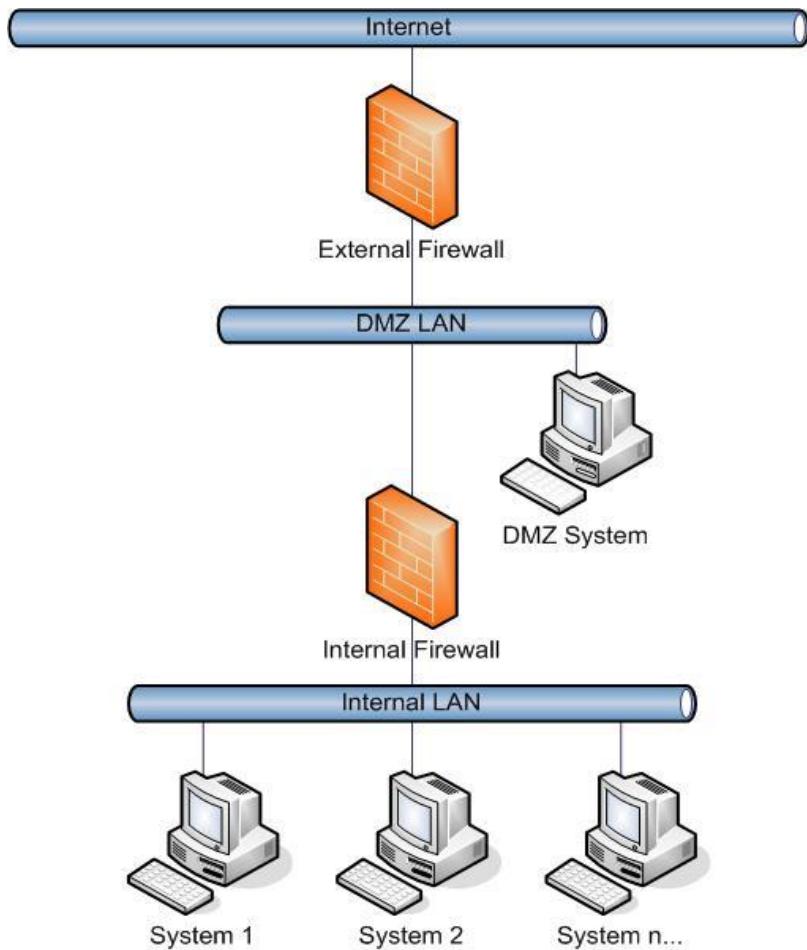
Firewall NAT – É um firewall que normalmente já é filtro de pacotes, mas que também possui recursos de Network Address Translator, ou NAT. Isso permite que computadores de uma rede interna consigam receber e trocar informações com sistemas de fora dessa rede.

Alguns exemplos de firewalls: ISA, IPTables, Winroute.

Zona Desmilitarizada

Uma prática bem comum também é utilizar mais um ponto de firewall em uma rede, criando o que chamamos de DMZ ou Zona Demilitarizada. Pode-se colocar um firewall no primeiro ponto de entrada da rede, posicionando ali servidores de menor risco, como o servidor WEB. Logo depois, então, teria outro firewall, bloqueando melhor os acessos à rede interna e a servidores mais importantes, como o de banco de dados. Isso é bom, pois cria uma camada extra de proteção.

Veja, na imagem a seguir, um exemplo conceitual de DMZ.



Instalando o IPTables

O iptables é um firewall muito poderoso e muito utilizado em sistemas Unix (especialmente Linux) . Para instalar em distribuições baseadas em Red Hat, utilize "urpmi iptables". Se for uma distribuição baseada em Debian, utilize "apt-get install iptables". Verifique o resultado da instalação na imagem a seguir.

```
busybin:~# iptables -v
-bash: iptables: command not found
busybin:~# apt-get install iptables
Reading Package Lists... Done
Building Dependency Tree... Done
Suggested packages:
  ipmasq iproute
The following NEW packages will be installed:
  iptables
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0B/381kB of archives.
After unpacking 1270kB of additional disk space will be used.
Selecting previously deselected package iptables.
(Reading database ... 51072 files and directories currently installed.)
Unpacking iptables (from .../iptables_1.2.11-10_i386.deb) ...
Setting up iptables (1.2.11-10) ...

busybin:~# █
```

Toda a configuração do Iptables é realizada em modo Shell, onde você deve inserir as regras uma de cada vez. É importante salvar as regras senão elas não serão inicializadas com o micro. Você pode utilizar o comando iptables-save para salvar as regras para um arquivo , e iptables-restore para restaurar essas regras..

Mesmo antes de começar a utilizar o firewall é importante definirmos algumas coisas.

Por exemplo, quais serviços e portas exatas devemos proteger? Pense que você está montando uma honeynet. Se fechar demais o firewall, todo o seu investimento será em vão pois nenhum invasor conseguirá acesso.

É interessante bloquear a todas portas (TCP e UDP) abaixo da 1024, com exceção apenas dos serviços que você quer permitir na honeynet (SMTP – 25, FTP – 21, etc).

Regras do iptables

As regras são como comandos passados ao iptables para que ele realize uma determinada ação (como bloquear ou deixar passar um pacote) de acordo com o endereço/porta de origem/destino, interface de origem/destino, etc. As regras são armazenadas dentro dos chains e processadas na ordem que são inseridas.

As regras são armazenadas no kernel, o que significa que, quando o computador for reiniciado, tudo o que fez será perdido. Por este motivo elas deverão ser gravadas em um arquivo para serem carregadas a cada inicialização. Um exemplo de regra: iptables -A INPUT -s 123.123.123.1 -j Drop.

Chains

```

root@serv1:/etc/sysconfig
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6902
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6903
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6904
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6905
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6906
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6907
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6908
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6909
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6910
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6911
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6912
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6913
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6914
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6915
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6916
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6917
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6918
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6919
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:6920
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:2855
ACCEPT  tcp  --  anywhere             anywhere             tcp dpt:15286

Chain FORWARD (policy ACCEPT)
target     prot opt source          destination
LOG       all  --  anywhere        anywhere            LOG level debug prefix 'BANDWIDTH_OUT:'
LOG       all  --  anywhere        anywhere            LOG level debug prefix 'BANDWIDTH_IN:'
LOG       tcp  --  anywhere        anywhere            tcp flags:!SYN,RST,ACK/SYN state NEW LOG level info prefix 'FIREWALL: NEW sem syn: '
DROP      tcp  --  anywhere        anywhere            tcp flags:!SYN,RST,ACK/SYN state NEW
ACCEPT    all  --  anywhere        anywhere            state NEW,RELATED,ESTABLISHED
REJECT   tcp  --  anywhere        anywhere            tcp dpt:135 reject-with icmp-port-unreachable
ACCEPT    tcp  --  anywhere        anywhere            tcp flags:SYN,RST,ACK/SYN limit: avg 2/sec burst 5
ACCEPT    icmp --  anywhere       anywhere            icmp echo-request limit: avg 1/sec burst 5

Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
LOG       all  --  anywhere        anywhere            LOG level debug prefix 'BANDWIDTH_OUT:'
ACCEPT   all  --  anywhere        anywhere            state NEW,RELATED,ESTABLISHED
[root@serv1 sysconfig]#

```

Chains são locais onde as regras do firewall definidas pelo usuário são armazenadas para operação do firewall. Existem dois tipos : os embutidos (como os chains INPUT, OUTPUT E FORWARD) e os criados pelo usuário. Os nomes dos chains embutidos devem ser especificados sempre em maiúsculas.

Tabelas

Tabelas são os locais usados para armazenar os chains e conjunto de regras com uma determinada característica em comum. As tabelas podem ser referenciadas com a opção -t tabela e existem três tabelas disponíveis no iptables:

Filter

A filter é a tabela padrão do Iptables. Ela contém três padrões:

OUTPUT - Tráfego que sai da máquina.

INPUT - Tráfego que chega à máquina.

FORWARD – Tráfego redirecionado para outra interface de rede ou outra máquina (roteado).

O INPUT e OUTPUT se tratam somente de conexões originadas/destinadas ao host local..

Mangle

Utilizado para modificar o tráfego com base em características especiais, como por exemplo o tipo de serviço utilizado (TOS). A Mangle possui cinco chains:

INPUT - Utilizado quando os pacotes precisam ser modificados antes de serem enviados para o chain INPUT da tabela filter.

OUTPUT - Utilizado quando o tráfego precisa ser modificado antes de ser enviado para o chain OUTPUT da tabela nat.

FORWARD - Utilizado quando o tráfego precisa ser modificado antes de ser enviado para o chain FORWARD da tabela filter.

PREROUTING - Consultado quando os pacotes precisam ser modificados antes de ser enviados para o chain PREROUTING da tabela nat.

POSTROUTING - Utilizado quando o tráfego precisa ser modificado antes de ser enviado para o chain.

Nat

Utilizado para criação de regras de NAT (Network Address Translator), permitindo a tradução dinâmica de endereços. Possui três padrões:

OUTPUT - Utilizado quando o tráfego gerado localmente precisa ser modificado antes de ser roteado. É importante notar que essa chain somente é consultada para conexões que se originam de IPs locais.

PREROUTING - Utilizado quando o tráfego precisa ser modificado logo que chega. É o chain perfeito para realização de redirecionamento de portas e DNAT.

POSTROUTING - Utilizado quando o tráfego precisar ser modificado após o tratamento de roteamento. É o ideal para realização de SNAT e IP Masquerading.

Utilização de Chains

O iptables trabalha com uma tabela de regras que é analisada uma a uma até que a última seja processada. Por padrão, se uma regra tiver qualquer erro, uma mensagem será mostrada e ela descartada. O pacote não conferirá e a ação final (se ele vai ser aceito ou rejeitado) dependerá das regras seguintes.

As opções passadas ao iptables usadas para manipular os chains são sempre em maiúsculas. As seguintes operações podem ser realizadas:

Adicionando regras – A

Para início, deve-se uma regra que impede o acesso à máquina local (127.0.0.1). Vamos incluir uma regra no chain INPUT (-A INPUT) que bloqueia (-j DROP) qualquer acesso ao endereço de loopback 127.0.0.1 (-d 127.0.0.1):

```
iptables -t filter -A INPUT -d 127.0.0.1 -j DROP
```

A opção -A é utilizada para adicionar novas regras no final do chain. A opção –t especifica a tabela filter (omitindo essa opção define essa tabela por padrão). Ao invés de -j DROP que descarta os pacotes, pode-se usar -j ACCEPT para aceitar pacotes. A opção -j é chamada de alvo da regra ou somente alvo, pois define o destino do pacote que atravessa a regra.

Apagando uma regra – D

Existem duas formas para se apagar um chain. A primeira é referenciando o número diretamente no comando:

```
iptables -t filter -D INPUT 1
```

Esta opção não é boa quando temos um firewall complexo com um grande número de regras por chains. Aqui, a segunda opção é a melhor:

```
iptables -t filter -D INPUT -d 127.0.0.1 -j DROP
```

Então, a regra correspondente no chain INPUT será automaticamente apagada (confira listando o chain com a opção "-L"). Caso o chain possua várias regras semelhantes, somente a primeira será apagada. Não é possível apagar os chains defaults do iptables (INPUT, OUTPUT...).

Inserindo regras – I

Se quiser que pacotes que venham do endereço 192.168.0.100 consigam entrar no firewall, devemos inserir a nova regra antes da regra que bloqueia todo o tráfego ao endereço 127.0.0.1 na primeira posição:

```
iptables -t filter -I INPUT 1 -s 192.168.0.100 -d 127.0.0.1 -j ACCEPT
```

Após este comando, temos a regra inserida na primeira posição do chain e a antiga regra número 1 passa a ser a segunda. Portanto, a primeira será consultada liberando o tráfego de entrada do IP 192.168.0.100.

Listando regras – L

A sintaxe utilizada é:

```
iptables [-t tabela] -L [chain] [opções]
```

tabela : é uma das tabelas usadas pelo iptables. Se a tabela não for especificada, a tabela filter será usada como padrão.

chain: um dos chains na tabela . Caso o chain não seja especificado, todos os chains da tabela serão mostrados.

Para listar a regra criada anteriormente, usamos o comando:

```
#iptables -t filter -L INPUT
```

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP      all  --  anywhere             localhost
```

Inserindo regras – I

Para que um tráfego proveniente de 192.168.0.100 consiga entrar no firewall, temos que inserir a nova regra antes da regra que bloqueia todo o tráfego ao endereço 127.0.0.1 na primeira posição:

```
iptables -t filter -I INPUT 1 -s 192.168.0.100 -d 127.0.0.1 -j ACCEPT
```

Após este comando, temos a regra inserida na primeira posição do chain e a antiga regra número 1 passa a ser a segunda. Portanto, a primeira será consultada liberando o tráfego de entrada do IP 192.168.0.100.

Substituindo regras – R

Se eu quiser alterar o conteúdo de uma regra qualquer, tenho duas alternativas: apagar a regra e inserir uma nova no lugar ou modificar diretamente a regra já criada sem afetar outras regras existentes e mantendo a sua ordem no chain (isso é muito importante). Use o seguinte comando:

```
iptables -R INPUT 2 -d 127.0.0.1 -p icmp -j DROP
```

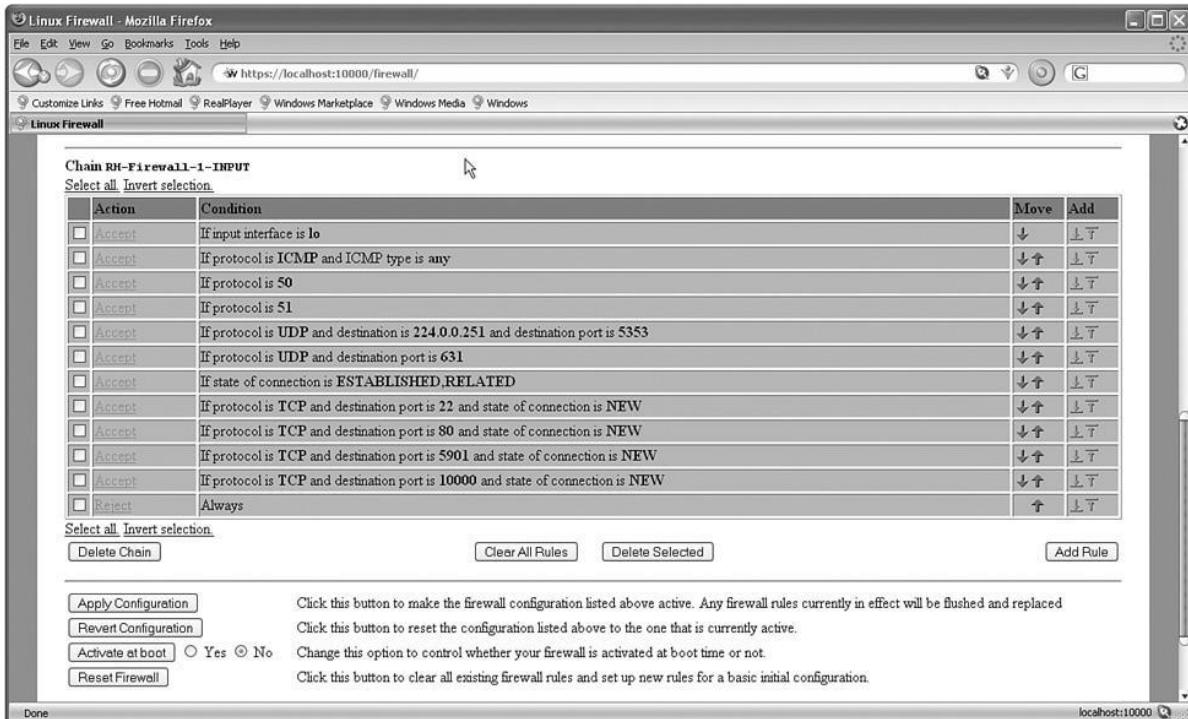
O número 2 é o número da regra que será substituída no chain INPUT e deve ser especificado. O comando acima substituirá a regra 2 do chain INPUT (-R INPUT 2) bloqueando (-j DROP) qualquer pacote icmp (-p icmp) com o destino 127.0.0.1 (-d 127.0.0.1).

Criando novos chains

Se o firewall que você utiliza tiver um grande número de regras (o que normalmente não irá acontecer em relação à honeynet, a não ser que você queira simular um ambiente complexo), é interessante criar chains individuais (com a opção **-N**) para organizar regras de um mesmo tipo ou que tenha por objetivo analisar um tráfego de uma mesma categoria (interface, endereço de origem, destino, protocolo, etc.), pois podem consumir muitas linhas e tornar o gerenciamento do firewall confuso e, consequentemente, causar sérios riscos de segurança. O tamanho máximo de um nome de chain é de 31 caracteres e podem conter tanto letras maiúsculas quanto minúsculas.

```
iptables [-t tabela] [-N novochain]
```

Na figura, a seguir, está um chain criado através da interface gráfica webmin para o Iptables. Veja que existem várias regras criadas no chain. Se você não tiver muita experiência em modo Shell, talvez utilizar um programa gráfico como o webmin poderá acelerar a definição das regras para a sua honeynet.



Para criar o chain internet, que pode ser usado para agrupar as regras de internet, usamos o seguinte comando: **iptables -t filter -N internet**

Para inserir regras no chain internet basta especifica-lo após a opção -A:

iptables -t filter -A internet -s 201.100.100.100 -j DROP

E, então, criamos um pulo (-j) do chain INPUT para o chain internet:

iptables -t filter -A INPUT -j internet

Definindo um Alvo

O alvo (-j) é o destino que um pacote terá quando a regra for corretamente conferida. Esse alvo pode indicar para bloquear a passagem do pacote (-j DROP), aceitar a passagem do pacote (-j ACCEPT) ou rejeitar o pacote (-j REJECT). Existem mais alvos como, por exemplo, redirecionar um pacote com -j REDIRECT. Porém, esses quatro são os principais. Vejamos um pouco sobre cada um deles.

ACCEPT

O pacote é ACEITO e procede normalmente para dentro/fora da rede. É o padrão utilizado nas principais chains.

REJECT

Este é um módulo que faz a mesma função do alvo DROP. A diferença é que uma mensagem ICMP é retornada para a máquina de origem, informando que a máquina rejeitou o pacote.

DROP

Bloqueia o pacote, e o processamento das regras daquele chain é concluído. Pode ser usado como alvo em todos os chains de todas as tabelas do iptables e também pode ser especificado no policiamento padrão das regras do firewall.

Essa é apenas uma pequena visão da grande gama de configurações que o iptables possui. Mas creio que você já tenha uma idéia básica de como criar e modificar regras de forma a conseguir filtrar todo o tráfego que desejar na sua honeynet. Caso deseje aprofundar-se mais nesse assunto, recomendo um livro sobre IPTABLES ou ISA Server. Focar demais em firewalls não é o objetivo desse material.

Utilizando o NIDS Snort

O Snort é um NIDS de fácil utilização, que pode ser facilmente colocado no seu computador honeywall, junto ao firewall IPTABLES, visto anteriormente. Vou demonstrar como instalar e utilizá-lo nos sistemas Linux e Windows.

Instalação do Snort no Linux

A maneira mais simples de se instalar o snort é através do comando de utilização repositórios apt-get ou urpmi:

apt-get install snort

Para confirmar se a instalação foi executada com sucesso, digite:

snort -v e veja se você obtém a resposta do programa, como mostrado a seguir:

```
Running in packet dump mode
Initializing Network Interface eth0
```

```
--Initializing Snort ==
Initializing Output Plugins!
Decoding Ethernet on interface eth0
== Initialization Complete ==,,_
```

```
Snort! Version 2.3.0 (Build 10)
By Martin Roesch & The Snort Team: http://www.snort.org/team.html
(C) Copyright 19982004
Sourcefire Inc., et al.
02/2508:
```

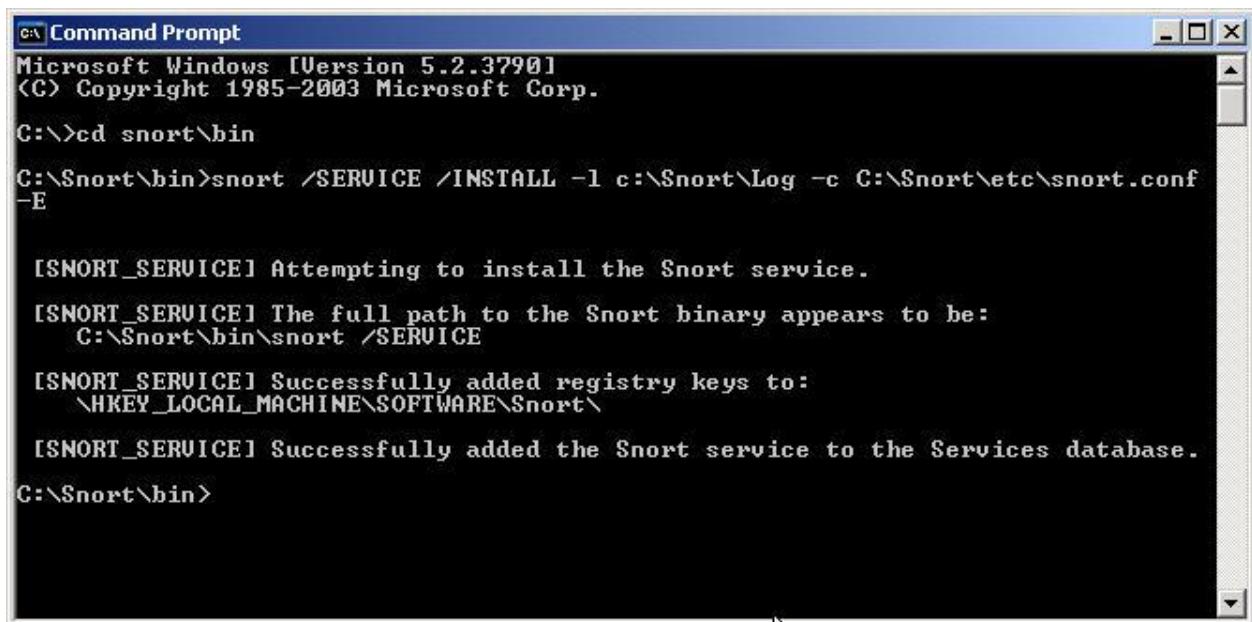
```
39:10.293573 xxx.xxx.xxx.xxx:1111 >
xxx.xxx.xxx:1111
UDP TTL:64 TOS:0x0 ID:1876 IpLen:20 DgmLen:178 DF
Len: 150
=====+
=====
Snort received 2 packets
Analyzed: 2(100.000%)
Dropped: 0(0.000%)
=====
Breakdown by protocol:
TCP: 0 (0.000%)
UDP: 1 (100.000%)
ICMP: 0 (0.000%)
ARP: 0 (0.000%)
EAPOL: 0 (0.000%)
IPv6: 0 (0.000%)
IPX: 0 (0.000%)
OTHER: 0 (0.000%)
DISCARD: 0 (0.000%)
=====
Action Stats:
ALERTS: 0
LOGGED: 0
PASSED: 0
=====
Snort exiting
```

Instalação do Snort

O Snort pode ser instalando compilando-se os fontes ou utilizando algum gerenciador de pacotes (yum apt-get ou urpmi). Consulte a documentação do sistema operacional que você está usando. Pode-se instalar o Snort em diversos sistemas, como Windows, Linux e FreeBSD. Acesse o site www.snort.org para saber mais.

Como exemplo, se baixar a versão para Windows pode instalar o Snort como um serviço do sistema, indicando o diretório de logs e o arquivo de configuração.

```
Snort /SERVICE /INSTALL -l c:\Snort\Log -c C:\Snort\etc\snort.conf
```



The screenshot shows a Microsoft Windows Command Prompt window with the title 'Command Prompt'. The window displays the following text:

```
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

C:\>cd snort\bin
C:\Snort\bin>snort /SERVICE /INSTALL -l c:\Snort\Log -c C:\Snort\etc\snort.conf
-E

[SNORT_SERVICE] Attempting to install the Snort service.
[SNORT_SERVICE] The full path to the Snort binary appears to be:
  C:\Snort\bin\snort /SERVICE
[SNORT_SERVICE] Successfully added registry keys to:
  \HKEY_LOCAL_MACHINE\SOFTWARE\Snort\
[SNORT_SERVICE] Successfully added the Snort service to the Services database.

C:\Snort\bin>
```

Após, o serviço foi registrado com sucesso e será inicializado junto com o Windows. Se você quiser executar o Snort diretamente como uma aplicação, basta rodar o executável. Digite a palavra “snort” no prompt de comandos para trazer o menu do programa com todas as opções disponíveis. Veja na próxima imagem.

Lembrando que o Snort, assim como o IPTables, possui vários programas que podem ser utilizados como interface gráfica (GUI), facilitando a administração do software. Mais adiante, você poderá ver o exemplo do IDS Center que é uma interface para Windows. Mas, no caso do Linux, também existem diversas outras que podem ser conferidas na página do Snort: www.snort.org

Configuração do Snort

Snort.conf é o arquivo de configuração do Snort. A localização do mesmo pode diferenciar dependendo do sistema operacional utilizado. Normalmente em sistemas *nix ele está no diretório /etc/snort. Esse arquivo contém as regras e

ações a serem tomadas para cada pacote recolhido e confrontado com ele. O resultado será gerado no diretório /var/log/snort, ou outro diretório previamente estipulado. O arquivo snort.conf deve estar presente no diretório corrente ou ser digitado o diretório onde ele se encontra. Abaixo, você vê as configurações necessárias mais comuns:

Alguns exemplos básicos de comandos do Snort

Snort -b

Captura todos os tipos de pacotes

Snort -b -h 192.168.10.0/24

Captura todos os tipos de pacotes apenas para o endereço de rede 192.168.10.0, classe C

snort -dev -l /logscapturados/logs.txt

Primeiro, o diretório deve ser criado (logscapturados ou outro nome). Após criá-lo e executar o comando, o snort irá gerar um arquivo chamado logs.txt de todos os pacotes capturados

Snort -b -l /logscapturados/logs.txt

Gravará todos os dados capturados (sem exceção) no arquivo de log especificado

snort -vd

Mostra apenas cabeçalhos dos protocolos.

snort -vde

Mostra cabeçalhos e dados contidos neles também.

snort -dev -l ./logscapturados -h 192.168.10.0/24

Nesse caso, o snort irá capturar todos os dados (cabeçalhos e dados) relacionados à rede 192.168.10.0 (Classe C) e irá gravá-los no diretório meuslogs, com cada endereço IP tendo seu próprio arquivo.

Serviços

Uma rede real deve ter serviços rodando em seus computadores honeypot, senão o invasor acreditará que está entrando em uma “rede fantasma”. É importante utilizar de diversos serviços reais. Cada serviço tem a sua importância, mas três em particular eu penso serem essenciais de se possuir : DHCP, DNS e VPN. Vamos dar uma olhada em cada um deles.

DHCP

É um serviço (protocolo) que possibilita computadores clientes receberem endereços IPs de forma dinâmica. Ele trabalha com as portas 67 (escuta) e 68 (transmissão). Você poderia muito bem configurar todas as máquinas da sua honeynet de forma fixa, mas sinceramente, acho inviável. Até porque, dependendo do software de honeypot que você for utilizar, ele vai requerer obrigatoriamente um endereço IP obtido de forma automática. Outro fator: hoje é quase impossível encontrar uma rede que não utilize dhcp para endereçamento automático. Só se for realmente uma rede muito pequena. Caso não houver esse serviço na honeynet, além de lhe dar mais trabalho para configurar manualmente os endereços, pode levar aos invasores a desconfiarem de alguma coisa.

Vou exemplificar brevemente a instalação do servidor DHCPD em sistemas Linux:

Você pode instalar o DHCPD baixando os fontes e as bibliotecas ou utilizando alguma das ferramentas de repositório, como o apt-get, yum ou urpmi. Verifique qual dos dois a sua distribuição utiliza. No caso de sistemas CentOS, por exemplo, você utilizaria:

```
yum install dhcpcd
```

Após a instalação, será criado o arquivo `/etc/dhcpd.conf`. Mande abrir o arquivo utilizando o editor de textos VI.

```
ddns-update-style none;
subnet 10.2.80.0 netmask 255.255.255.0 {
    # default gateway
    option routers 10.2.80.80;
    option subnet-mask 255.255.255.0;

    option domain-name "tec80.net";

    # Setting up an ip address is better here
    option domain-name-servers srv80.tec80.net;
    option nis-domain "domain.org";

    range 10.2.80.100 10.2.80.150;
    default-lease-time 21600;
    max-lease-time 43200;
```

Vamos entender as configurações mais importantes do arquivo:

Subnet 10.2.80.0 netmask 255.255.255.0

Define a utilização de uma subrede (**subnet**) , classe C (255.255.255.0) utilizando o endereço 10.2.80.0 (que como é classe C, irá de 10.2.80.0 a 10.2.80.255, obviamente excluindo o primeiro e último IP para fins de endereçamento da rede e broadcast).

option routers 10.2.80.80

Define o gateway padrão que será configurado nas máquinas. Pode ser o roteador (caso seu honeypot seja baseado em solução de software), ou se for uma honeynet completa, este será o endereço do computador atuando como Honeywall.

option subnet-mask 255.255.255.0

Define a máscara de rede a ser utilizada. Nesse caso não fez muita diferença pois já a havíamos definido na diretiva subnet.

option domain-name “tec80.net”

Todos os computadores que requisitarem configuração via dhcp terão o seu domínio configurado como tec80.net.

range 10.2.80.100 10.2.80.150

Essa opção é importante. Ela está definindo quais o intervalo de endereços IPs (dentro da faixa que especificamos em subnet) que serão distribuídos para os clientes na hora da realização dos pedidos.

default-lease-time 21600

Esse é o tempo padrão de duração da concessão do endereço IP pelo DHCP. O tempo é expresso em segundos.

max-lease-time 43200

Tempo máximo permitido para a concessão do endereço IP pelo DHCP. O tempo é expresso em segundos.

Existem diversas outras opções mais profundas, como a utilização de group para definir grupos e host para definir configurações individuais de IP (como para fornecer endereço ip fixo). Consulte o manual do dhcpd para saber mais digitando **man dhcpd**.

Para iniciar o serviço (ou reiniciá-lo), digite: **service dhcpd restart**

DNS

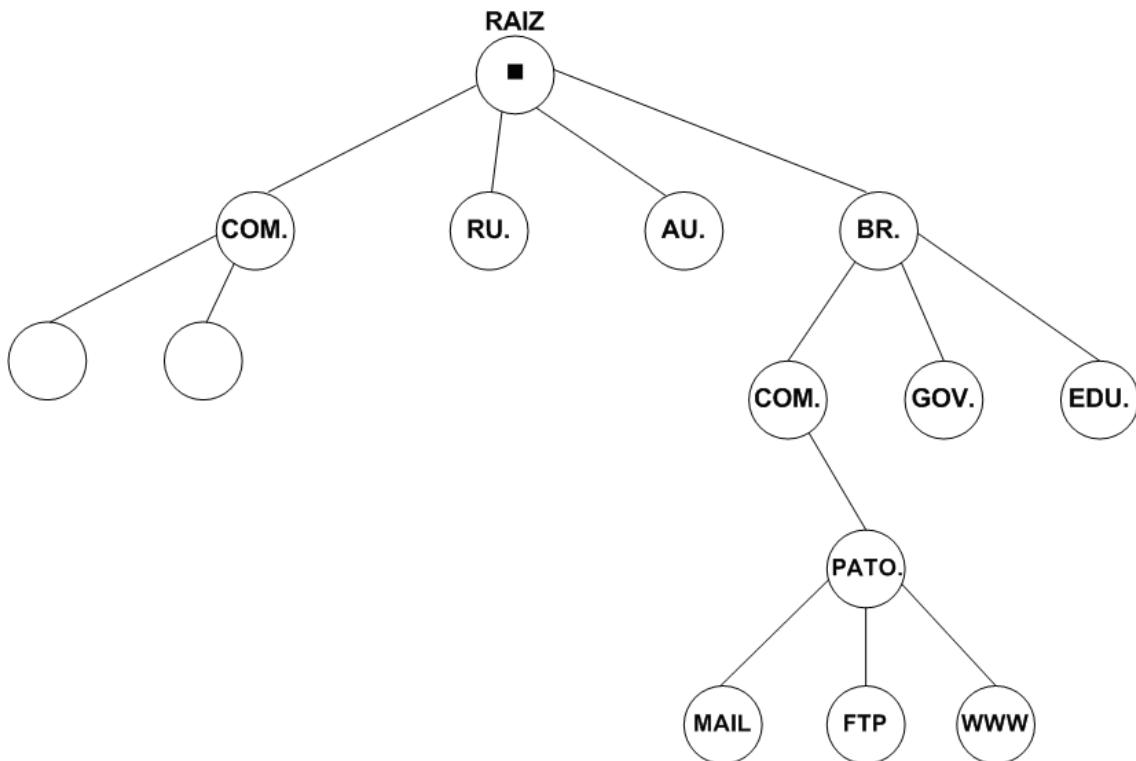
O DNS é um serviço essencial para qualquer rede que se conecte à internet hoje. E será essencial no nosso caso. Se você adquirir um domínio, como por exemplo “bancointernational.com” e quiser utilizá-lo em sua honeynet para simular por exemplo uma rede de instituição bancária com maior fidelidade, vai necessitar de um servidor de nomes bem configurado.

É através do DNS que consegue-se realizar a resolução de nomes para endereços IPs e vice-versa. Você pode pensar nele como uma espécie de banco de dados hierárquico, que se inicia nos servidores raiz (os principais servidores de resolução de nomes na internet) . Os principais computadores que atual como a raiz do DNS, entre outros, são:

Letra	Operador	Localização
A	VeriSign	Dulles, Virginia, USA
B	ISI	Marina Del Rey, California, USA
C	Cogent	(Distribuído, usando anycast)
D	University of Maryland	College Park, Maryland, USA
E	NASA	Mountain View, California, USA
F	ISC	(Distribuído, usando anycast)
G	U.S. DoD NIC	Columbus, Ohio, USA
H	U.S. Army Research Lab	Aberdeen Proving Ground, Maryland, USA
I	Autonomica	(Distribuído, usando anycast)

J	VeriSign	(Distribuído, usando anycast)
K	RIPE NCC	(Distribuído, usando anycast)
L	ICANN	Los Angeles, California, USA
M	WIDE Project	Tokyo, Japan

O nome de um computador completo na Internet, incluindo sem caminho desde o servidor raiz, é chamado de FQDN (Fully Qualified Domain Name).



Exemplo: o servidor de correio da zona “pato”, refere-se ao FQDN: mail.pato.com.br

No DNS, cada parte dessa hierarquia é considerada uma “zona”. No exemplo anterior, existe então a zona raiz, a zona BR, a zona COM e a zona PATO. Os servidores MAIL e FTP e WWW estão sobre o domínio desta última zona. Isso significa que é ela que irá informar qual o endereço IP desses computadores quando receber uma consulta.

Quando se trata do DNS, há dois tipos de consulta que podem ser realizados: a consulta recursiva e a consulta iterativa.

A consulta recursiva é aquela que é realizada entre o computador cliente e o DNS quando é requisitado um endereço IP.

A consulta iterativa é aquela que é realizada entre servidores DNS para se obter a resposta para a consulta recursiva.

Existem também três tipos de servidores DNS: Primário, Secundário e Caching-Only

O servidor DNS primário (ou master) é o responsável principal pela zona. É ele que responderá diretamente as consultas realizadas e também irá sofrer constantes alterações em seus registros para suprir as mudanças na rede (novo computador ou dispositivo adicionado, endereço IP de máquina mudado, etc).

O servidor DNS secundário (slave) age como uma espécie de backup. Ele receberá a zona do servidor primário através de um processo chamado de transferência de zona. Assim, se o servidor primário deixar de funcionar, o secundário assumirá imediatamente o seu lugar, respondendo às requisições dos clientes.

Já o servidor Caching-Only, apenas armazenará consultas já realizadas para tentar acelerar o processo de resolução de endereços para domínios que são visitados com determinada freqüência.

Registro de Recursos

Os RR (registro de recursos) são as informações que um servidor DNS possui em sua zona. Pense neles como as entradas de um banco de dados. É através de cada um desses registros que você consegue realizar o mapeamento de nomes para IPs, IPs para nomes, definir a zona, definir apelidos para as máquinas. Enfim, são os tipos de registros que podem ser utilizados. Aqui estão alguns deles:

Tipo de Registro	Descrição	Uso
SOA	Registro tipo Start of Authority	Especifica atributos referente à zona, como nome do domínio, contato administrativo, número de série da zona, TTL, etc.
PTR	Registro tipo ponteiro	Mapeamento de endereço IP em FQDN.
NS	Registro tipo name server	Define o nome do servidor da zona
A	Registro tipo endereço	Mapeamento de FQDN em endereço IP.
CNAME	Registro tipo nome canônico	Nome de domínio canônico para um alias (nome alternativo).
MX	Registro tipo Mail Exchange	Nome do servidor de correio eletrônico para o domínio.

Criando zonas com o servidor de nomes BIND

Como um exemplo prático para você possuir em sua honeynet, um dos servidores DNS mais utilizados é o BIND. É um processo muito tranquilo a sua instalação, e não requer maiores complicações. É um serviço que instalarei em uma máquina Red hat Linux, nesse exemplo. Mas poderia ser em qualquer distribuição que você desejar.

Primeiramente, você pode instalar o bind utilizando alguma das ferramentas de repositório, como o apt-get, yum ou urpmi. Verifique qual dos dois a sua distribuição utiliza. No caso de sistemas CentOS por exemplo, você utilizaria:

yum install bind

```
[root@centos ~]# yum install bind
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package bind.i386 30:9.3.6-4.P1.el5 set
--> Finished Dependency Resolution
```

Após a instalação é criado o arquivo de configuração **/etc/named.conf** onde definiremos o nome das zonas (se não for criado automaticamente, copie os modelos de **/usr/share/doc/bind-x.x.x/sample**. Troque o “x” pelo número da versão). E também será criado o diretório **/var/named** (sem o uso de chroot).

Você pode iniciar o serviço digitando **service named start**

Abra o arquivo **/etc/named.conf** :

```

// generated by named-bootconf.pl

options {
    directory "/var/named";
    /*
     * If there is a firewall between you and nameservers you want
     * to talk to, you might need to uncomment the query-source
     * directive below. Previous versions of BIND always asked
     * questions using port 53, but BIND 8.1 uses an unprivileged
     * port by default.
     */
    // query-source address * port 53;
};

// a caching only nameserver config
//
controls {
    inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};
zone "." IN {
    type hint;
    file "named.ca";
};
"named.conf" 38L, 762C

```

1,1 Top

Hora de criar as zonas primárias. Criarei uma zona direta (para mapear nomes a endereços IP) e uma zona reversa (para mapear endereços IP de volta a nomes).

Utilizei o nome “unatec.com.br” (tradução: um domínio qualquer). É necessário especificar o **type** como master (pois é uma zona primária) e na opção **file** deve-se colocar o arquivo da zona onde estarão os registros de recursos.

```

zone "unatec.com.br" {
    type master;
    file "direto";
};

zone "0.168.192.in-addr.arpa" {
    type master;
    file "reverso";
};

```

Esses arquivos “direto” e “reverso” devem ser criado no diretório /var/named. Use o vim para criar cada um dos dois. Usando o arquivo localdomain.zone

como modelo (está dentro do diretório sample originalmente), crio a zona direta para edição.

```
[root@centos ~]# cd /var/named  
[root@centos named]# ls  
data          my.external.zone.db  named.ip6.local  
localdomain.zone  my.internal.zone.db  named.local  
localhost.zone    named.broadcast      named.root  
[root@centos named]# cp localdomain.zone direto  
[root@centos named]# vim direto
```

Esse arquivo conterá a configuração da zona. Todos os registros de recursos que já foram explicados anteriormente estão demonstrados aqui. Detalhe para os valores do registro SOA: a menos que você utilize um servidor de DNS “escravo” (secundário), não precisa se preocupar com eles. Pode manter os padrões.

É aqui que você irá especificar o nome das suas máquinas e o endereço IP correspondente do seu computador honeypot., o que permitirá a resolução dos nomes sem problemas.

```
$TTL 86400  
@ IN SOA servidor.unatec.com.br root.unatec.com.br. (  
        42           ; serial (d. adams)  
        3H           ; refresh  
        15M          ; retry  
        1W           ; expiry  
        1D )         ; minimum  
@ IN NS      servidor.unatec.com.br.  
servidor   IN A      192.168.20.1  
servidor2  IN A      192.168.20.2  
clientexp  IN A      192.168.20.3  
www       IN CNAME  servidor2.unatec.com.br  
~
```

Após terminar a configuração do arquivo da zona direta, abra o arquivo “reverso” usando o mesmo modelo anterior para a zona reversa. Exemplo:

```

$TTL 86400
0 IN SOA servidor.unatec.com.br root.unatec.com.br. (
        42           : serial (d. adams)
        3H           : refresh
        15M          : retry
        1W           : expiry
        1D )         : minimum
0 IN NS servidor.unatec.com.br.
1 IN PTR servidor.unatec.com.br.
2 IN PTR servidor2.unatec.com.br.
3 IN PTR clientexp.unatec.com.br.

```

Salve e reinicie o bind utilizando o comando **service named restart**

```

[root@centos named]# service named restart
Parando named: [ OK ]
Iniciando named: [ OK ]
[root@centos named]#

```

Caso ocorram erros, verifique o log digitando **tail /var/log/messages**. Para mais informações consulte o man page do Bind ou a documentação relativa à distribuição que você utiliza.

VPN

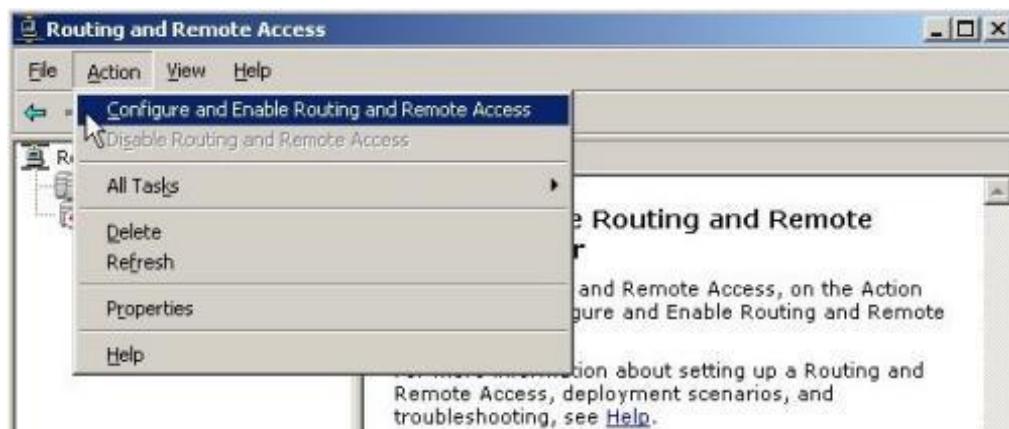
Outro exemplo de serviço que pode ser facilmente configurado para implementar a sua honeynet, é um servidor VPN. Um serviço desse tipo diz para o invasor que você costuma acessar a rede para trabalhar. Ou seja, passa ainda mais credibilidade de que a rede é verdade, e não uma armadilha (o que para azar do invasor, é verdade). Existem vários softwares excelentes para se montar uma VPN, como o OpenVPN, Forefront e diversos outros.

Vou mostrar brevemente como configurar uma VPN com o Windows Server:

Primeiro, deve-se acessar Roteamento e Acesso Remoto (Routing and Remote Access).



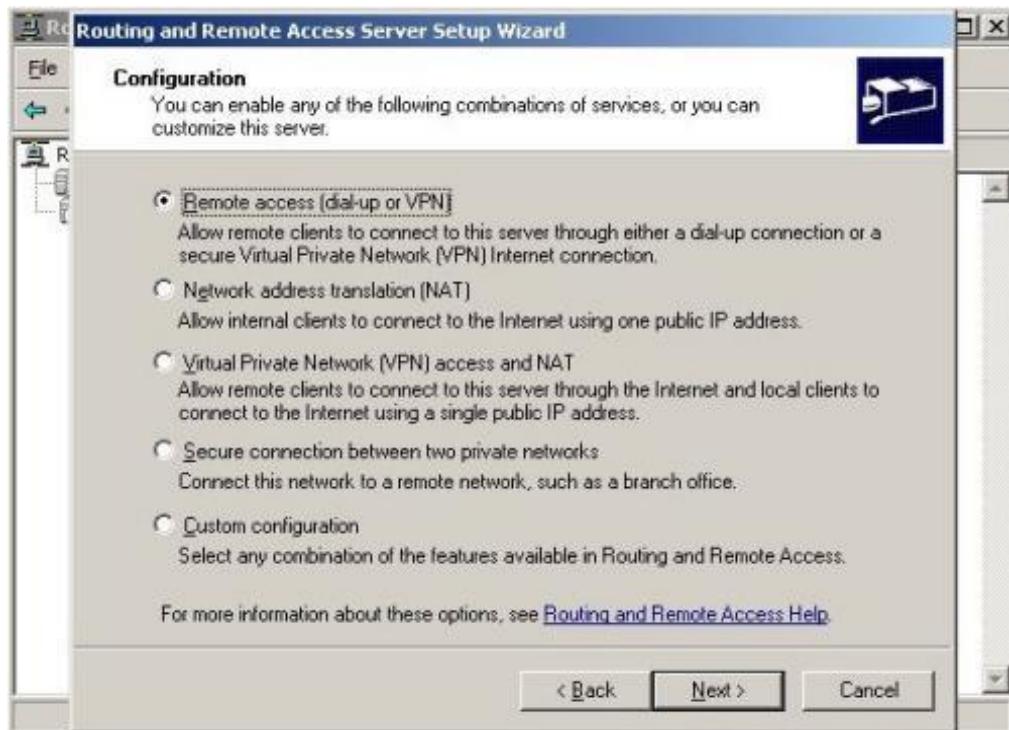
Agora, clique em Ação (Action, em inglês) e selecione Configurar Roteamento e Acesso Remoto (Configure and Enable Routing and Remote Access).



Irá aparecer, na tela, o Wizard para ajudá-lo a configurar o servidor. Clique em próximo (Next) para avançar à próxima tela.



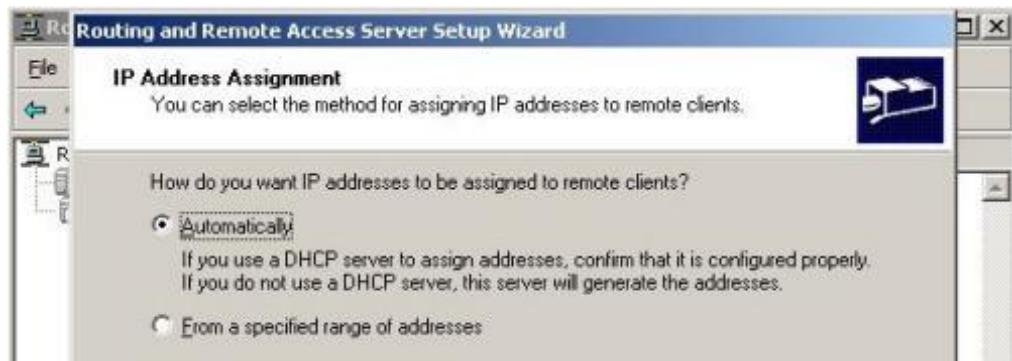
Vai aparecer um menu com muitas opções. Marque a primeira: Acesso Remoto (Remote Access). Clique em Avançar (Next).



Outro menu com mais duas opções se abrirá. Nesse menu você poderá configurar o sistema para receber ligações por modem (Dial-up) e por VPN (acesso pela internet). Em ambos os casos, o usuário receberá um endereço IP da rede interna e conseguirá se comunicar com os outros computadores. A única diferença é o meio: um utiliza um meio físico (conexão discada) e, no caso da VPN, a internet. Clique em avançar (next).



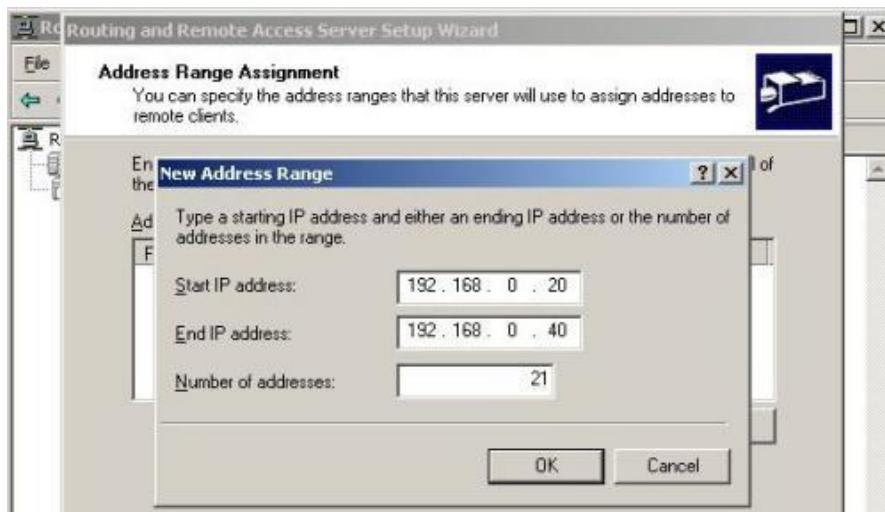
Na próxima janela, o Windows Server 2003 irá lhe perguntar como os endereços IPs devem ser atribuídos aos clientes remotos. Existem duas opções: automaticamente, através de um servidor DHCP ou de um intervalo específico de endereços. Escolha a opção que melhor servir a seu caso. Nós escolheremos automaticamente. Clique em Avançar (Next).



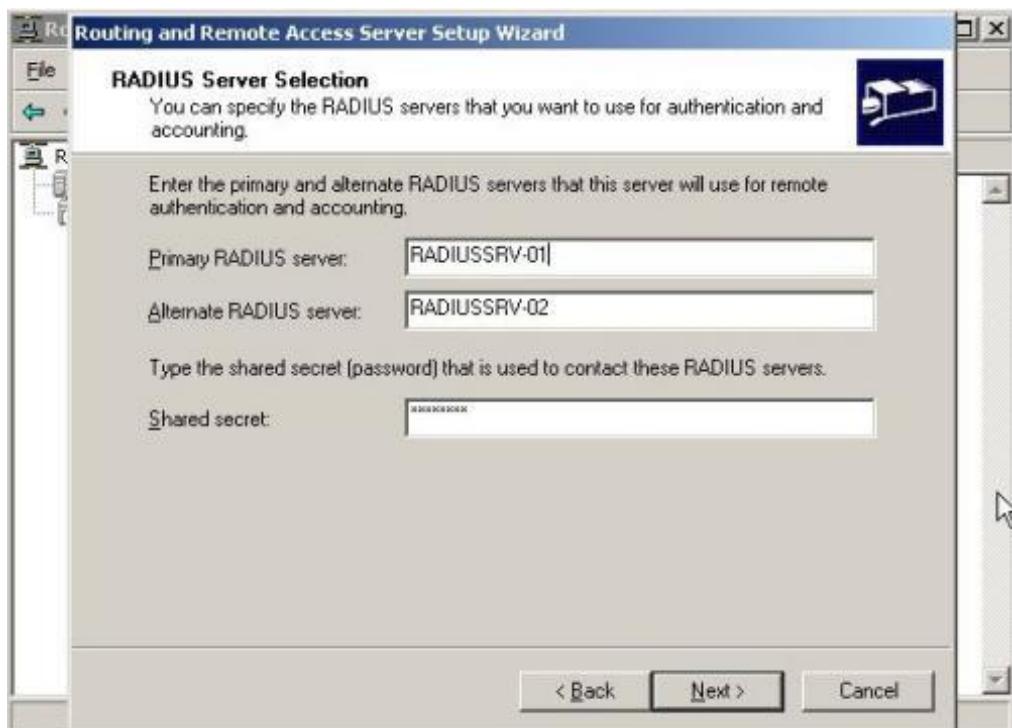
Dependendo da opção da janela anterior, você deve especificar qual o intervalo de endereços IPs que será atribuído para os clientes remotos. Escolha um número adequado e clique em Ok. Agora você irá visualizar o intervalo de endereços IPs que foi definido na tela anterior. Clique em Avançar (Next).



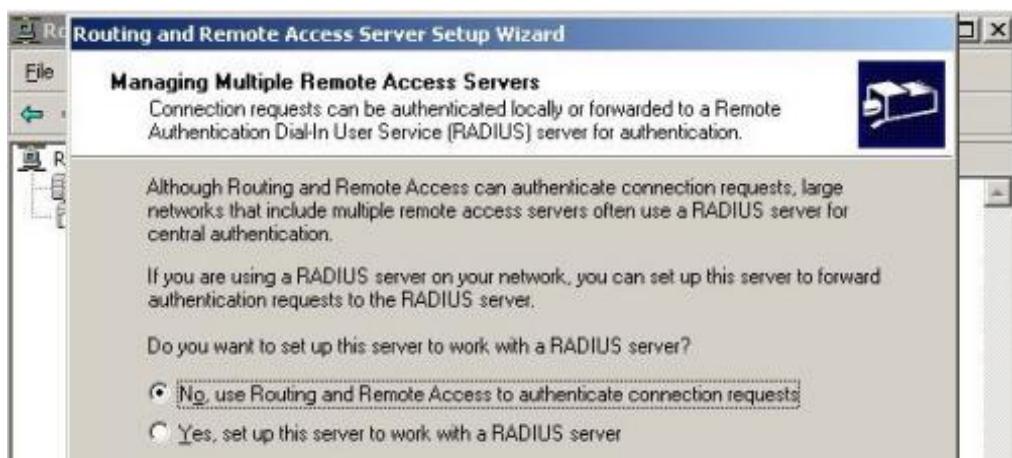
A tela seguinte se refere à autenticação dos usuários. Ela pode ser realizada localmente (no próprio servidor), ou direcionada para um servidor RADIUS que é próprio para a tarefa de autenticação. Escolha a opção que melhor se adapte ao seu caso e avance para a próxima tela.



Caso a opção do RADIUS tenha sido escolhida, é necessário identificar o servidor primário e o secundário, assim como a chave (senha) utilizada para realizar a comunicação com esses dois servidores. Após realizar as configurações pedidas, avance para a próxima tela.

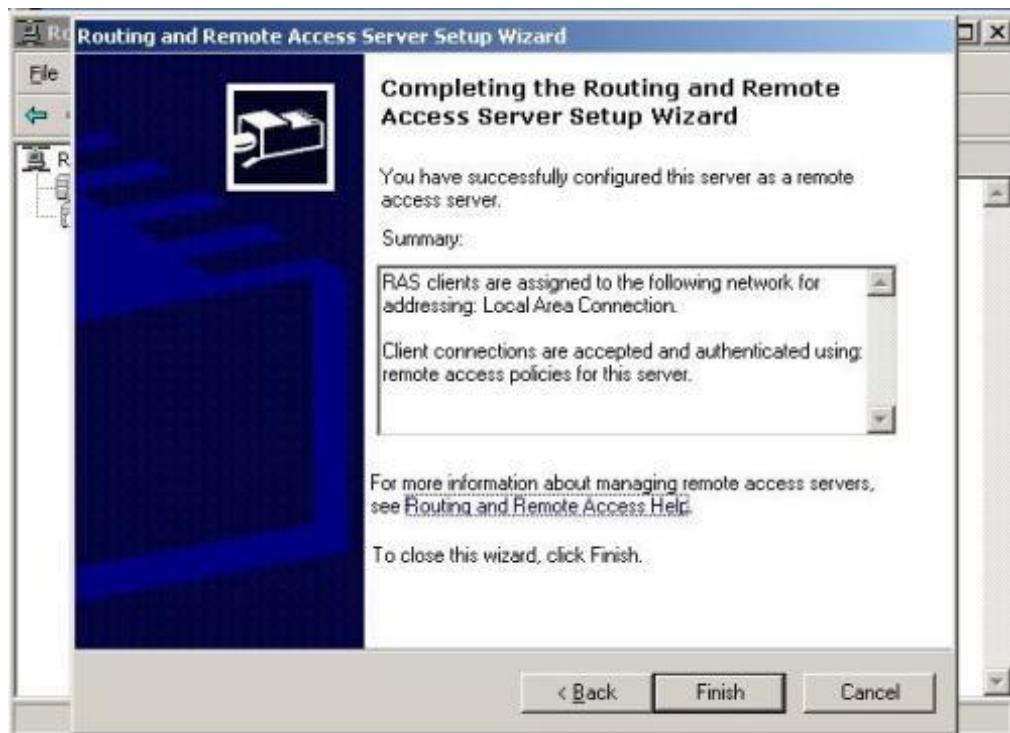


Na próxima tela, irá aparecer a mensagem que você configurou o servidor com sucesso. Aparecerá, também, um sumário das opções que foram configuradas anteriormente. Clique em Finalizar (Finish) para encerrar o Wizard.



Se a opção de deixar os endereços IPs serem atribuídos pelo DHCP for marcada, aparecerá uma mensagem avisando que você deve configurar

posteriormente o endereço IP do seu servidor DHCP dentro do Servidor de Roteamento e Acesso Remoto. Clique em Ok.



Pronto. Após finalizar a configuração, o seu servidor VPN já estará ativo e funcionando completamente. Agora é só aguardar os atacantes tentarem utilizá-lo como alvo de ataques.

Outros serviços

APACHE
HTTP SERVER



Servidor HTTP – Apache ou IIS

É importante ter um servidor de páginas na honeynet , especialmente para a disseminação de honeytokens. Você pode criar informações falsas como nomes de funcionários, datas de nascimento, configurações de rede, até mesmo ir mais a fundo e criando logotipos e cenários para causar uma impressão de maior realismo. Outra coisa interessante é o fato de poder detectar ataques do tipo CGI/Script.

Servidor de FTP – Proftpd ou IIS

O servidor de transferência de arquivos é um dos que com certeza será um dos primeiros serviços que um invasor irá tentar entrar. Especialmente se for daqueles invasores que adoram tentar trocar a página inicial de um site, fazendo uma pichação (defacers). Também é um alvo freqüente a ataques de força-bruta.

Servidor Proxy – Squid ou ISA Server

O servidor Proxy é outro serviço interessante para se ter, só precisa tomar alguns cuidados. Nunca permita que os invasores utilizem o Proxy para se conectar a outros endereços na Internet. Caso isso aconteça, você pode vir a ter problemas. Instale e configure o serviço, mas trabalhe com o formato de White list por garantia.. bloqueie tudo e libere apenas alguns endereços.

Servidor de correio – Exchange, Postfix ou Qmail

O servidor de correio eletrônico é outro que não pode faltar. Mais até do que o servidor http, esse serviço pode ser utilizando de forma maravilhosa para a utilização dos honeytokens dos mais variados tipos. Mensagens pessoais de funcionários, propostas, projetos... todos os tipos de e-mail podem estar contidos nos servidores, apenas esperando para serem lidos pelos invasores.

Servidor de acesso remoto – Telnet, SSH, Remote Desktop

Um serviço desse tipo é um tremendo chamativo para qualquer um que esteja tentando atacar a rede. O telnet por décadas foi um dos tipos de serviços mais atacados, e caiu em desuso graças à sua falta de segurança. Implementar um servidor SSH ou Remote Desktop (Windows), fará com que o invasor tente o possível para obter acesso remoto à seu computador e seus dados.

Para informações mais detalhadas sobre como configurar estes serviços, existem excelentes publicações disponíveis.

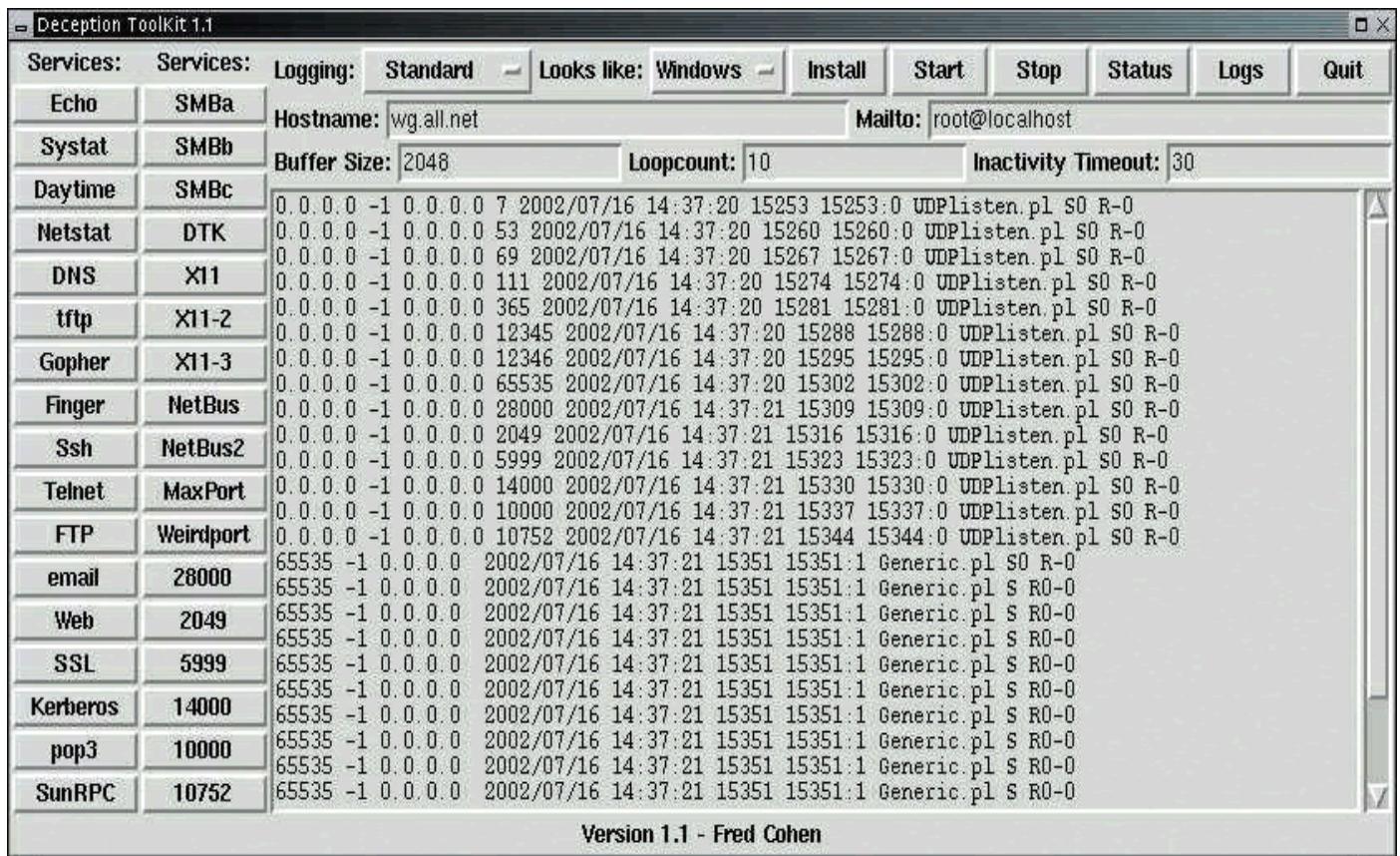
Softwares

Nesse capítulo iremos aprender mais sobre os softwares que podem ser utilizados para criação dos Honeypots. Uma das maiores vantagens em se utilizar um software específico é a facilidade de centralização e monitoração de todos os logs de ataque. Isso sem falar no quanto é complicado de se configurar e monitorar serviços reais a serem monitorados como o Apache, IIS, ProFTPD, etc. Como citei anteriormente, apesar de fornecer um melhor nível de interação com o atacante, uma HoneyNet baseada apenas em sistemas reais pode ser extremamente complexa de se criar e monitorar.

Alguns dos programas iremos apenas conhecer superficialmente, enquanto outros aprofundaremos em maior detalhes. Vamos à eles:

Deception Toolkit

O primeiro honeypot baseado em software foi chamado de Deception Toolkit ou DTK, e foi desenvolvido por Fred Cohen. O lançamento do DTK versão 0.1 foi em novembro de 1997. Ele é um pote de mel de baixa interação que realiza apenas simulações mais básicas da maioria dos serviços conhecidos. Mais informações podem ser obtidas no site do autor do programa: <http://www.all.net>



Honeyd

O Honeyd é o software mais conhecido quando se trata de honeypots. Ele permite a criação de hosts virtuais e o anexo de scripts personalizados em Perl para a criação de interações em determinadas portas. Tem também o código fonte livre pela GPL. Pode ser obtido em www.honeyd.org. Possui versões para Windows e Linux. Abordarei brevemente a versão Linux.

Uma coisa importante a se notar é: o honeyd necessita de endereços IPs não utilizados na rede para montar seus scripts de honeypot, ao contrário de programas como o Valhala Honeypot que rodam utilizando o endereço IP atual.

Pré-requisitos:

Para instalar o Honeyd, as seguintes bibliotecas/pacotes devem estar presentes:

- libevent
- libdnet

- libpcap
- arpd

Após baixar a versão mais nova do Honeyd e instalar as dependências, instale utilizando os comandos tradicionais (caso não possua as bibliotecas do python, deve utilizar a opção without-python ou ocorrerá um erro):

```
./configure [--without-python]
Make
Make install
```

Após a instalação, devemos configurar o arquivo /etc/honeyd.conf .

Uma típica configuração, seria algo como o seguinte:

```
create default
  set default personality "Windows 2008"
  set default default tcp action reset
  set default default udp action open
  add default tcp port 110 "sh scripts/pop.sh"
  add default tcp port 25 block
  add default tcp port 21 "sh scripts/ftp.sh"
  bind 192.168.1.1 default

create linux
  set linux personality "Mandriva Linux 2009"
  set linux default tcp action open
  set linux default udp action block
  add linux tcp port 110 "sh scripts/pop3.sh"
  add linux tcp port 25 "sh scripts/smtp.sh"
  add linux tcp port 22 "sh scripts/test.sh $ipsrc $dport"
  bind 192.168.1.5 linux

create freebsd
  set freebsd personality "FreeBSD 2.2.1-STABLE"
  add freebsd tcp port 80 "sh scripts/web.sh"
  add freebsd port 22 "sh scripts/test.sh $ipsrc $dport"
  bind 192.168.1.15 freebsd
```

Existem dezenas de outras opções que podem ser utilizadas. Consulte o site do Honeyd para maiores detalhes, não é o objetivo aprofundar agora em cada uma delas. Vamos ao menos entender as utilizadas no exemplo acima:

create <perfil> - Essa é a opção que cria um novo perfil de “host virtual”. O padrão é o default, que normalmente é o primeiro a ser configurado. Mas pode

ser outros, como fiz no exemplo anterior configurando um perfil para “Linux” e outro para “freebsd”.

set <perfil> personality <nome> - Define a personalidade padrão do perfil. Esse é um passo importante pois, esse personalidade será utilizada para enganar o processo de identificação de sistemas operacionais (fingerprint) de ferramentas como o NMAP. Portanto é uma opção que realmente ajuda na ilusão de criar um falso sistema. A sub-opção perfil refere-se ao nome do perfil (default, linux ou freebsd no caso), e sub-opção nome é o nome da personalidade. Por exemplo, colocamos “*Mandriva Linux 2009*” no perfil do Linux.

Tome cuidado pois para realmente enganar o NMAP, não podemos colocar qualquer nome como personalidade. Tem de ser um nome que exista no banco de dados de identificação de fingerprints do programa. O nome do perfil Freebsd por exemplo, foi criado com base nessa informação “*FreeBSD 2.2.1-STABLE*”.

set <perfil> default <protocolo> action <ação> - Esse comando irá decidir a ação padrão a ser tomada para todas as portas, sejam TCP ou UDP.

A sub-opção perfil refere-se ao nome do perfil (default, linux ou freebsd no caso).

Protocolo refere-se a um dos protocolos de transporte, TCP ou UDP. A ação que virá a seguir refere-se ao protocolo escolhido aqui.

Já a ação, pode ser:

block – Bloqueia o acesso às portas, enviando uma resposta de conexão recusada.

reset – Faz “cair” a conexão, desconectando o invasor sem enviar nenhuma resposta

open – Mantém o status das portas abertas. Nesse caso, você poderá configurar individualmente cada porta que desejar bloquear utilizando a opção “add”, como visto a seguir.

add <perfil> <protocolo> port <número> <ação> - É com essa opção que iremos adicionar as portas a serem monitoradas no host virtual.

A sub-opção perfil refere-se ao nome do perfil (default, linux ou freebsd no caso). Protocolo refere-se ao protocolo de transporte utilizado, da mesma maneira que na opção anterior.

Número refere-se à qual porta deverá ser aberta.

Já a ação pode ser alguma das já vistas anteriormente (block, reset ou open) ou pode também ser indicado o caminho de um script (que normalmente é em shellscript ou Perl) para que este cuide da interação com os invasores nesta porta. O honeyd já vem com uma grande coleção de scripts.

bind <ip> <perfil> - Essa opção é extremamente importante. É ela que vai indicar qual o endereço IP que o perfil irá utilizar. Lembre que o honeyd necessita de IPs não utilizados na rede para usar em seus hosts virtuais. No exemplo acima designei o IP 192.168.1.1 para o perfil default, 192.168.1.5 para o perfil Linux e 192.168.1.15 para o perfil freebsd.

Após configurar com sucesso o arquivo honeyd.conf, é só executar o honeyd como mostrado a seguir:

```
honeyd -p nmap.prints -f /etc/honeyd.conf -I /var/honeyd/log 192.168.1.1-192.168.1.20
```

Explicando:

honeyd é o executável

A opção **-p nmap.prints** vai tentar fazer com que o NMAP identifique o host virtual da forma que foi configurado na personalidade

A opção **-f** indica o arquivo de configuração dos hosts virtuais a ser carregado

A opção **-I** indica qual o arquivo de log será utilizado para gravar os dados

E a faixa de endereços IPs (**192.168.1.1-192.168.1.20**) , indica o intervalo de endereços que o honeyd irá reservar para utilizar com seus hosts virtuais.

Para testar se o honeypot está funcionando, existe algo muito simples que dá para ser feito, sem mesmo ter que se conectar aos serviços. Pode-se utilizar o NMAP com a opção de detecção de fingerprint habilitada , tendo como destino o host virtual que desejamos. Na imagem a seguir, o teste foi feito contra o IP 192.168.1.15, que está rodando a perfil “FreeBSD”. Veja o resultado:

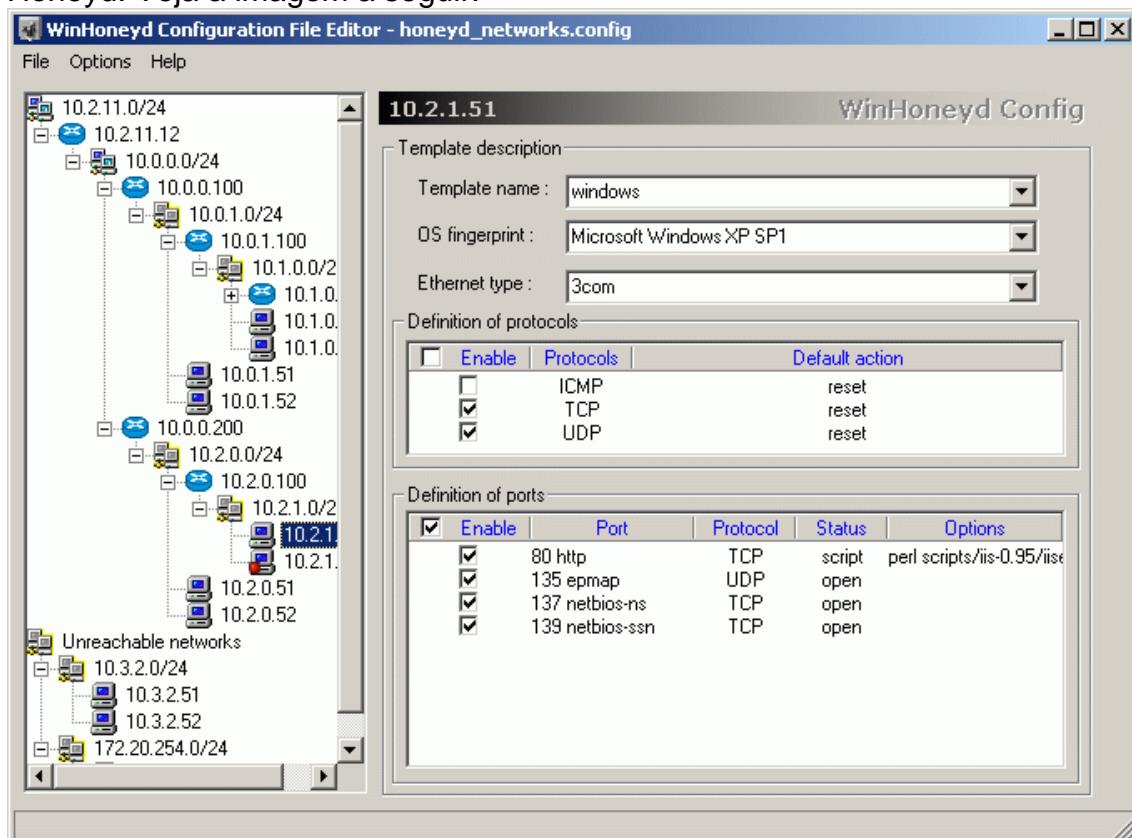
```

xterm
cisd-dhcp-15# nmap -sT -O 192.168.1.15
Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Interesting ports on  (192.168.1.15):
(The 1599 ports scanned but not shown below are in state: closed)
Port      State  Service
22/tcp    open   ssh
80/tcp    open   http
Remote operating system guess: FreeBSD 2.2.1-STABLE

Nmap run completed -- 1 IP address (1 host up) scanned in 6 seconds
cisd-dhcp-15#

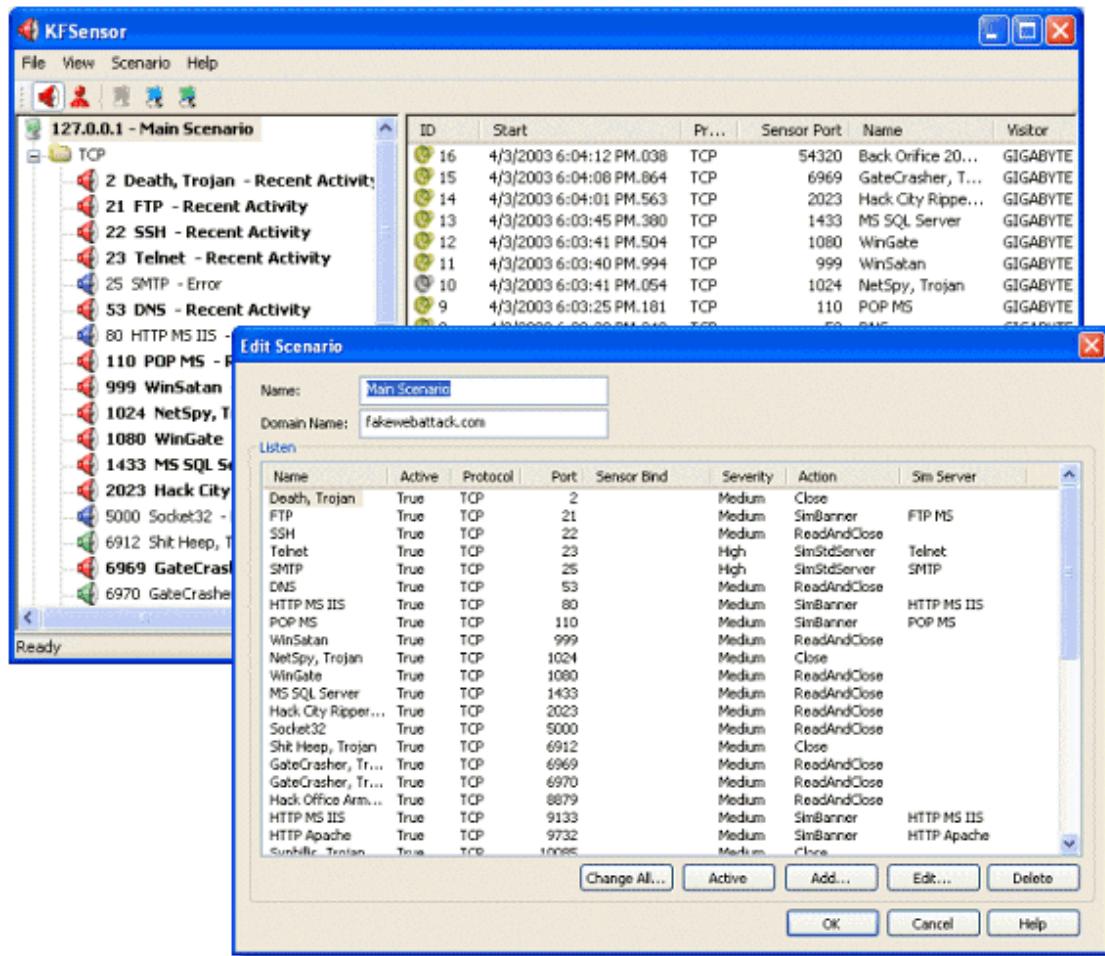
```

Caso você ache complicado utilizar o honeyd em sistemas Linux, pode tentar a versão para Windows. Mas já aviso que ela não é tão estável, as utilizando o WinHonMeyd você pode configurar todo o programa sem precisar utilizar uma única linha de texto. Para mais informações sobre ele, visite a página do Honeyd. Veja a imagem a seguir:



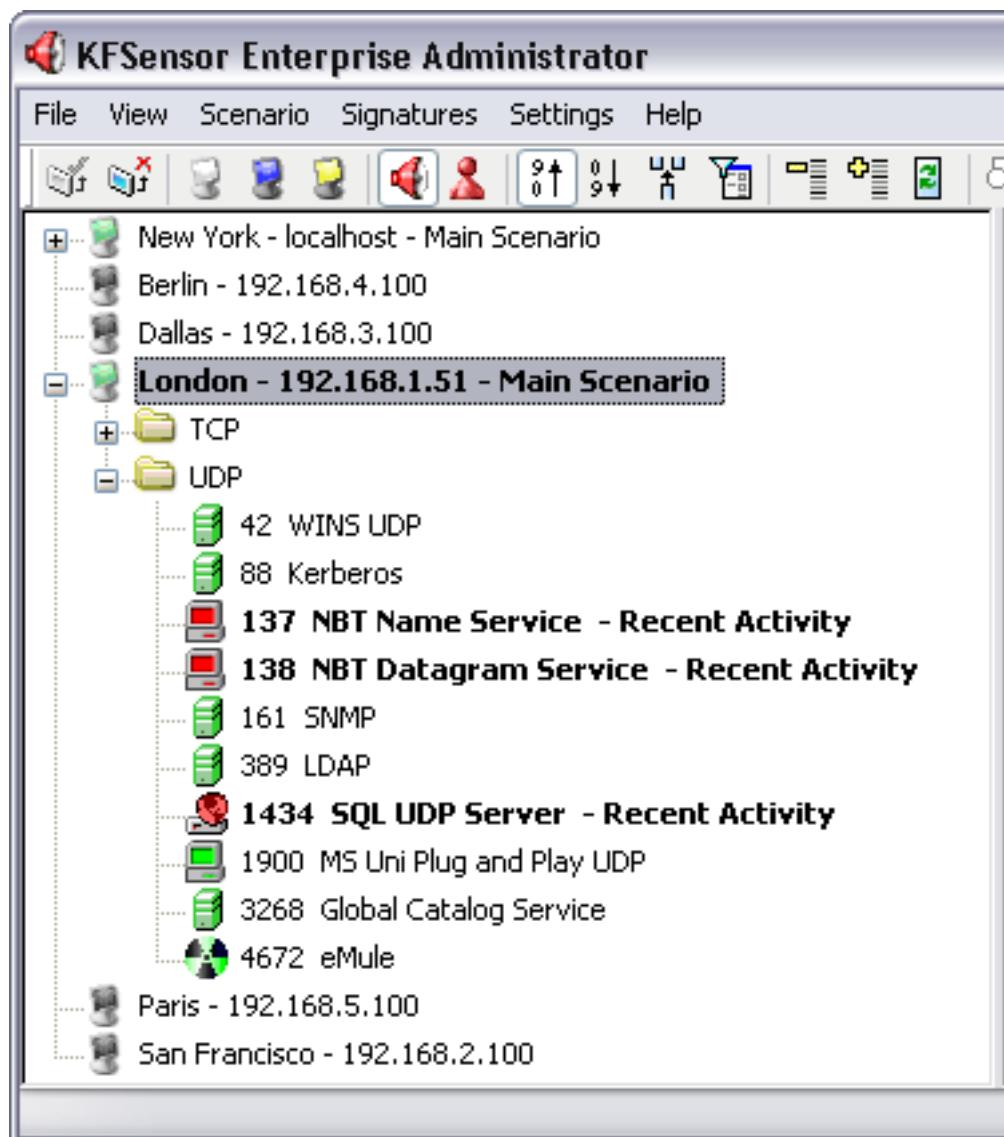
KFSensor

O KFSensor (KeyFocus Sensor) é um honeypot comercial para Windows. Foi um dos primeiros que apareceu no segmento. Foi desenvolvido pela KeyFocus (www.keyfocus.com) .



Assim como o honeyd ele trabalha primariamente com recursos de baixa interatividade, simulando respostas ao invés de fornecer dados reais, como no caso de alta interatividade. Um dos recursos interessantes é a capacidade de instalação de “cenários” em locais estratégicos da rede, que permitem monitorar o tráfego e enviar a resposta remotamente para o computador que está rodando o servidor do KFSensor. Na imagem anterior dá para visualizar os serviços e portas ativas em apenas um endereço IP, o 127.0.0.1 (localhost).

Em um cenário um pouco mais complexo, a instalação de mais sensores irá permitir a visualização e configuração dos serviços em qualquer endereço IP com grande facilidade, como visto a seguir:



O interessante também é que os logs são divididos por “cenários”. No exemplo da imagem anterior, foi clicado no cenário de Londres, que irá mostrar apenas os eventos que ocorreram nos serviços configurados naquele local. Isso ajuda muito quando você precisa encontrar algum evento específico em um determinado lugar, sem se preocupar em ter que filtrar os que não lhe servem. Observe os eventos sendo exibidos à direita da tela:

The screenshot shows the KFSensor Enterprise Administrator interface. The left pane displays a hierarchical tree of sensor activity, including categories like TCP, SMTP, DNS, and various service ports. The right pane is a detailed log table with columns for Sensor ID, ID, Start Time, Pr..., Sens..., Name, Visitor, Sig. Message, and Received. The log entries show various network interactions, such as TCP connections from New York and Berlin to different ports (e.g., 21, 22, 25, 445, 80, 139) and services (e.g., NetBus, GateCrasher, MS RPC, NBT SMB, SQL Server). The 'Received' column contains raw log data, and the bottom status bar indicates the server is running with 23 visitors and 39 events.

Sensor ID	ID	Start Time	Pr...	Sens...	Name	Visitor	Sig. Message	Received
New York	1039	12:04:07.609	TCP	20034	NetBus	192.168.1.41	BN[1F 00 02 00 DC]3[05 00]A[0C	[16 03 00 00]a[01 00 00][03 00]
New York	1038	12:02:02.671	TCP	6969	GateCrasher, T...	192.168.1.41	[16 03 01 00]a[01 00 00][03 01]	[16 03 01 00]a[01 00 00][03 01]
New York	1037	12:01:27.671	TCP	5631	PC Anywhere 1	192.168.1.41		[05 00 08 03 10 00 00 00]H[00 0
Berlin	1036	12:00:20.937	TCP	135	MS RPC	81.193.24.196		[05 00 08 03 10 00 00 00]H[00 0
Berlin	1035	11:58:20.859	TCP	135	MS RPC	host81-153-139-157....		[05 00 08 03 10 00 00 00]H[00 0
New York	1034	11:57:51.968	TCP	445	NBT SMB	host81-153-68-191.r...	NBT SMB - ASN.1 Kill Bil...	SMB:1 [neg protocol][0D 0A] Pr
New York	1033	11:56:14.093	TCP	139	NBT Session Se...	218.37.25.20	NBT OpaSoft Worm ins...	NBT:1 Session Request[0D 0A]C
Berlin	1032	11:55:25.281	TCP	1080	SOCKS	60.220.1.32		SOCKS 5 Authenticate Request:
New York	1031	11:41:36.453	TCP	445	NBT SMB	host81-153-16-239.r...		
Berlin	1030	11:40:42.234	TCP	4444	Blaster, Trojan	ACC89EE8.pt.aol.com	Command console wor...	tftp -i 172.200.158.232 GET teel
New York	1029	11:39:04.312	UDP	1434	SQL UDP Server	host81-153-27-215.r...	SQL UDP Server Resol...	[04 90 90 90 90 90 90 90 9
New York	1028	11:38:34.828	TCP	1433	SQL Server	host81-152-241-78.r...	SQL Server logon atte...	TDS Packet: Num:1 Type id:12 T:
Berlin	1027	11:36:34.703	TCP	1433	SQL Server	host81-152-241-78.r...	SQL Server logon atte...	TDS Packet: Num:1 Type id:12 T:
Berlin	1026	11:35:25.046	TCP	139	NBT Session Se...	ADSL-TPLUS-16-55.in...	NBT OpaSoft Worm ins...	NBT:1 Session Request[0D 0A]C
London	1025	11:31:30.187	TCP	445	NBT SMB	host81-153-63-210.r...		SMB:1 [neg protocol][0D 0A] Pr
London	1024	11:31:30.125	TCP	445	NBT SMB	host81-153-63-210.r...		
New York	1023	11:26:09.453	TCP	445	NBT SMB	I220-109-122-148.50...		SMB:1 [neg protocol][0D 0A] Pr
London	1022	11:25:58.484	TCP	80	IIS	host81-153-130-36.r...	IIS view script source c...	OPTIONS / HTTP/1.1[0D 0A]tran
New York	1021	11:22:30.640	UDP	137	NBT Name Service	host81-153-15-137.r...	NBT NS Packet: Op: Name Query	
New York	1020	11:21:57.890	TCP	80	IIS	host81-153-183-246....	IIS - RBOT Worm prop...	GET /HTTP/1.0[0D 0A]Host: 81.
New York	1019	11:20:17.203	TCP	3128	HTTP Proxy	209.200.16.65		GET http://www.rezilient.net/tes
London	1018	11:18:15.156	TCP	8080	HTTP Proxy	194.186.26.24		GET http://top.list.ru/counter?id
New York	1017	11:17:32.031	TCP	25	SMTP	47.Red-81-45-234.po...		EHLO[0D 0A]X-LINK2STATE CHUL
New York	1016	11:16:59.562	TCP	53	DNS	192.168.1.41		[00] [EF B3 01 00 00 01 00 00 00 0C]
New York	1015	11:15:19.140	TCP	4899	radmin	IGLD-80-230-252-73.i...		[01 00 00 01 00 00 00 08 08]
London	1014	11:14:46.625	TCP	1433	SQL Server	host81-153-4-53.ran...	SQL Server logon atte...	TDS Packet: Num:1 Type id:12 T:
New York	1013	11:13:01.796	TCP	9996	Sasser worm co...	host81-153-96-237.r...	Sasser worm transfer ...	echo off&echo open 81.153.96.2
New York	1012	11:12:24.562	TCP	80	IIS	host81-153-255-122....	IIS view script source c...	PROPFIND /C%24 HTTP/1.1[0D 0
London	1011	11:11:22.531	TCP	80	IIS	host81-153-255-122....	IIS view script source c...	PROPFIND /C%24 HTTP/1.1[0D 0
New York	1010	11:10:22.343	TCP	80	IIS	host81-153-255-122....	IIS view script source c...	OPTIONS / HTTP/1.1[0D 0A]tran
New York	1009	11:09:21.015	TCP	80	IIS	host81-153-255-122....	IIS view script source c...	PROPFIND /C%24 HTTP/1.1[0D 0

Visualizando informações de log

O KFSensor guarda um registro completo para cada detecção realizada por ele. É só clicar duas vezes no evento, que uma janela irá aparecer. Quatro abas existem em uma janela de evento:

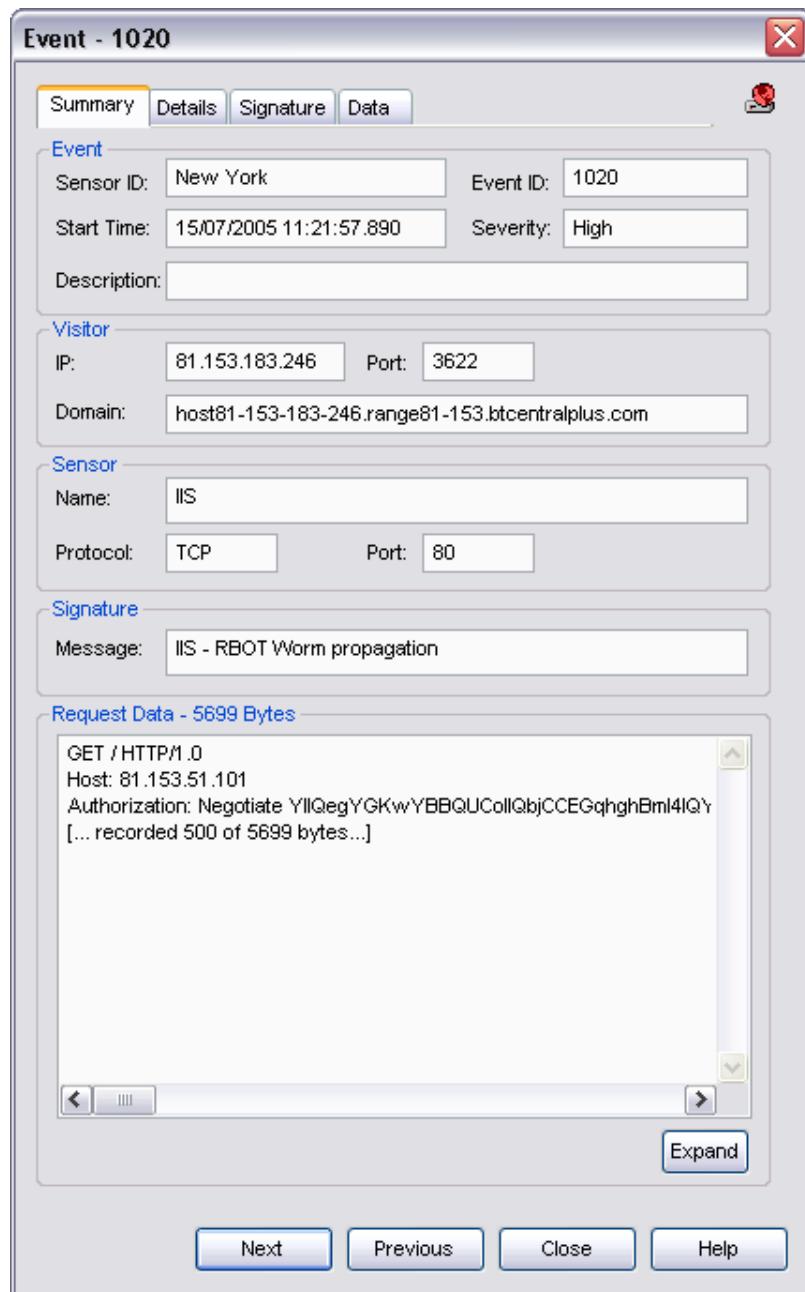
Summary : Mostra informações gerais sobre o evento, como o sensor, IP, porta, etc.

Details : Mostra alguns detalhes técnicos mais profundos sobre o evento, como por exemplo que interações o invasor realizou no serviço.

Signature : Mostra a assinatura do evento, que pode ser alterada caso você deseje. Veremos isso um pouco mais à frente.

Data : Possíveis dados recolhidos durante a intrusão.

Em sumário, temos os dados demonstrados na imagem a seguir:



Informações do evento:

Sensor ID: O sensor que detectou o ataque

Event ID: A identificação numérica dada ao evento pelo KFSensor

Start time: A data e a hora que o evento ocorreu

Severity: A gravidade do evento. Baixa, média ou alta.

Description: Uma descrição sobre o evento.

Informações do visitante:

IP: endereço ip que fez o ataque

Port: porta que o invasor utilizou para se conectar

Domain: o nome pertencente ao IP, descoberto através de consulta DNS reversa.

Informações do Sensor:

Name: o nome do sensor que foi utilizado. Normalmente o sensor (no contexto do KFSensor) é um script ou módulo que irá “simular” a interação com o invasor. Na imagem anterior, foi o IIS, que simula o servidor WEB da Microsoft.

Protocol: O protocolo utilizado pelo sensor. Normalmente é TCP, UDP ou ICMP.

Port: A porta utilizada pelo sensor para receber conexões.

Informações de assinatura (signature)

Message: mensagem acusada quanto o tráfego capturado for equivalente à assinatura configurada no sensor.

Request Data: Um resumo dos dados que o KFSensor capturou durante a tentativa de intrusão.

Para saber mais detalhes do que o atacante fez, é necessária clicar na aba “Details”, que fornecerá informações mais detalhadas e específicas sobre o ataque que foi detectado. Observe a seguir em detalhes outro evento capturado pelo KFSensor:

Event Details Text Viewer

Format: Text Close Help

```
NBT:1 Session Request
Called Name : GIMLI<20 File Server Service>
Calling Name: MOMEADD<20 File Server Service>

NBT:2 SMB: [tree con X]
    (!)\GIMLI\{00}A:[00]

NBT:3 SMB: [create file]
    ([04]WINDOWS\speedy.pif[00])

NBT:4 SMB: [write]
    Write 512 bytes at offset 0

NBT:51 SMB: [close file]

File Operation:
Uploaded file: speedy_pif_200_97_219_90_1217.bin MD5: 3e004c7e4e73dee4c4cdc40b63bb6537

NBT:52 SMB: [open file]
    ([04]WINDOWS\win.ini[00])

NBT:53 SMB: [read]
```

Visualizando e alterando assinaturas de ataque

Para visualizar a assinatura do evento, acesse a aba “Signature”. Observe:

Event - 1020

Summary	Details	Signature	Data													
Event <table border="1"> <tr> <td>Sensor ID:</td> <td>New York</td> <td>Event ID:</td> <td>1020</td> </tr> <tr> <td>Start Time:</td> <td>15/07/2005 11:21:57.890</td> <td>Severity:</td> <td>High</td> </tr> </table>				Sensor ID:	New York	Event ID:	1020	Start Time:	15/07/2005 11:21:57.890	Severity:	High					
Sensor ID:	New York	Event ID:	1020													
Start Time:	15/07/2005 11:21:57.890	Severity:	High													
Signature <table border="1"> <tr> <td>ID:</td> <td>KFAGC174151</td> </tr> <tr> <td>Message:</td> <td>IIS - RBOT Worm propagation</td> </tr> <tr> <td>Reference:</td> <td>http://www.keyfocus.net/kfsensor/signature/sigb/</td> <td>Browse</td> </tr> <tr> <td>Source:</td> <td>KeyFocus</td> </tr> <tr> <td>Created:</td> <td>03/07/2005 17:44:27.375</td> </tr> <tr> <td>Edited:</td> <td>05/07/2005 23:08:08.062</td> </tr> </table>				ID:	KFAGC174151	Message:	IIS - RBOT Worm propagation	Reference:	http://www.keyfocus.net/kfsensor/signature/sigb/	Browse	Source:	KeyFocus	Created:	03/07/2005 17:44:27.375	Edited:	05/07/2005 23:08:08.062
ID:	KFAGC174151															
Message:	IIS - RBOT Worm propagation															
Reference:	http://www.keyfocus.net/kfsensor/signature/sigb/	Browse														
Source:	KeyFocus															
Created:	03/07/2005 17:44:27.375															
Edited:	05/07/2005 23:08:08.062															
<input type="button" value="Edit"/> <input type="button" value="Create"/>																

Provavelmente você vai querer também criar novas assinaturas ou editar as que já existentes. Elas funcionam mais ou menos da mesma maneira como em um IDS tradicional, como o Snort. O KFSensor permite que você faça isso com grande facilidade, lhe fornecendo inclusive uma lista com todas as assinaturas disponíveis para que você altere ou simplesmente faça uma nova.

The screenshot shows a software interface titled 'Edit Signatures'. At the top, there are buttons for 'Edit...', 'Add...', 'Delete', 'Scan Sort', 'Export', 'Purge', 'OK', 'Cancel', 'Help', and checkboxes for 'Active' and 'Archived' status. A progress bar indicates '31/31'. The main area is a table with columns: ID, Message, Source, From Filter, To Filter, Signature, Severity, and Action. The table lists numerous signatures, mostly related to IIS and KeyFocus protocols, with various severity levels (High, Medium) and actions (Lock out, etc.).

ID	Message	Source	From Filter	To Filter	Signature	Severity	Action
KFAGE132613	Blaster UDP Nessus probe	KeyFocus	any	UDP 69	[00 01]nessus1...	High	Lock out
KFAGC164507	Command console worm upload attempt	KeyFocus	any	Command c...	tftp -i	Medium	
KFAGH163832	FTP - Nessus FTP probe	KeyFocus	any	TCP 21	nessus.org	High	Lock out
KFAGE132723	HTTP Proxy - Nessus probe	KeyFocus	any	IIS	GET http://ww...	High	Lock out
KFAGD111440	HTTP Proxy - Russian proxy checker	KeyFocus	any	HTTP Proxy	GET http://ww...		
KFAGH162029	IIS - CGI script	KeyFocus	any	IIS	.cgi	High	
KFAGH152527	IIS - cmd.exe access attempt	KeyFocus	any	IIS	cmd.exe	High	
KFAGH153319	IIS - CodeRed 2 remote access	KeyFocus	any	IIS	/root.exe	Medium	
KFAGH152211	IIS - directory traversal	KeyFocus	any	IIS	../	High	
KFAGH152354	IIS - directory traversal windows	KeyFocus	any	IIS	..\..	High	
KFAGH153821	IIS - Exchange address list	KeyFocus	any	IIS	exchange/root....	High	
KFAGH163144	IIS - Front page bin	KeyFocus	any	IIS	/_vti_bin/	High	
KFAGH162844	IIS - iisamples	KeyFocus	any	IIS	/iisamples/	High	
KFAGE131937	IIS - Nessus probe	KeyFocus	any	IIS	User-Agent: ...	High	Lock out
KFAGH154014	IIS - PERL script	KeyFocus	any	IIS	.pl	High	
KFAGH162133	IIS - PHP script	KeyFocus	any	IIS	.php	High	
KFAGD111856	IIS - Proxy connection	KeyFocus	any	IIS	CONNECT	High	
KFAGC174151	IIS - RBOT Worm propagation	KeyFocus	any	IIS	[0D 0A]Authori...	High	
KFAGH153044	IIS - Script insertion attempt	KeyFocus	any	IIS	<SCRIPT>	High	

Os dados fornecidos são:

ID : A identificação da assinatura

Message: A mensagem mostrada quando a assinatura é detectada no tráfego.

Source: O tipo de assinatura. Se ela foi criada para o KFSensor, indicará “KeyFocus”.

From Filter: É o filtro de protocolo, que pode ser por protocolo (TCP , UDP ou ambos), por porta, ou outros. “Any” significa nenhum filtro específico.

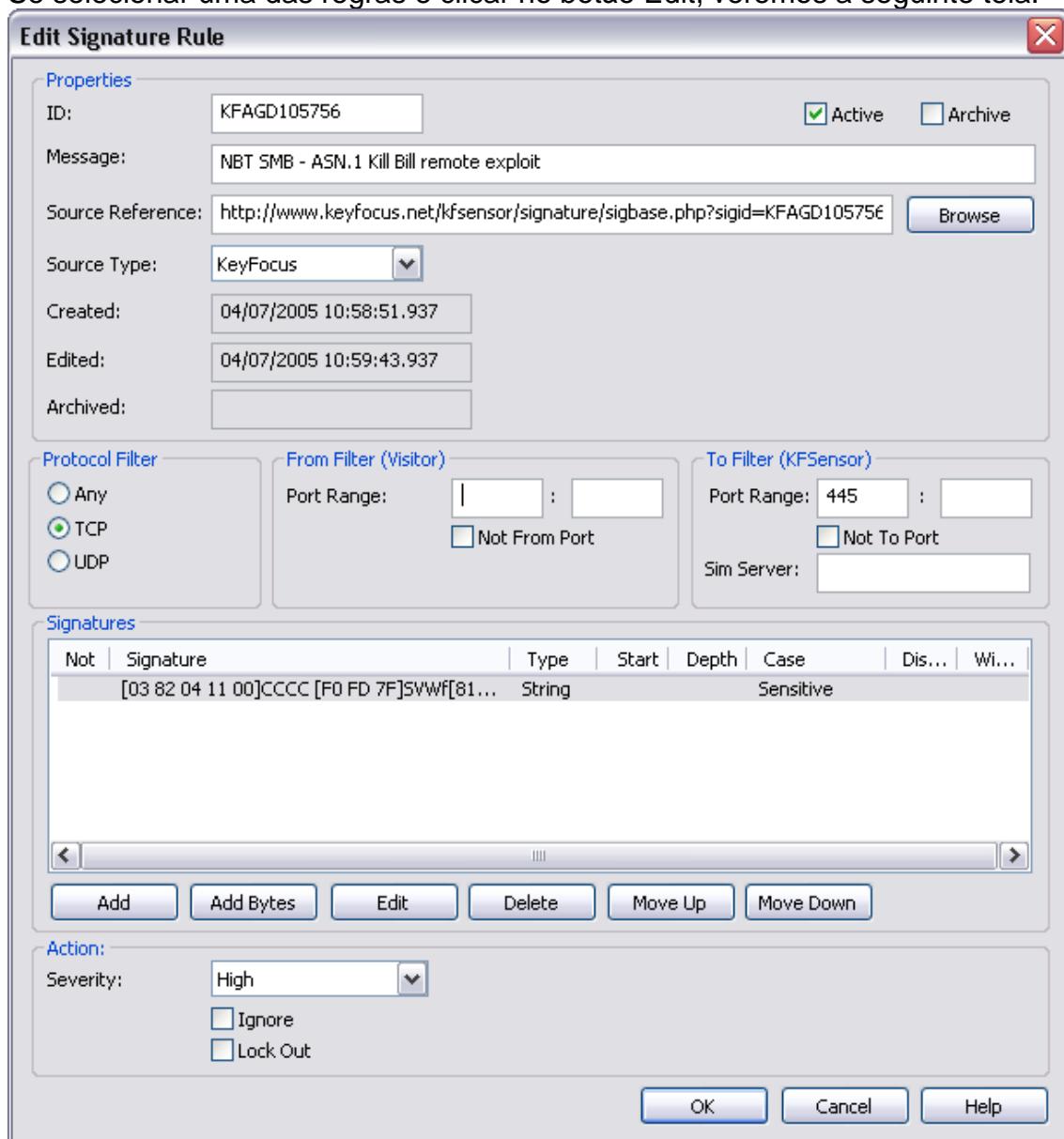
To Filter: Para qual módulo/protocolo o ataque se destina.

Signature: A assinatura em si. Em outras palavras, o que deve ser detectado para que o “alarme” toque e a mensagem de alerta seja enviada.

Severity: A gravidade da assinatura. Baixa (low), Média (Medium) ou Alta (High).

Action: A ação ao ser tomada caso um ataque seja detectado. Normalmente as opções são “Ignore” (ignorar) ou “Lock out” (desconectar o invasor e impedir novas conexões no serviço).

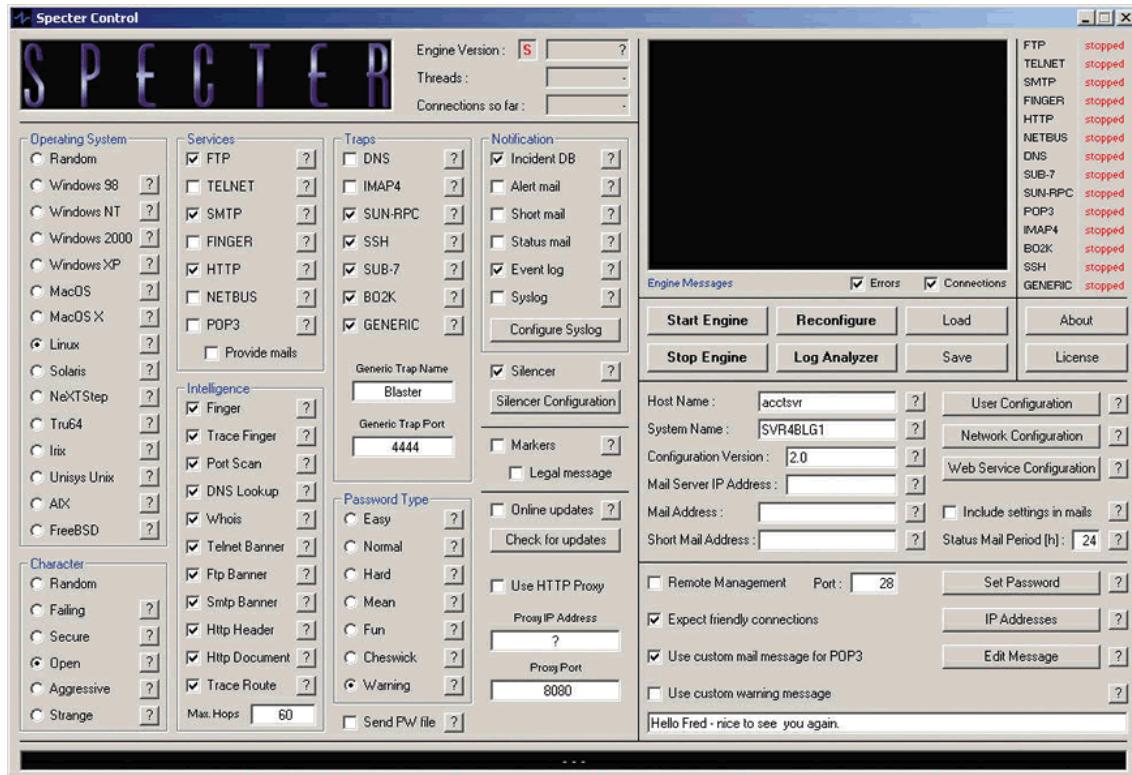
Se selecionar uma das regras e clicar no botão Edit, veremos a seguinte tela:



É nesse tela que você irá alterar todas as opções citadas anteriormente, como a assinatura, o filtro de origem e destino a ser utilizado, a gravidade do ataque, a ação a ser tomada, a mensagem de alerta, e outras opções.

Specter

SPECTER é um honeypot comercial de baixa interação para Windows. Pode ser encontrado em www.specter.com. Ele possui diversos recursos interessantes, como por exemplo a capacidade de simular muitos diferentes sistemas operacionais (*Windows 98, Windows XP, Windows 2003, Linux, Solaris, AIX, FreeBSD, MacOS , Tru64, etc*).



Ele consegue também simular os seguintes serviços (*ftp, telnet, finger, pop3, http, ssh, dns, netbus, Sun-rpc, imap4, smtp, bo2k, generic trap e sub-7*).

Todas as conexão são logadas junto com o endereço IP do atacante, a hora, tipo de serviço e o estado do serviço simulado.

Existem 5 diferentes modos que podem ser utilizada para o host simulado:

Aberto – O sistema se comporta como se estivesse mal configurado.

Seguro – O sistema se comporta como bem configurado em termos de segurança.

Estranho – O sistema se comporta de maneira estranha e deixa o invasor imaginando o que está acontecendo.

Agressivo – O sistema se comunica o quanto for necessário para coletar informações sobre o atacante.

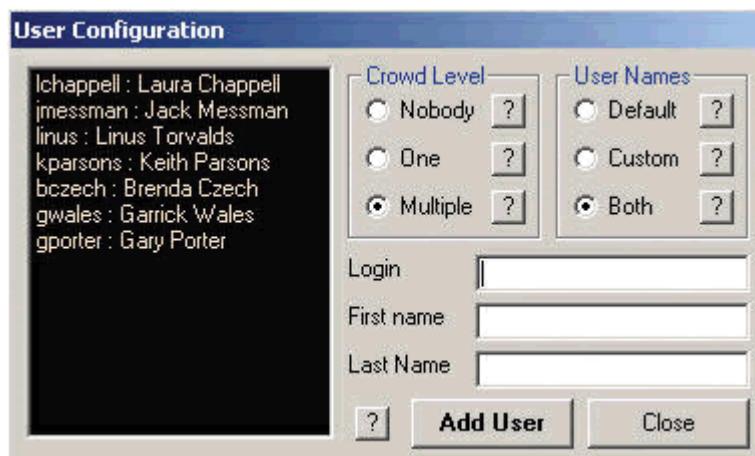
Falho – O sistema se comporta como um pc com problemas de hardware/software.

Outra coisa interessante é que o SPECTER pode ser configurado para fornecer um arquivo de senhas ao atacante, para tornar a simulação bem real. Podem ser tanto no formato Windows NT (SAM) quanto Unix-like (PASSWD / SHADOW).

Configuração de usuários

Uma opção que está se tornando comum em quase todos os honeypots comerciais é a capacidade de lhe permitir criar usuários. O motivo para isso é simples: imagine que o invasor tentará invadir um servidor FTP, TELNET ou SSH. Esses serviços normalmente requerem uma autenticação inicial. Configurar então o nome dos usuários que podem ser utilizados nos serviços do honeypot é uma coisa muito importante.

Para realizar isso no SPECTER, basta clicar no botão “User Configuration”



As únicas informações que devem ser fornecidas são:

Login: o nome da conta em si

First name: primeiro nome do dono da conta

Last name: último nome do dono da conta

Você pode estar imaginando de que importa o primeiro e último nome do usuário, se o que o invasor irá realmente utilizar é o login. É simples. Essas opções são utilizadas para a criação do falso arquivo de senhas ou para ser fornecidos em serviços como o finger.

Visualizando incidentes

Para visualizar os logs capturados, clique no botão “Log Analyzer”. Uma outra tela será aberta demonstrando as capturas realizadas até o momento. As informações salvas são o tipo de protocolo, a data, o IP de origem, e o arquivo de log no qual as informações foram gravadas. À esquerda você pode filtrar quais dos eventos deseja visualizar, por protocolo. Muito útil caso você deseje procurar um evento que aconteceu em um serviço específico, sem se preocupar com os eventos de outros serviços.

Veja a imagem:

The screenshot shows the 'Log Analyzer' window with the following details:

- Services / Traps / Watcher:** A list of protocols with checkboxes:
 - FTP (checked)
 - TELNET (checked)
 - SSH (unchecked)
 - SMTP (checked)
 - FINGER (checked)
 - HTTP (checked)
 - NETBUS (unchecked)
 - DNS (unchecked)
 - SUB-7 (unchecked)
 - SUN-RPC (unchecked)
 - POP3 (checked)
 - IMAP4 (unchecked)
 - B02K (unchecked)
 - GENERIC (unchecked)
 - IRC (unchecked)
 - ICMP (checked)
 - TCP (checked)
 - UDP (checked)
- Service/Trap Filter:** A checked checkbox.
- Source IP address:** A dropdown menu set to "Single address" with the value "192.168.1.1".
- Source IP Address Filter:** An unchecked checkbox.
- Time Frame:** A section with "From" and "To" date/time inputs. The "From" input shows "2001 1 1 00 00" and the "To" input shows "2005 12 31 23 59". Both sections have dropdown menus for Year, Month, Day, Hour, and Min.
- Time Frame Filter:** A checked checkbox.
- Table:** A grid showing captured events with columns: Type, Time, Source IP, and File name.

Type	Time	Source IP	File name
FTP	2001/11/03 05:16:42	212.185.235.84 (pD4B9EB54.dip.t-dialin.net)	FTP-20011103-051642.txt
SMTP	2002/01/10 18:38:54	24.39.62.186	SMTP-20020110-183854.txt
SMTP	2002/01/10 18:38:58	24.39.62.186	SMTP-20020110-183858.txt
TELNET	2002/10/27 23:07:59	66.31.254.174 (h525400e90505.ne.client2.attbi.com)	TELNET-20021027-23075..
POP3	2003/04/26 19:03:49	195.49.71.204	POP3-20030426-190349.txt
FTP	2003/05/14 14:22:35	211.189.88.171	FTP-20030514-142235.txt
HTTP	2003/05/14 19:52:30	195.40.196.230 (Int-2-230.easynet.co.uk)	HTTP-20030514-195230.txt
HTTP	2003/05/14 19:52:32	195.40.196.230 (Int-2-230.easynet.co.uk)	HTTP-20030514-195232.txt
HTTP	2003/05/14 19:52:34	195.40.196.230 (Int-2-230.easynet.co.uk)	HTTP-20030514-195234.txt
HTTP	2003/05/14 21:34:34	207.195.212.171	HTTP-20030514-213434.txt
HTTP	2003/05/14 22:30:53	207.195.212.171	HTTP-20030514-223053.txt
FTP	2003/05/15 06:47:44	24.145.176.13 (user-0c93c0d.cable.mindspring.com)	FTP-20030515-064744.txt
FTP	2003/05/15 06:53:04	24.145.176.13 (user-0c93c0d.cable.mindspring.com)	FTP-20030515-065304.txt
FTP	2003/05/15 07:01:06	80.13.2.21 (AGrendle-102-1-2-21.w80-13.abo.wana...	FTP-20030515-070106.txt
FTP	2003/05/15 11:08:25	80.13.2.21 (AGrendle-102-1-2-21.w80-13.abo.wana...	FTP-20030515-110825.txt
HTTP	2003/05/15 11:55:00	195.166.231.118	HTTP-20030515-115500.txt
HTTP	2003/08/13 23:48:25	195.49.71.204	HTTP-20030813-234825.txt
HTTP	2003/08/13 23:48:46	195.49.71.204	HTTP-20030813-234846.txt
TCP	2005/01/26 17:51:38	172.16.1.204 (nellis.netsec.local)	TCP-20050126-175138.txt
ICMP	2005/01/27 16:14:55	172.16.1.160 (vanadium.netsec.local)	ICMP-20050127-161455.txt
ICMP	2005/01/27 17:22:16	172.16.1.176 (narria.netsec.local)	ICMP-20050127-172216.txt
ICMP	2005/01/27 17:24:13	172.16.1.204 (nellis.netsec.local)	ICMP-20050127-172413.txt
ICMP	2005/01/27 17:24:14	172.16.1.204 (nellis.netsec.local)	ICMP-20050127-172414.txt
UDP	2005/01/27 22:36:58	172.16.1.160 (vanadium.netsec.local)	UDP-20050127-223658.txt
- Buttons:** Search, Show All, Clear, Help, Page Number (24), Close.

Ao selecionar um dos eventos, você pode acessar o seu conteúdo (que foi salvo no arquivo de log mostrado na frente do evento). Assim poderá ver toda a interação que foi realizada pelo invasor durante aquela tentativa.

Veja o conteúdo de uma tentativa de acesso através do servidor FTP:

The screenshot shows a Windows-style application window titled "Incident Info". The title bar includes standard window controls (minimize, maximize, close). The main area displays a log entry for an FTP connection. At the top, it shows "Type : FTP", "Source IP : 80.117.220.222 [host222-220.pool80117.interbusiness.it]", "Time : 2003/11/30 06:11:28", and "File : FTP-20031130-061128.txt". Below this, a "Protocol Log" section contains the following text:

```

Client connecting: 80.117.220.222
Client tries anonymous Login
-->331 Guest login ok, send your complete e-mail address as password.
PASS 'ano@ano.com'. Revealing identity, sending custom warning message
-->421 Hello Fred - good to have you back again.

Closing connection with 80.117.220.222

```

At the bottom of the window is a toolbar with several buttons: "Protocol Log" (highlighted in blue), "Finger", "Port Scan", "Telnet Banner", "Ftp Banner", "Smtp Banner", "Http Header", "Http Doc.", "Trace Route", "Whois", and "Close".

Abaixo, um registro completo de log capturado pelo Specter:

Date: Mon, 31 Mar 2003 15:07:24 -0600
 From: SPECTER on OUTPOST
 To: lance@honeynet.org
 Subject: FTP connection (hacker.honeypots.com) - Attempt 1/2 (FTP/Total)

FTP connection from 192.168.1.1 (hacker.honeypots.com) (FTP attempts: 1, Total attempts: 2) on Mon Mar 31 15:07:46 2003

Date: Mon, 31 Mar 2003 15:08:07 -0600
 From: SPECTER on OUTPOST-01
 To: lance@honeynet.org
 Subject: FTP connection (hacker.honeypots.com) - Attempt 1/2 (FTP/Total)

FTP connection
 Host : 192.168.1.1 (hacker.honeypots.com)
 Login : anonymous
 Pass : evilattacker@hello.com
 Time : Mon Mar 31 15:07:46 2003

Log:
Client connecting: 192.168.1.1
Client tries anonymous Login
--->331 Guest login ok, send your complete e-mail address as password.
Client sent PASS 'evilattacker@hello.com'
--->230 User anonymous logged in.
Client sent SYST
--->215 UNIX Type: L8 (Linux)
Client changed type to I
--->200 Type set to I.
Client set port to 50739, IP to 192.168.1.1
--->200 PORT command successful.
Client wants to transfer file passwd
--->501 File not found.
Client closed connection
--->221 Goodbye.
Closing connection with 192.168.1.1

Valhala Honeypot

O Valhala Honeypot é um pote de mel desenvolvido completamente por mim, totalmente em português. Minha intenção ao criá-lo foi fornecer a funcionalidade da detecção de intrusos baseada em Honeypots de forma gratuita, simples e sem muita complexidade de implementação. De fato, verá que a configuração e utilização do Valhala é o processo mais simples possível. O programa foi desenvolvido para os sistemas Windows, mas pode rodar em distribuições Linux através da utilização do software de emulação Wine. Como todo software livre, o seu código está à disposição para ser estudado e melhorado por todos que assim o desejarem.

Primeiramente, deve ser entendido que o Valhala Honeypot possui tanto serviços de baixa e alta interação, como será apresentado. Relembrando: os de baixa interação permitem que o invasor descubra o honeypot mais rapidamente pois eles apenas rodam uma versão “emulada” do serviço. Já o de alta interação é difícil de ser descoberto, pois é um serviço totalmente funcional. Outra coisa, que ao contrário do gratuito honeyd e de outros honeypots comerciais, o Valhala ainda não suporta utilização de captura de IPs disponíveis na rede para a criação de hosts virtuais. Você deverá utilizar o endereço IP atual da máquina na qual instalará o programa.

Os serviços que o Valhala possui são:

- HTTP
- FTP
- SMTP

- POP3
- TELNET
- TFTP
- FINGER
- PROXY

Você verá como configurar devidamente cada um deles.

Antes de tudo, baixe a versão mais nova do valhala em
<http://valhalahoneypot.sourceforge.net> ou em
<http://www.sourceforge.net/projects/valhalahoneypot> .

Vamos conhecer então a interface principal do programa:



As opções existentes no menu principal são:

À direita, temos a tela que loga as tentativas capturadas de ataque. Sempre que uma conexão ou interação ocorrer com um serviço, ela será mostrada aqui.

Vamos conhecer a função dos botões à esquerda:

Monitorar – É onde deve ser clicado quando se deseja que o programa entre em operação. Assim que esse botão for ativado, todos os serviços configurados serão ligados e entrarão em atividade, aguardando as tentativas de acesso não autorizados. Sem clicar em monitorar, o Valhala não funcionará.

Parar – Irá interromper todos os serviços que estão em execução. O programa irá parar completamente de receber conexões e tentativas de ataque. Você pode querer parar por vários motivos, mas normalmente o faremos em uma situação específica. Vamos supor que o servidor de telnet foi reconfigurado e você quer que as novas alterações sejam lidas pelo programa. Você terá que clicar no botão parar e logo depois, no botão Monitorar novamente para que as novas configurações sejam lidas e adaptadas ao serviço. Lembrando que não é necessário clicar em parar para fechar o programa, o Valhala encerra os serviços ao ser finalizado.

Limpar – Limpa a tela de tentativas de ataque capturadas.

Salvar – Salva as tentativas de ataque em um arquivo de log, com o nome que você especificar.

Opções – Opções gerais de configuração do Valhala. Aqui existem várias coisas que pode ser ativadas, como: envio das tentativas por e-mail, modo oculto, portas extras, porta utilizada para o modo console e outras. Todas essas opções serão explicadas pouco mais à frente nesse capítulo.

Configurar – Esse é o local onde você deve vir para configurar os servidores. Todos os serviços utilizados pelo honeypot são configurados aqui. Não o confunda com o menu “opções”, que são apenas opções gerais de configuração.

Modo console – Essa opção é utilizada para realizar a centralização dos logs em um único servidor. Funciona assim: você instala o Valhala Honeypot em várias máquinas da rede e deixa uma delas como servidor de recebimento de logs de invasão. Em todas as outras, deve configurar (como você verá logo) o envio dos logs para o servidor através do menu opções. É um recurso muito útil que permite a monitoração de diversos computadores de uma vez só.

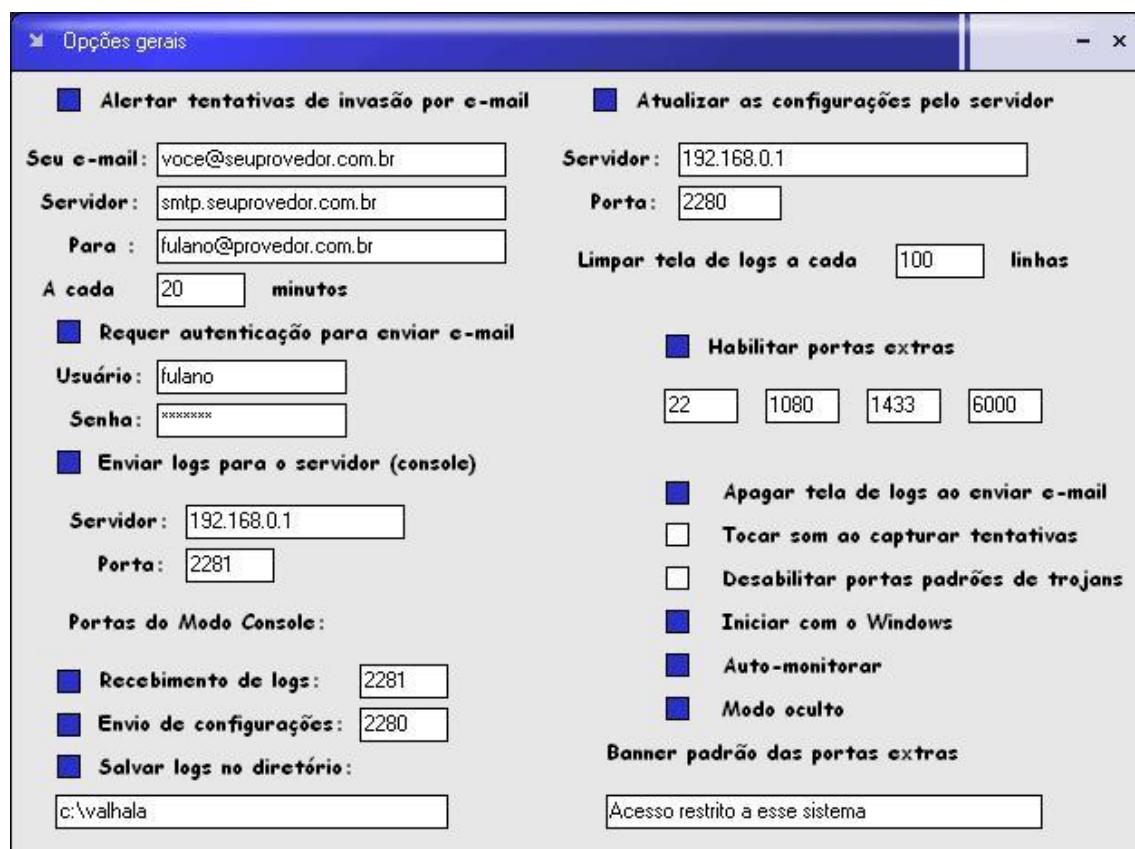
Sobre – Informações sobre o programa e o autor.

Ao clicar em monitorar, o comportamento padrão do programa é se esconder no system tray (bandeja do sistema). Para visualizá-lo novamente, clique com o botão direito do mouse no ícone do programa e em “Mostrar Honeypot”. A opção “Esconder Honeypot”, faz o contrário... oculta a janela do Valhala.



Menu Opções

Vamos conhecer o menu opções e entender cada uma das configurações possíveis:



Alertar tentativas de invasão por e-mail:

Envia os logs capturados na tela principal para um e-mail que você especificar a cada X minutos, que podem ser configurados. Para isso você deve utilizar uma conta de e-mail real.

Em *seu e-mail*, coloque o e-mail principal da conta.
Exemplo: mflavio2k@yahoo.com.br

Em *servidor*, especifique o servidor de correio para envio do e-mail.
Exemplo: smtp.pop.mail.yahoo.com.br

Em *para*, especifique a conta que irá receber o e-mail. Pode ser o mesmo e-mail ou não, depende de como você quer configurar. Exemplo:
mflavioaa@hotmail.com

Se o servidor SMTP (de envio de correio) necessitar autenticação, marque a caixa *Requer autenticação para enviar e-mail*. Coloque o nome do seu usuário e a senha.

Enviar os logs para o servidor:

Envia cada um dos logs capturados imediatamente para um servidor, o qual o endereço IP ou o nome deve ser configurado aqui, assim como a porta correta (caso não tenha alterado, deixa a padrão). Esse servidor que irá receber os logs deve estar com estar configurado corretamente e com o Modo Console habilitado para que consiga receber esses dados.

Portas do Modo Console:

É aqui que você irá configurar as portas que serão utilizadas quando o Modo Console for ativado. O Valhala abrirá essas portas para aguardar clientes que tentarão enviar logs ou buscar configurações. Existe duas sub-opções:

Recebimento de logs: É a porta utilizada para receber as tentativas de invasão de outros clientes que estão rodando na rede.

Envio de configurações: É a porta utilizada para que os clientes que desejam obter novas configurações podem se conectar. A explicação dessa opção é bem simples: suponha que você instalou o Valhala em dezenas de máquinas. Você configura essas cópias para se conectar em um servidor e buscar novas configurações. Assim, todos irão sempre buscar mudanças na configuração do programa no servidor, e você não precisará reconfigurar novamente o Valhala em cada computador.

Salvar logs no diretório:

Especifique um diretório onde os logs serão automaticamente salvos em arquivos diários. Esses arquivos contém um formato data + extensão log. Exemplo: 20-08-09.log É uma opção recomendada ao invés de se salvar os logs manualmente utilizando o menu principal do programa.

Atualizar as configurações pelo servidor:

Para as cópias do Valhala que você deseja que tenham as suas configurações automaticamente atualizadas com base em um computador servidor, especifique o endereço IP/nome e a porta desta máquina aqui. Assim, sempre que o programa for inicializado ele buscará novas configurações no servidor.

Limpar a tela de logs a cada X linhas:

Limpa o conteúdo da tela principal, onde são capturadas as tentativas de invasão. Caso você tenha configurado o envio das configurações por e-mail, servidor, ou mesmo para salvar os logs em disco, não tem necessidade de ficar eternamente com esses logs na tela principal. Aqui então você pode configurar que a cada X linhas capturadas com tentativas de ataque (exemplo, 100, 200), o conteúdo da tela será limpo.

Habilitar portas extras:

Permite habilitar portas extras para serem monitoradas. Para entender a vantagem dessas portas é primeiro necessário entender uma das principais fases de um processo de ataque: a varredura. O processo de varredura de portas visa encontrar e detectar portas de serviço ativas em um sistema. Portanto, sem ativar as portas extras o invasor irá descobrir apenas as portas de serviços padrão do Valhala. Se você quiser que ele “pense” que há outros serviços no sistema, você deve marcar a opção das portas extras e colocar o número da porta associada ao serviço.. (exemplo: 22 – ssh)

Apaga tela de logs ao enviar e-mail:

O conteúdo de toda a tela de logs é enviado por e-mail dentro do intervalo de tempo configurado no envio. Se você quiser que automaticamente após o conteúdo ser enviado, a tela de logs seja apagada, marque essa opção.

Tocar som ao capturar tentativas:

Se desejar um pequeno aviso sonoro quando uma tentativa de invasão ocorre, marque essa opção. Não é necessário placa de som, é utilizado o pcspeaker.

Desabilitar portas padrões de trojans:

É o oposto das portas extras. Por padrão, o Valhala abre portas de cavalos de tróia conhecidos (como o netbus na porta 12345), para que durante o processo de varredura o atacante pense que aquele trojan existe na máquina. Se você não quiser essas portas abertas, marque esta opção.

Iniciar com o Windows:

Marque essa opção se quiser que o programa se inicie junto com o Windows, sem precisar abri-lo manualmente. Essa opção deve ser marcada junto com auto-monitorar se você deseja que o programa já se inicie monitorando automaticamente.

Auto-monitorar:

O próprio nome já diz. Se marcar essa opção, ao abrir o Valhala ele já irá entrar em modo de monitoração automaticamente.

Modo oculto:

Por padrão, o Valhala sempre aparece na tela a cada tentativa de invasão, mesmo que você o oculte manualmente. Ao selecionar modo oculto, ele não irá aparecer e continuará rodando apenas no system tray (bandeja do sistema). Para visualizar os logs, você deverá clicar no tray e selecionar “Mostrar Honeypot”

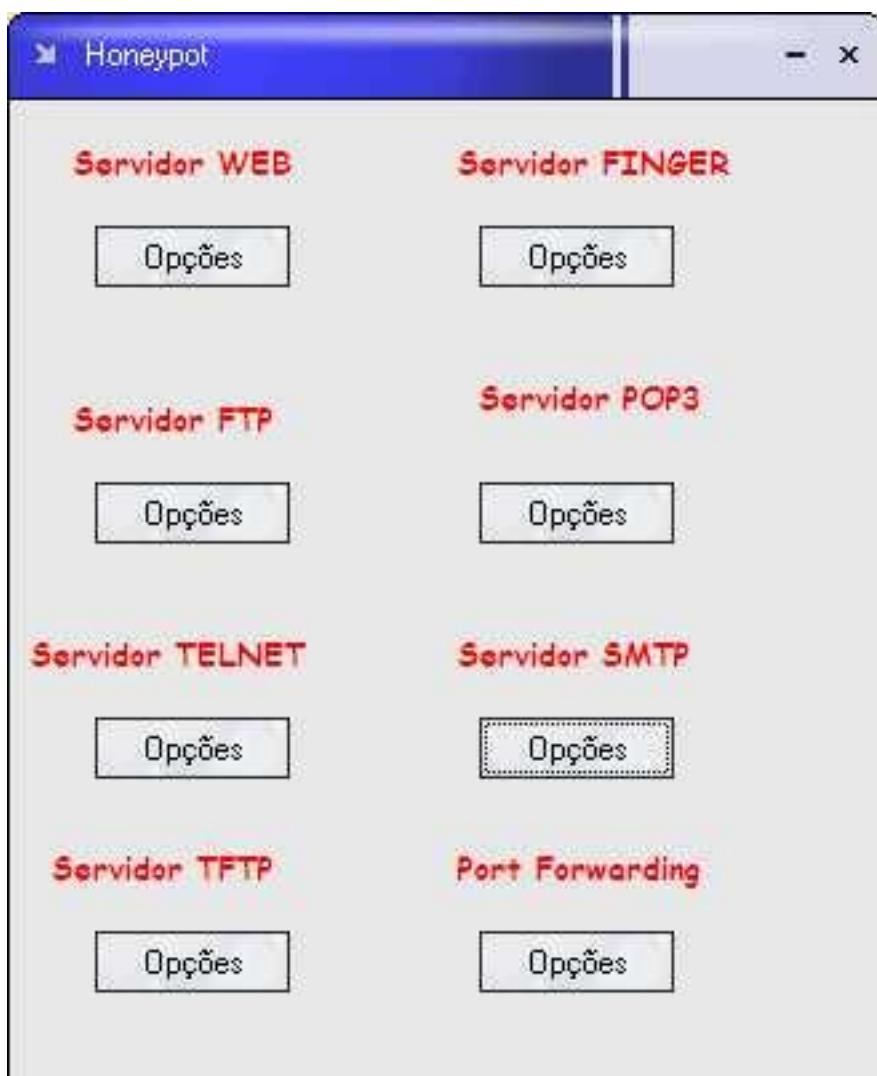
Banner padrão das portas extras:

Quando uma conexão for realizada por um invasor em uma das portas extras, ele espera receber um banner para tentar identificar o serviço. Esse processo é conhecido como enumeração. Aqui você pode colocar o que você quer que seja enviado quando alguém se conectar em alguma das portas.

Bom, agora que as opções gerais já são conhecidas, está na hora de entender a configuração dos servidores.

Menu Configurar

Em configuar que acessamos a configuração de todos os servidores. Eles estão divididos , como você pode ver na imagem a seguir:



Cada um dos botões “opções”, fornece acesso à configuração de cada servidor. É importante ressaltar que cada um dos serviços desse ser ativado na sua configuração. Mas como você pode fazer para saber qual serviço já está ou não ativo? É simples. Quando um serviço é ativado no Valhala Honeypot, o nome referente ao serviço passa da cor vermelha para verde, como pode ser visto :

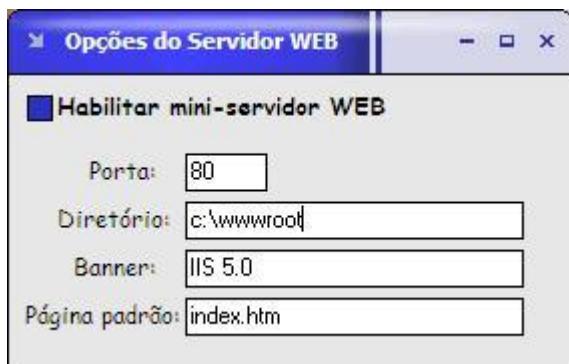


Bom, vamos aprender então sobre a configuração de cada um dos serviços do Valhala, como ativá-lo, sua função, e como ele interage com o invasor caso houver uma tentativa de acesso indevido à este serviço.

Servidor WEB

O servidor WEB permite ao invasor acessar páginas de hypertexto no computador que está rodando o Valhala Honeypot. Portanto, ele é um serviço

real, de alta interatividade e não apenas simulado. Entretanto, não possui um alto nível de risco ao ser utilizado. Para acessar as configurações do servidor web, acesse o menu Configurar / Servidor Web. A seguinte tela aparecerá:



Vamos às configurações:

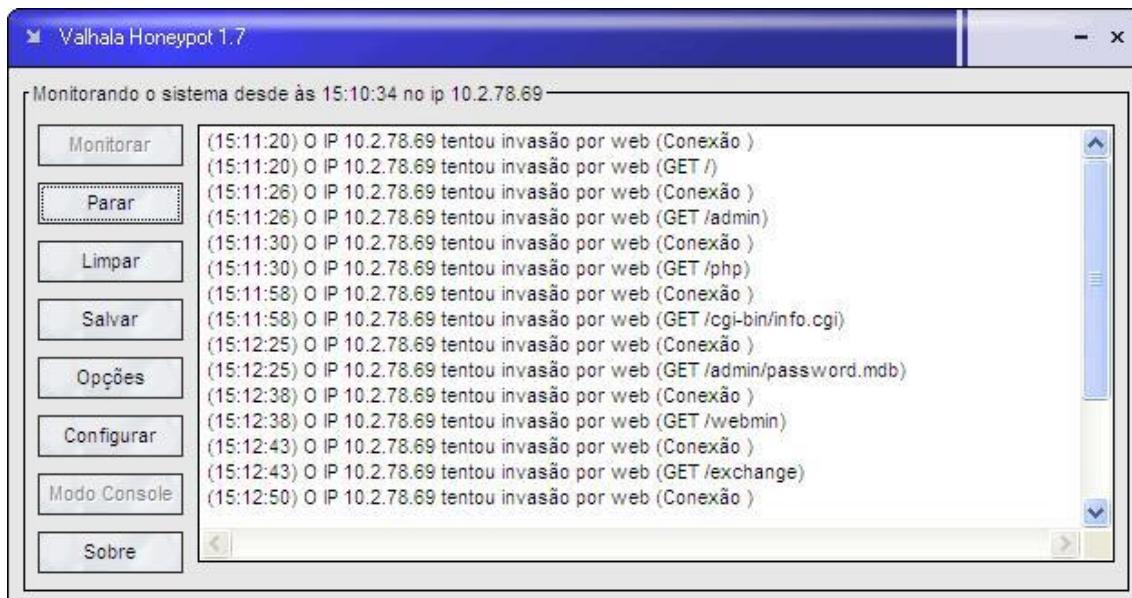
Porta: A porta que o servidor web irá utilizar. O padrão é a porta 80.

Diretório: O diretório onde existirão os arquivos do servidor web. Caso esse diretório não seja configurado corretamente ou não exista, o Valhala acusará um erro de que o diretório não existe.

Banner: Essa opção faz com que você engane o invasor, fazendo com que pense ter detectado o banner do servidor web. Exemplo: se você quiser que ele pense que é um servidor IIS da Microsoft, escreva IIS 6.0, por exemplo. Se desejar enganá-lo fazendo pensar que é o Apache, escreva: Apache 1.3.24 , e por aí vai.

Página padrão: É o arquivo principal que será carregado quando o usuário acessar o servidor web. Deve ser arquivo com extensão HTM e HTML (php e outros não é suportado). Por padrão, o nome é o index.htm, mas pode ser o que você quiser. Lembrando que: caso o arquivo não exista o Valhala irá acusar.

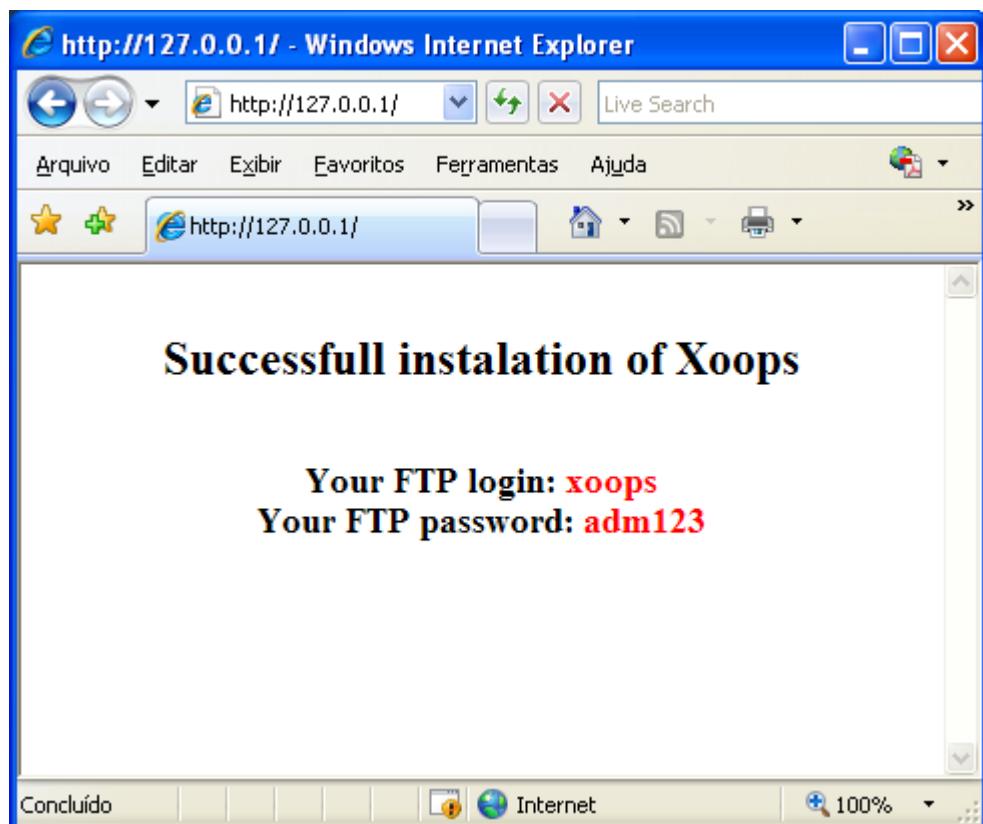
A monitoração do servidor WEB irá detectar todas as páginas que o invasor tentar acessar. Isso é comum de ser feito manualmente, ou através de um programa de detecção de falhas CGI, como Languard Network Scanner ou o Nessus. Assim que detectado conexões, o Valhala exibirá os logs como a seguir:



Esse é um claro caso onde o atacante está tentando detectar diretórios padrões de serviços e arquivos de configuração no nosso servidor WEB. Veja que o Valhala loga além do endereço IP de quem tentou realizar a invasão, mas também o horário, tipo de serviço e a informação que foi acessada naquele serviço.

Uma sugestão de utilização do servidor de páginas, é colocá-lo como uma espécie de Honeytoken. Crie uma página com alguns arquivos “esquecidos”, contendo senhas de acesso para outros serviços, como o FTP ou o POP3. Assim, se for um ataque real e não apenas um bot tentando acessar o serviço, ele com certeza tentará atacar os outros serviços com as credenciais que descobriu no servidor WEB.

Crie uma página web com um conteúdo parecido com o seguinte:



Com isso pensarão que o módulo Xoops (pode ser qualquer outro) foi instalado no sistema através de algum processo automatizado (como o Fantástico), e com certeza tentarão acessar o seu serviço FTP utilizando essas credenciais. Existem diversas outras maneiras de se criar honeytokens utilizando o servidor WEB. Basta apenas utilizar a sua imaginação, pois tudo depende da ilusão que pretende criar dentro de seu ambiente.

Servidor FTP

O FTP, ou protocolo de transferência de arquivos, é um serviço que vai permitir ao invasor pegar e colocar arquivos em seu computador. Talvez realizar outras ações, como apagar os arquivos. Por isso, ele é um serviço real de alta interatividade e deve ser utilizado com cuidado. O ideal é que quando você for utilizá-lo, rode-o dentro de uma máquina virtual, como já visto anteriormente.

Abaixo você vê as configurações do servidor FTP:



Vamos ao significado das opções:

Porta: É a porta padrão do servidor FTP. Por padrão é a porta 21. Tome cuidado pois na configuração do servidor Proxy, há uma opção para Proxy FTP, que utiliza a mesma porta. Se desejar utilizar os dois serviços juntos, deverá alterar a porta de um deles.

Banner: Nome do serviço que você quer que o invasor pense que você está rodando. Exemplo: WarFTPD, Wu-FtpD, ProFtpD, etc.

Login: Nome do usuário que terá acesso ao servidor FTP. Use algo mais fácil de ser descoberto, como root, FTP, etc. Isso claro, se você quiser que o invasor tenha acesso ao servidor FTP. Se quiser que ele fique apenas tentando descobrir o nome de usuário e a senha, coloque algo mais complexo.

Senha: A senha para o usuário definido na opção anterior. Novamente, se vai colocar uma senha complexa ou não depende do seu objetivo com o honeypot. Se desejar que o invasor a descubra rapidamente, tente algo como 12345 ou similar.

Diretório: O diretório onde ficarão os arquivos do servidor FTP. É nele que o invasor irá ter permissão para colocar e pegar arquivos. Nunca coloque um diretório do sistema ou algum que contenha arquivos importantes. Caso a configuração estiver errada o Valhala acusará que o diretório não existe.

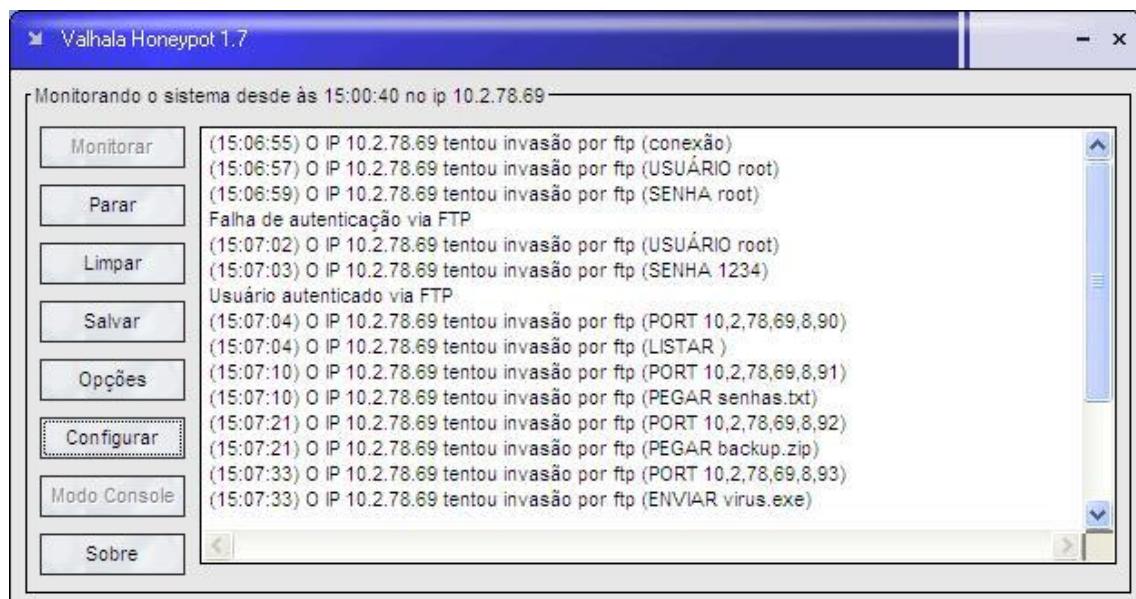
Após configurar corretamente o serviço deve-se apenas aguardar uma conexão de um invasor. Para exemplificar o que o atacante verá ao se conectar ao servidor, observe a imagem a seguir:

```

C:\>ftp 127.0.0.1
Conectado a 127.0.0.1.
220 Wu-ftpd 1.7.0
Usuário <127.0.0.1:(none)>: root
331 Password required for root.
Senha:
230 User root logged in.
ftp> dir
200 Port command successful.
150 Opening data connection for directory list.
drw-rw-rw- 1 ftp      ftp          0 Feb 23 16:18 .
drw-rw-rw- 1 ftp      ftp          0 Feb 23 16:18 ..
-rw-rw-rw- 1 ftp      ftp          34 Feb 23 16:18 index.htm
226 File sent ok
ftp: 180 bytes recebidos em 0,00Segundos 180000,00Kbytes/s.
ftp>

```

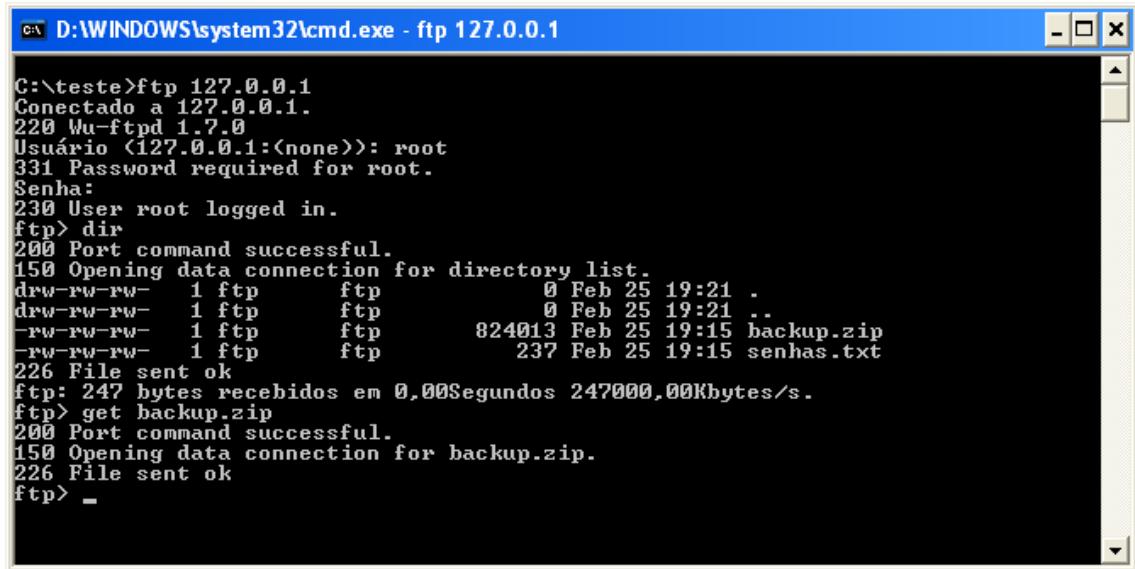
No exemplo acima, o invasor se conectou ao servidor FTP, recebeu o banner falso (Wu-ftpd), digitou o nome de usuário e a senha (usuário root) e listou os arquivos disponíveis no servidor com o comando “dir”. Observe agora como o Valhala captura ataques referentes ao serviço de FTP:



Essa é uma situação típica em um ataque real, mas nesse caso o invasor já sabia o nome de usuário e a senha antecipadamente (pode ter obtido essa informação no servidor web, como exemplifiquei anteriormente). Se o atacante não souber o nome de usuário ou a senha, ele pode tentar uma ataque de força-bruta, o que levaria o Valhala a detectar várias tentativas de login seguidas.

Uma das coisas interessantes na qual o servidor FTP pode ser utilizado, é para tentar executar alguma coisa no computador do invasor. Vou explicar melhor: imagine que você quer realizar uma espécie de contra-ataque, ou quer

simplesmente executar algum programa no computador dele com um fim específico. Nesse caso o ideal é colocar um arquivo dentro do diretório do FTP com um nome que seja atrativo o suficiente para o invasor querer puxá-lo e rodar no seu computador. Imagine que o atacante encontre o seguinte arquivo “**backup.zip**” no diretório:



```
C:\teste>ftp 127.0.0.1
Conectado a 127.0.0.1.
220 Wu-ftp 1.7.0
Usuário <127.0.0.1:<none>>: root
331 Password required for root.
Senha:
230 User root logged in.
ftp> dir
200 Port command successful.
150 Opening data connection for directory list.
drw-rw-rw- 1 ftp      ftp          0 Feb 25 19:21 .
drw-rw-rw- 1 ftp      ftp          0 Feb 25 19:21 ..
-rw-rw-rw- 1 ftp      ftp        824013 Feb 25 19:15 backup.zip
-rw-rw-rw- 1 ftp      ftp         237 Feb 25 19:15 senhas.txt
226 File sent ok
ftp: 247 bytes recebidos em 0,00Segundos 247000,00Kbytes/s.
ftp> get backup.zip
200 Port command successful.
150 Opening data connection for backup.zip.
226 File sent ok
ftp> _
```

Ele provavelmente irá baixar esse arquivo e descompactá-lo para ver o que tem dentro. A palavra “backup” tem uma força grande pois parece se tratar de algo importante, afinal ninguém faria uma cópia de segurança de dados irrelevantes. Ao abrir o arquivo, o invasor poderia se deparar um um outro arquivo de nome “**backup-selfextract.exe**”



Isso levaria os atacantes menos experientes a acreditar se tratar um arquivo “duplamente compactado” para a diminuição do tamanho. Primeiramente com a opção de auto-extração, que gera um executável, e

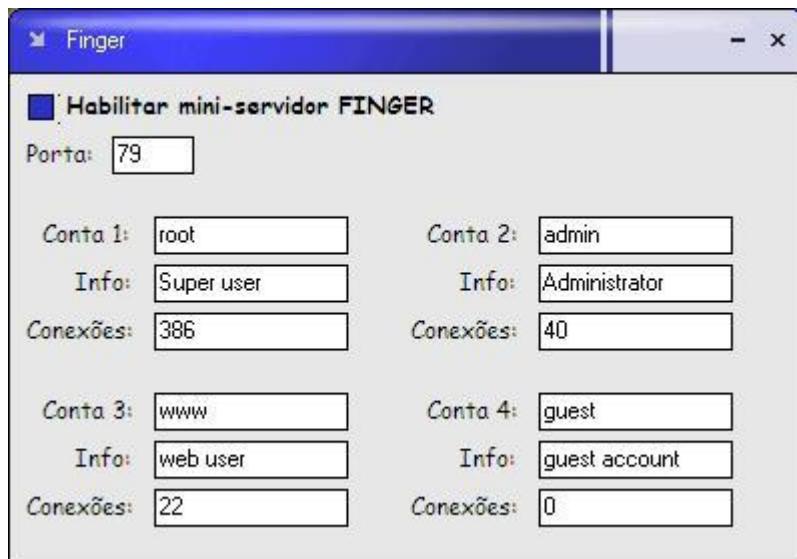
depois com o formato zip tradicional. Portanto, a maioria cairia na armadilha e executaria o programa.

Servidor Finger

O serviço Finger é o primeiro que abordamos que é de baixa interatividade. Em outras palavras, ele é simulado e não fornece informações reais do sistema. Seu objetivo de uso é devido ao fato de que o Finger é um dos clássicos serviços que normalmente sempre são checados ao se tentar um ataque, especialmente em sistemas Unix-like. Era um servidor muito disseminado antigamente, mas devido a problemas de segurança hoje ele é pouco utilizado. Ainda assim, muitos garotos que estão começando agora a ler materiais antigos sobre hacking, aprendem sobre o Finger e procuram por servidores na internet com este serviço habilitado.

Mas então, o que exatamente o finger faz? Ele permite realizar uma consulta de quais usuários estão conectados ao servidor e também pegar informações sobre um usuário específico, conhecendo dados como: nome do usuário, quantas vezes o usuário se logou no sistema, etc.

Abaixo, você vê a tela de configuração do Finger:



As opções são as seguintes:

Porta: a porta utilizada no serviço. Por padrão é a 79.

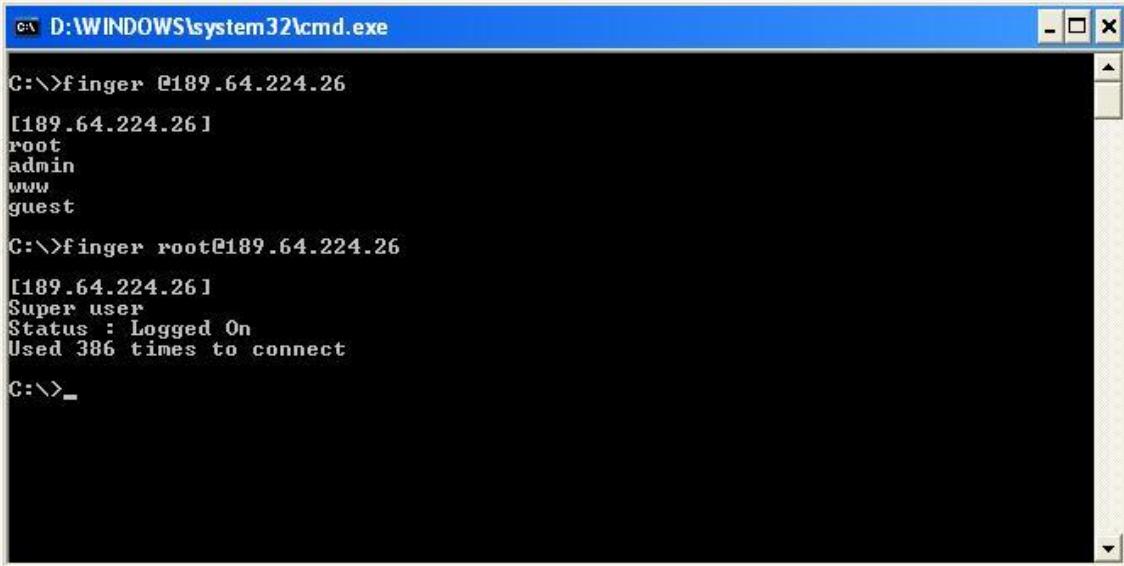
Todos os quatro usuários simulados possuem as mesmas opções de criação, são elas:

Conta: o nome da conta do falso usuário

Info: Informações sobre a conta, tal como o nome completo do usuário

Conexões: quantas vezes ele se conectou ao sistema

Para simular uma tentativa real, a imagem a seguir mostra o que o invasor iria enxergar ao tentar realizar um finger contra o computador rodando o Valhala Honeypot.

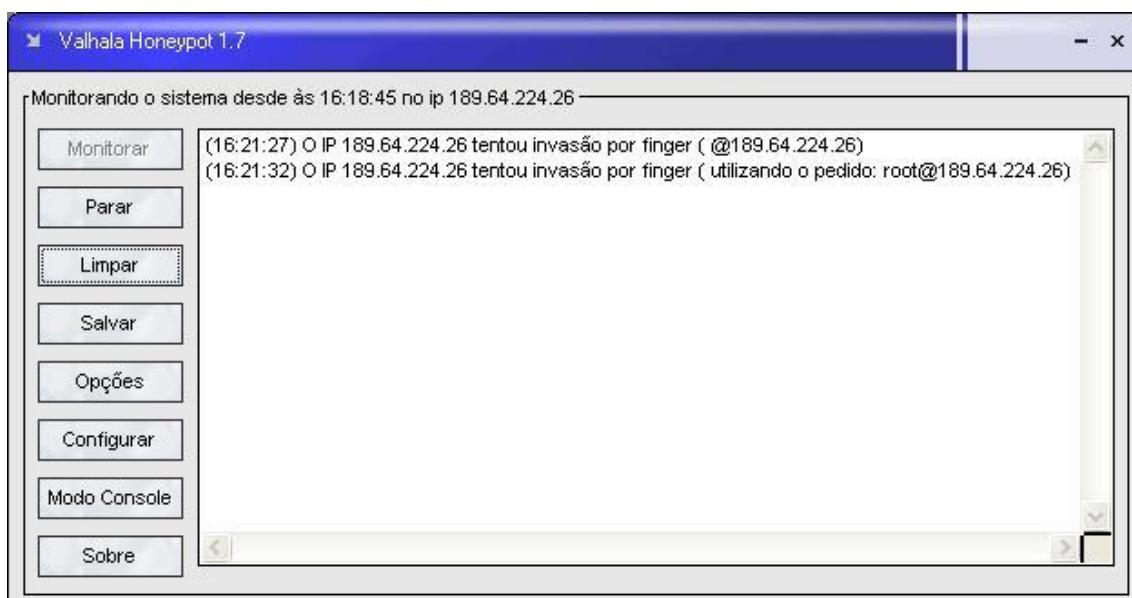


```
D:\WINDOWS\system32\cmd.exe
C:\>finger @189.64.224.26
[189.64.224.26]
root
admin
www
guest

C:\>finger root@189.64.224.26
[189.64.224.26]
Super user
Status : Logged On
Used 386 times to connect

C:\>_
```

E essas informações seriam detectadas automaticamente no honeypot como demonstrado a seguir:



No caso do Finger, não há muita coisa a ser feita em relação a honeytokens. Talvez apenas utilizar nomes mais comuns nos usuários, como João, Ana, Marcos, etc. Pois assim, um possível invasor tentaria o nome desses usuários para tentar acessar o servidor FTP, por exemplo. Mas é claro, ainda teria que descobrir a senha.

Servidor POP3

O serviço POP é o responsável pelo recebimento de mensagens de e-mail. Ele também é de baixa interatividade, ou seja, não fornecerá acesso a mensagens verdadeiras que estejam armazenadas no computador. O objetivo desse serviço é enganar o invasor incitá-lo a tentar descobrir a senha de algum e-mail que ele pensa estar armazenado no servidor.

As configurações do Servidor POP3 são as mostradas a seguir:



Vamos às opções:

Porta: A porta que o servidor POP3 irá operar. O padrão é a 110.

Banner: O “falso” software de servidor que o invasor pensará estar acessando.

Login: O nome do usuário que terá acesso ao servidor POP3. Lembre-se de colocar um nome de fácil descoberta caso queira que o invasor acesse o serviço leia a mensagem falsa.

Senha: A senha de acesso do usuário que deverá ser utilizada para se logar no servidor.

Agora, as opções referentes à mensagem de e-mail falsa. Quando o atacante conseguir se logar no servidor ele receberá esse falso e-mail como se fosse uma mensagem real, verdadeira.

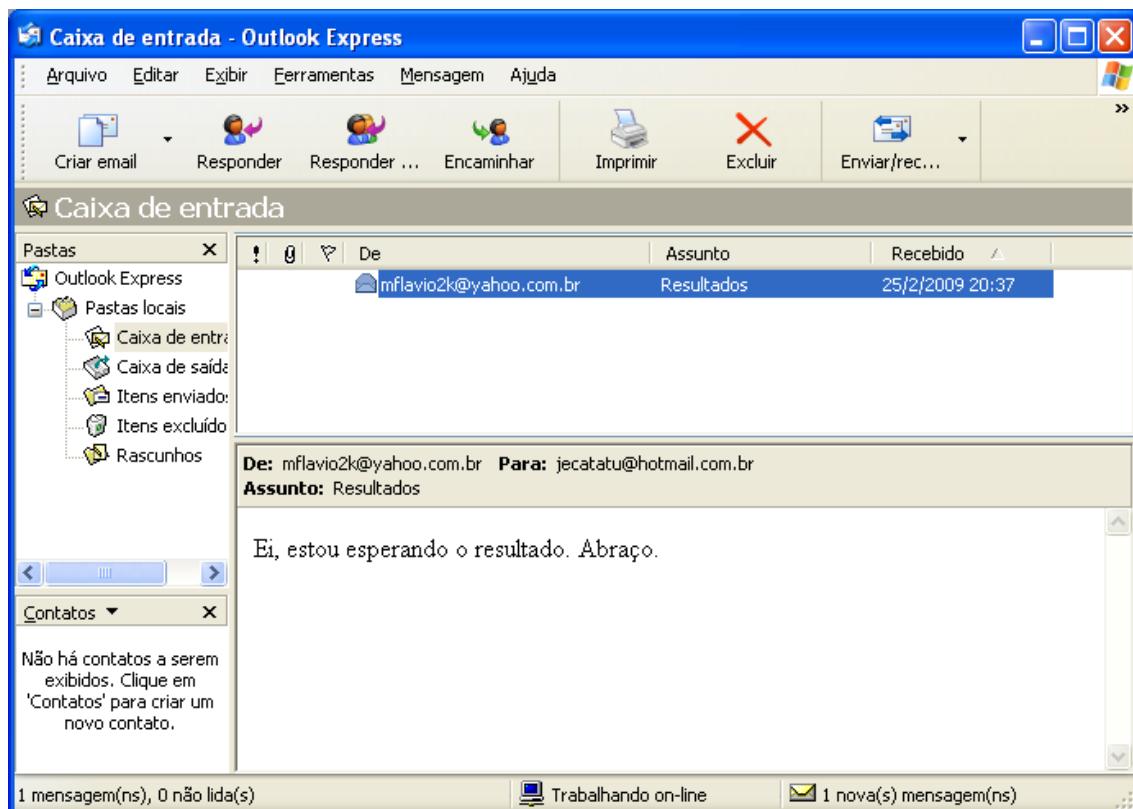
De: De qual e-mail a mensagem falsa parecerá estar se originando

Para: Para qual e-mail a mensagem falsa parecerá estar se destinando

Assunto: O assunto da falsa mensagem.

Texto no corpo da mensagem: Escreva um texto nesse campo que será mostrado como o conteúdo da mensagem. Apesar de ter apenas uma linha, você pode formatar a mensagem usando caracteres de quebra de linha.

Suponhendo que um invasor consiga ter acesso ao servidor POP3, ele poderia ler a mensagem falsa utilizando um cliente de e-mails qualquer, como por exemplo o Outlook Express:



Agora, observe como os acessos realizados ao serviço POP3 são capturados através da tela do Valhala Honeypot:



O melhor tipo de honeytoken para se utilizar no servidor POP3 é concentrar-se na mensagem falsa. Você pode colocar no conteúdo da mensagem falsa tudo o que achar que fará o atacante se aprofundar ainda mais no sistema. Algumas sugestões do que o texto poderia conter:

- O nome de usuário e senha de outro serviço, como telnet ou FTP. O motivo explicado no e-mail seria que o dono havia se esquecido de suas credenciais e pediu para que fossem enviadas.

- Uma referência ao arquivo “backup.zip”, o honeytoken que citamos anteriormente no servidor FTP. Algo falando que os dados guardados ali são muito importantes e que é para ter cuidado. Se o invasor ainda estava em dúvida se devia ou não baixar este arquivo, ele definitivamente o fará agora.

- Ou mesmo, se você não ligar para o seu honeypot ser comprometido, deixe um e-mail para o atacante parabenizando-o por chegar até ali. Seria divertido, com certeza.

Servidor SMTP

Esse é o servidor responsável pelo envio de e-mail. Também é um serviço que se utiliza de baixa interação, especialmente para evitar que os spammers se utilizem do Valhala. Ele mostra que um e-mail foi enviado; apenas essa mensagem nunca chegará ao seu destino. Também , é um dos serviços que contém a configuração mais simples

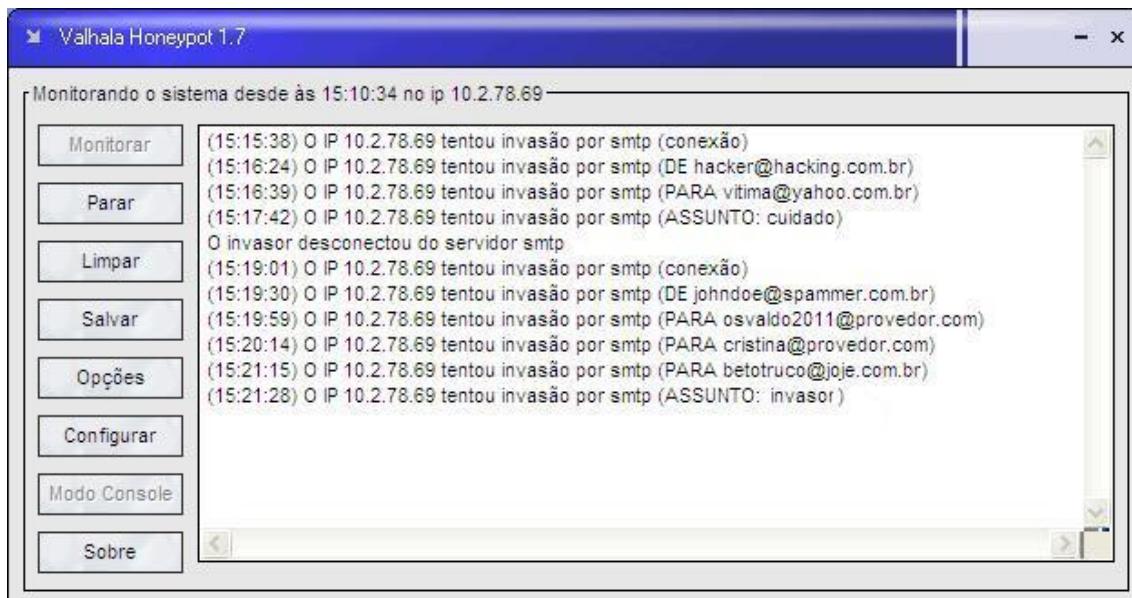


Opções:

Porta: a porta do servidor SMTP. A padrão é a porta 25.

Banner: “Falso” serviço que o invasor pensará estar utilizando.

A configuração do SMTP é realmente muito simples. Toda vez que alguém tentar utilizá-lo para enviar um e-mail, você receberá o aviso no Valhala Honeypot:

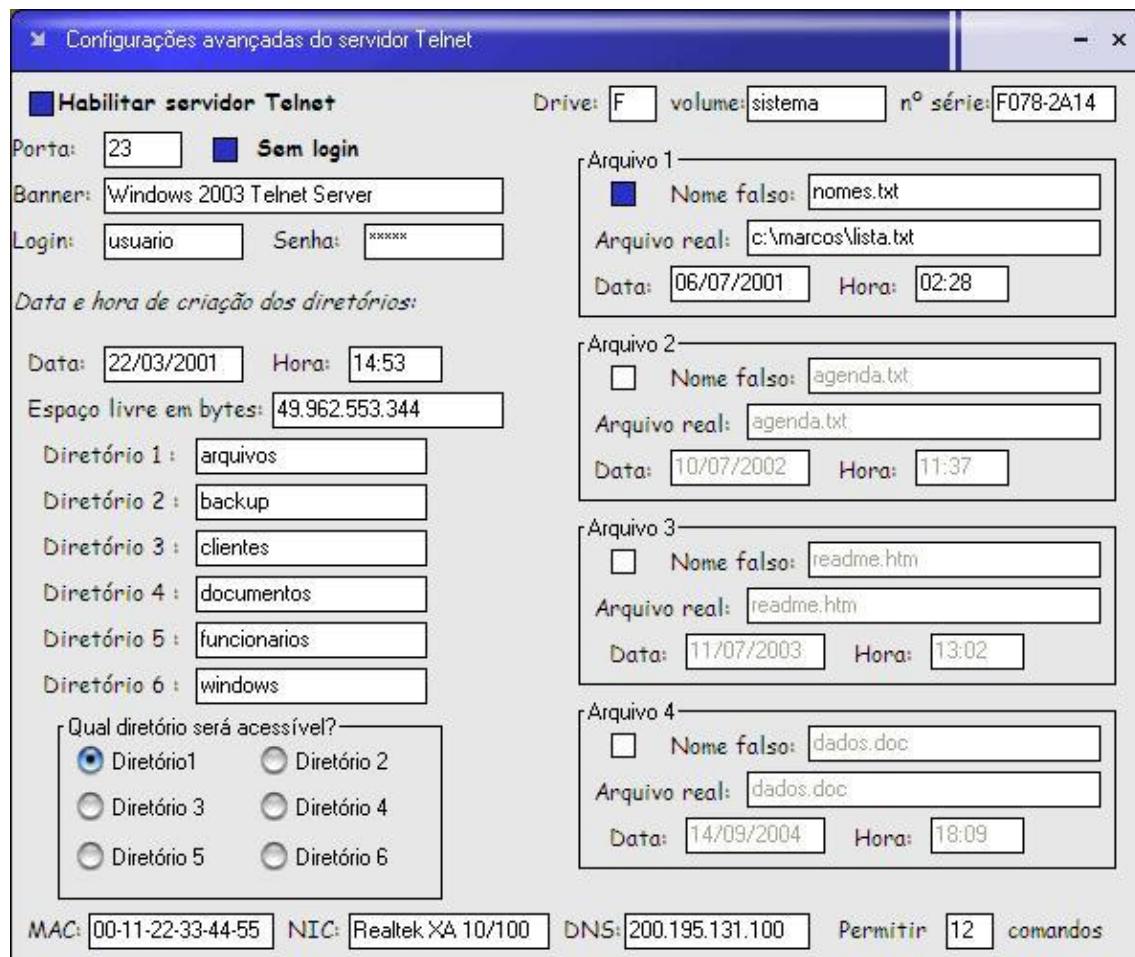


Nesse caso, não há como tentar criar nenhum tipo de Honeytoken, pois a simplicidade de configuração do serviço não permite.

Servidor telnet

Esse particularmente é o meu serviço favorito no Valhala. Ele dá acesso a um Shell de comandos do MS-DOS para o invasor. Mas não se engane pois ele não é um serviço de alta interação, e sim de baixa. Todos os comandos são simulados e criados para tentar interagir o melhor possível com o usuário. Claro que um ambiente destes pode ser facilmente descoberto por um invasor com um pouco mais de experiência, mas oferece também uma grande vantagem: a segurança de que a máquina não poderá ser comprometida, pois o acesso ao shell real nunca será utilizado.

Abaixo , a tela de configuração do servidor de telnet:



Apesar do servidor de telnet possuir um Shell simulado, ele possui a vantagem de poder ser totalmente personalizado. Você poderá alterar praticamente tudo no ambiente, como veremos. Vamos às opções:

Porta: A porta em que o servidor telnet irá operar. A padrão é a 23.

Sem login: Se estiver marcada essa opção, o invasor cairá no Shell simulado sem precisar entrar com o nome de usuário e senha

Banner: Informação falsa passada para o atacante quanto ao tipo de software de servidor utilizado

Login: O nome do usuário do servidor telnet

Senha: A senha para o usuário configurado acima.

Agora para as opções de simulação do Shell:

Drive: A letra do drive de disco simulado. Pode ser C:, D:, etc. Configure da forma que achar mais interessante.

Volume: O nome do volume do disco rígido.

Nº de série: Configure o número de série para o disco rígido.

Data: A data que será mostrada quando os diretórios forem listados. É a data de suposta criação dos diretórios.

Hora: A hora que será mostrada quando os diretórios forem listados. É a hora da suposta criação dos diretórios.

Espaço livre em bytes: É o espaço livre disponível no disco. Lembre-se que ele deve ser expresso em bytes, pois é assim que é mostrado no shell real do MS-DOS.

Diretório: Essa opção se repete 6 vezes por um simples motivo. O Shell simulado mostra seis diretórios, os quais você definirá os nomes preenchendo essas opções. Uma coisa muito importante é colocar os diretórios na ordem ALFABÉTICA para não gerar suspeitas.

Qual diretório será acessível: O próprio nome diz: qual dos seis diretórios criados anteriormente será acessível. Quando o atacante tentar entrar nele, conseguirá sem problemas. Para todos os outros ocorrerá um erro de acesso negado.

MAC: O endereço físico (MAC) da placa de rede que será mostrado quando o invasor digitar o comando ipconfig.

NIC: O nome da placa de rede que será mostrado quando o invasor digitar o comando ipconfig.

DNS: O endereço IP do servidor DNS, também mostrado quando o invasor digitar ipconfig.

Permitir X comandos: Essa opção permite desconectar automaticamente o invasor após ele ter digitado um certo número de comandos. Especifique um número maior se deseja que ele permaneça mais tempo nos servidores telnet. Ou um número menor se não quer dar chance a ele de descobrir que é um ambiente simulado.

Arquivos: Essas quatro opções permitem colocar arquivos reais (de texto apenas) para ter o seu conteúdo exibido com o comando type. Estarão localizados dentro do diretório que o invasor poderá ter acesso (opção configurada anteriormente). Para ativar, deve primeiramente marcar uma das caixas disponíveis. As sub-opções são

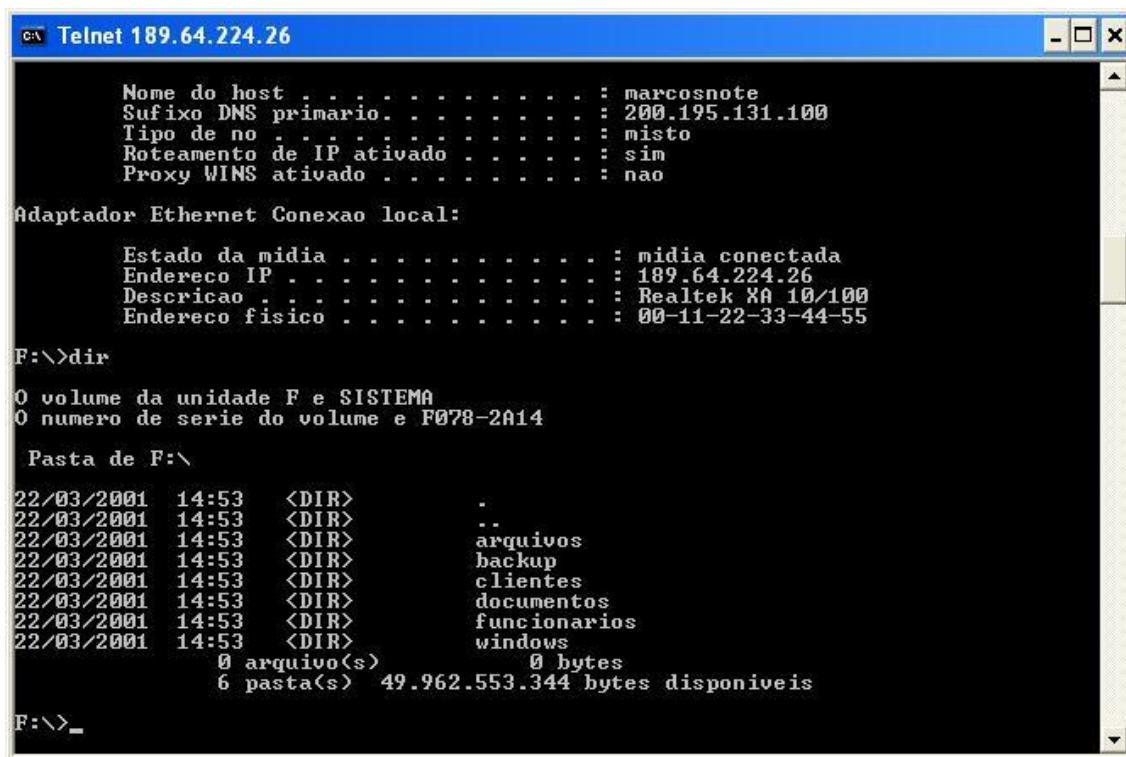
Nome falso: O nome que será criado no ambiente simulado para o arquivo.
Exemplo: **usuarios.txt**

Arquivo real: O caminho completo da localização do arquivo real no disco.
Exemplo: **c:\teste\texto.txt**

Data: A data de criação do arquivo simulado.

Hora: A hora de criação do arquivo simulado.

Abaixo visualizamos como o invasor enxergará o shell simulado. Perceba que a primeira vista dá para realmente confundir com a versão real do prompt de comandos do MS-DOS.



The screenshot shows a Telnet session titled "Telnet 189.64.224.26". The window displays the following information:

```
Nome do host . . . . . : marcosnote
Sufixo DNS primario . . . . . : 200.195.131.100
Tipo de no . . . . . : misto
Roteamento de IP ativado . . . . . : sim
Proxy WINS ativado . . . . . : nao

Adaptador Ethernet Conexao local:
Estado da media . . . . . : media conectada
Endereco IP . . . . . : 189.64.224.26
Descricao . . . . . : Realtek XA 10/100
Endereco fisico . . . . . : 00-11-22-33-44-55

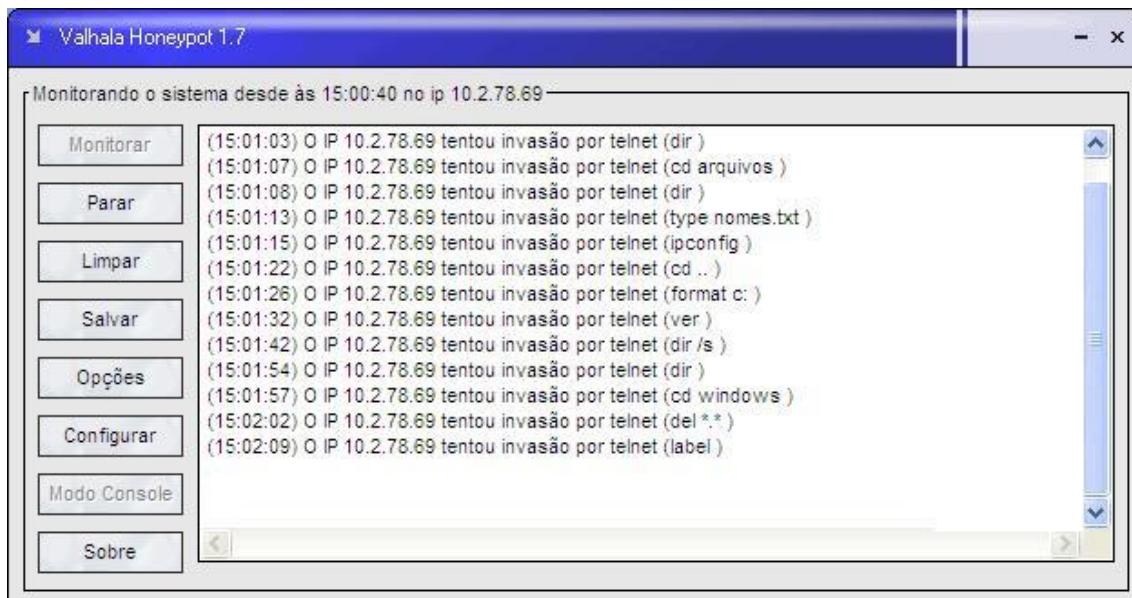
F:\>dir
O volume da unidade F e SISTEMA
O numero de serie do volume e F078-2A14

 Pasta de F:\

22/03/2001 14:53 <DIR> .
22/03/2001 14:53 <DIR> ..
22/03/2001 14:53 <DIR> arquivos
22/03/2001 14:53 <DIR> backup
22/03/2001 14:53 <DIR> clientes
22/03/2001 14:53 <DIR> documentos
22/03/2001 14:53 <DIR> funcionarios
22/03/2001 14:53 <DIR> windows
          0 arquivo(s)        0 bytes
       6 pasta(s)  49.962.553.344 bytes disponiveis

F:\>_
```

E claro, o Valhala irá capturar em sua tela principal todos os comandos digitados no ambiente simulado. É um recurso interessante pois você pode saber antecipadamente o que esse invasor em particular faria se tivesse tido acesso a um Shell de comandos real. Ele tentaria apagar alguma coisa? Copiar um arquivo? Veja na imagem a seguir:



A melhor sugestão para a utilização de honeytokens com o servidor telnet é a criação de diretórios com nomes relevantes. Pastas com o nome de “**salarios**”, “**banco**” ou mesmo “**funcionarios**”, podem levar o atacante a pensar que se trata de um computador de um departamento de recursos humanos, por exemplo. Você ainda poderá avançar mais na farsa ao permitir acesso ao diretório **funcionários** e dentro dele adicionar um arquivo falso com uma suposta lista de pessoas, com o nome por exemplo de **contratados.txt**. Enfim, como em todos os outros serviços, a imaginação é o limite quando se trata de Honeytokens.

Servidor TFTP

Assim como o FTP, o TFTP é um serviço de alta interação, permitindo transferência real de arquivos entre máquinas distintas. Mas ao contrário de todos os outros serviços no Valhala que utilizam o protocolo TCP para transporte, ele utiliza o protocolo UDP, que não é orientado a conexão. Isso faz com que o TFTP seja um serviço considerado inseguro, já que não exige nome de usuário ou senha para se conectar a um sistema. Mas por outro lado, não há como listar os arquivos disponíveis em um servidor TFTP.

De fato, só dá para se realizar duas ações: PEGAR ou COLOCAR um arquivo.

Vejamos as configurações do servidor TFTP:



É o servidor que possui a configuração mais simples de todos. Vejamos:

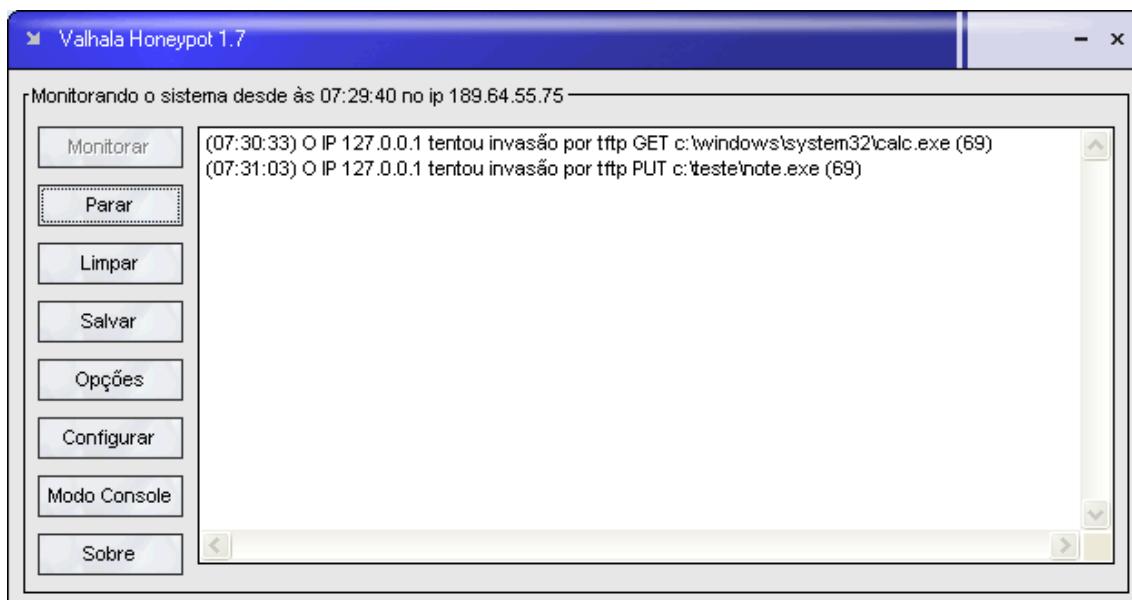
Porta: É a porta que o servidor TFTP atua. O padrão é a 69 UDP (ao contrário dos outros serviços que utilizam todos portas TCP)

Modo funcional: Se essa opção estiver marcada, o atacante vai ter a capacidade de fazer o download de TODO e QUALQUER arquivo da máquina. Assim como substituir qualquer arquivo por upload. Só marque essa opção se estiver montando uma Honeynet virtual, ou seja, em um ambiente voltado apenas para servir de Honeypot. Se estiver rodando o Valhala em seu computador pessoal, desmarque essa opção e isso fará que o atacante seja impedido de realizar uploads e downloads.

A seguir, temos uma visão de um atacante se comunicando com o servidor TFTP. Veja que o comando é muito simples, com uma só linha é possível pegar ou colocar arquivos no servidor:

```
C:\>tftp -i 127.0.0.1 GET c:\windows\system32\calc.exe
Transferência bem sucedida: 114688 bytes em 1 segundo, 114688 bytes/s
C:\>tftp -i 127.0.0.1 PUT c:\windows\system32\notepad.exe c:\teste\note.exe
Transferência bem sucedida: 69120 bytes em 1 segundo, 69120 bytes/s
C:\>
```

Na próxima imagem, você pode visualizar os logs de captura do Valhala referentes a transferência dos arquivos no servidor TFTP:

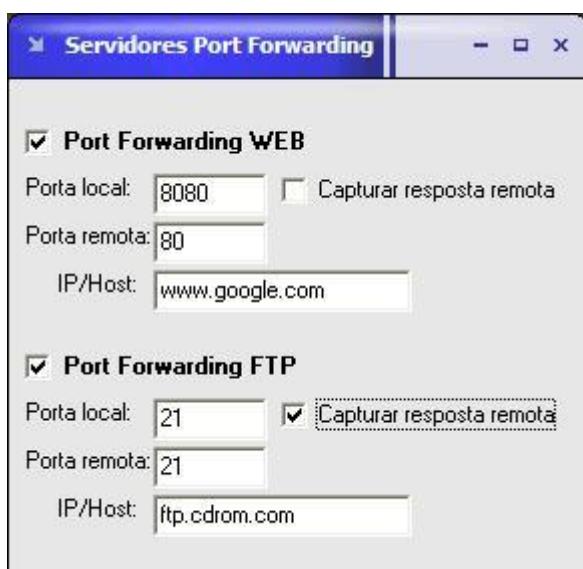


Infelizmente, assim como no SMTP, o TFTP é tão simples que não permite a criação de honeytokens. A única liberdade que você tem é: permitir ao invasor acesso a todos os seus arquivos, ou não.

Servidor Proxy

Por último, o Valhala apresenta também um serviço simples de Proxy. Funciona mais como um redirecionamento. O invasor conecta-se em uma porta e você o direciona para um determinado endereço IP , na porta que desejar. Portanto, o serviço de Proxy é de alta interatividade.

Abaixo, veja as configurações do servidor Proxy:



Inicialmente, existem dois tipos de redirecionamento (Port Forwarding). O primeiro, é baseado em WEB. Ou seja, irá redirecionar o invasor para uma página de sua escolha quando ele tentar acessar a porta do Proxy. O segundo é baseado em FTP, e irá redirecionar o atacante para outro servidor de transferência de arquivos quando ele se conectar na porta.

Em ambos os casos as opções são as mesmas, vamos conhecê-las:

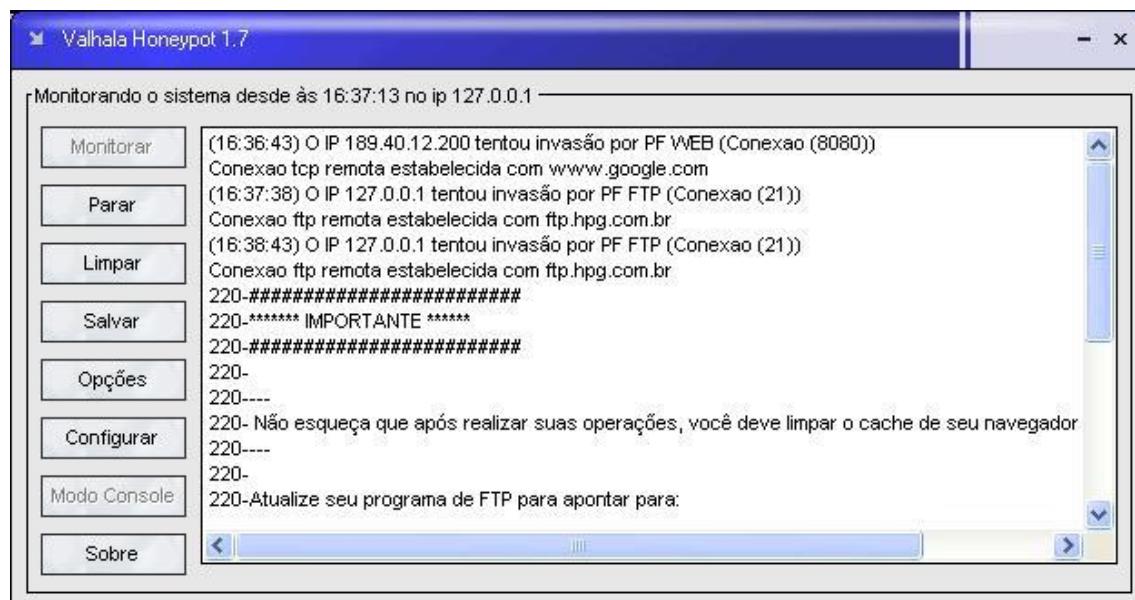
Porta local: É a porta do Proxy que será aberta no seu computador. É essa porta que o invasor irá se conectar. O padrão para Proxy web é a 8080 e para o Proxy FTP é 21 (atenção: o Proxy FTP usa a mesma porta que o servidor FTP normal. Cuidado para não habilitar os dois sem antes alterar a porta de um deles).

Porta remota: É a porta na qual o invasor será direcionado no sistema remoto. Em outras palavras, é a porta que deverá ser acessada no destino.

IP/HOST: É o endereço IP ou nome do servidor remoto para onde o atacante será redirecionado.

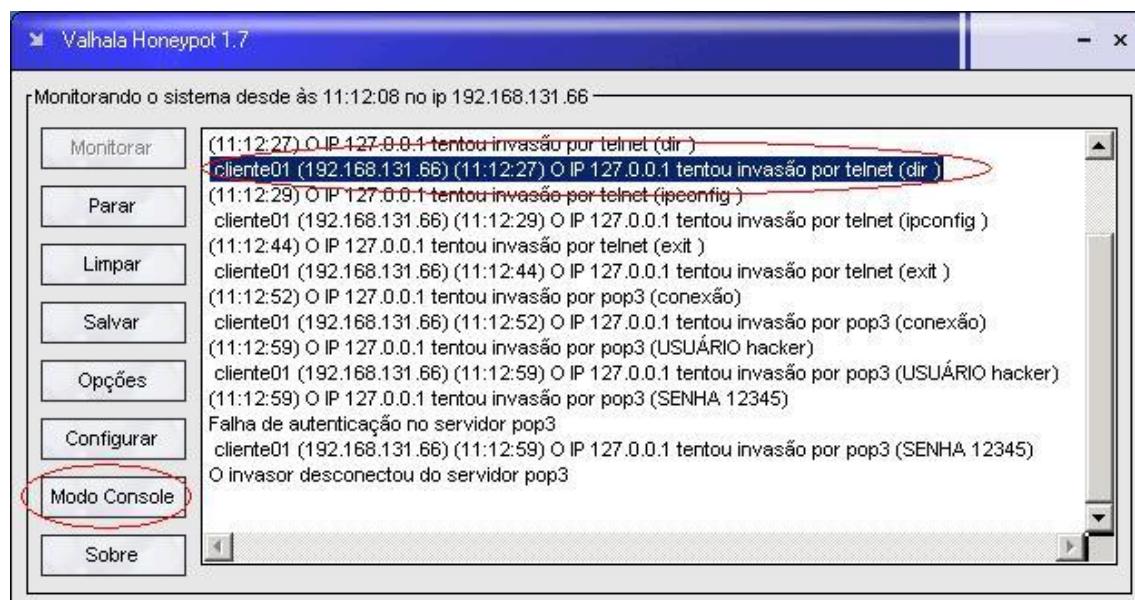
Capturar resposta remota: Se essa opção for marcada, o Valhala capturará não só as tentativas de acesso ao servidor remoto (o padrão), como também irá tentar capturar o tráfego resultante (página acessada, sessão do servidor FTP remoto, etc).

Veja na imagem a seguir a captura do Valhala de tentativas de acesso ao servidor Proxy, tanto o Proxy Web quanto o Proxy FTP:



Modo Console

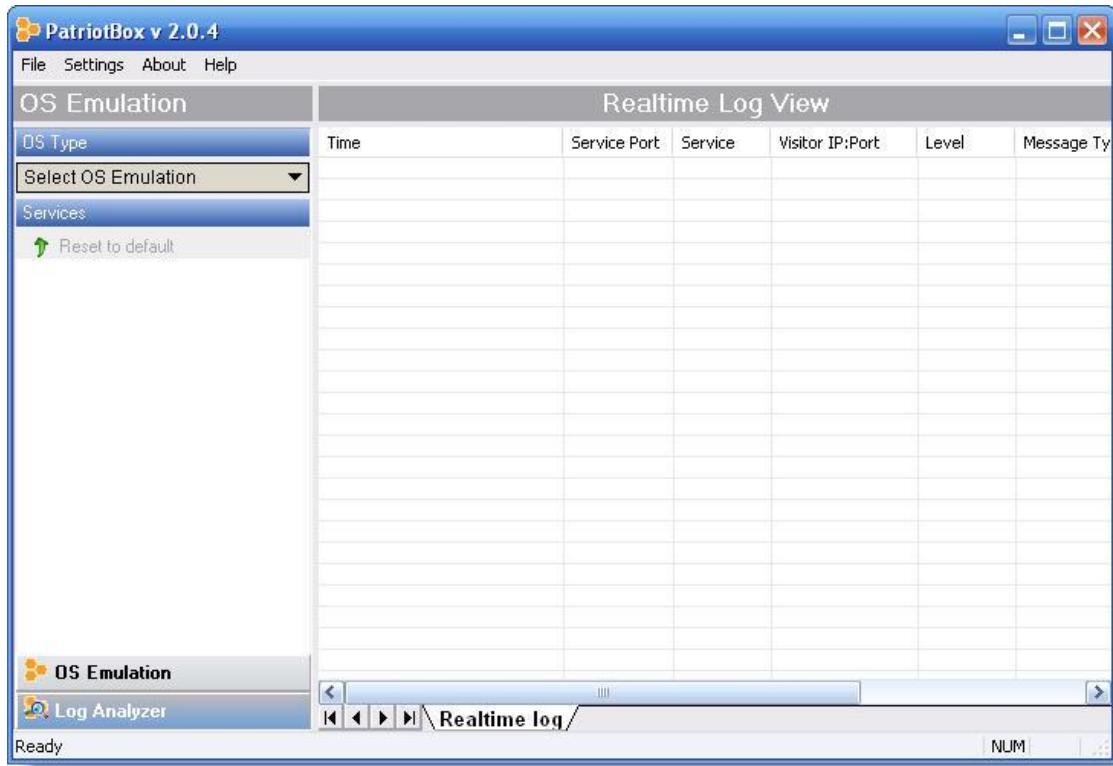
Como citei anteriormente o modo console é uma maneira de permitir que o Valhala atue como servidor de centralização de logs. Portanto, se você ativar esse modo no computador que atuar como servidor, e configurar todos os clientes para enviar as tentativas de acesso, ele irá capturar os dados mostrando o nome e o endereço IP do computador da rede que enviou o log, além da tentativa de ataque ... como mostrado na linha marcada da imagem a seguir:



PatriotBox

O PatriotBox é um Honeypot comercial para Windows, que possui diversos recursos bem interessantes. A empresa que o desenvolveu é a Alkasis (www.alkasis.com). Uma das vantagens é que você pode aumentar a funcionalidade do PatriotBox baixando novos plugins e scripts. Ele também possui suporte para utilizar scripts do Honeyd. O funcionamento deste honeypot é relativamente simples.

O primeiro passo é baixá-lo no site do fabricante. Após rodar a instalação, você verá a seguinte tela, que é a principal do programa:



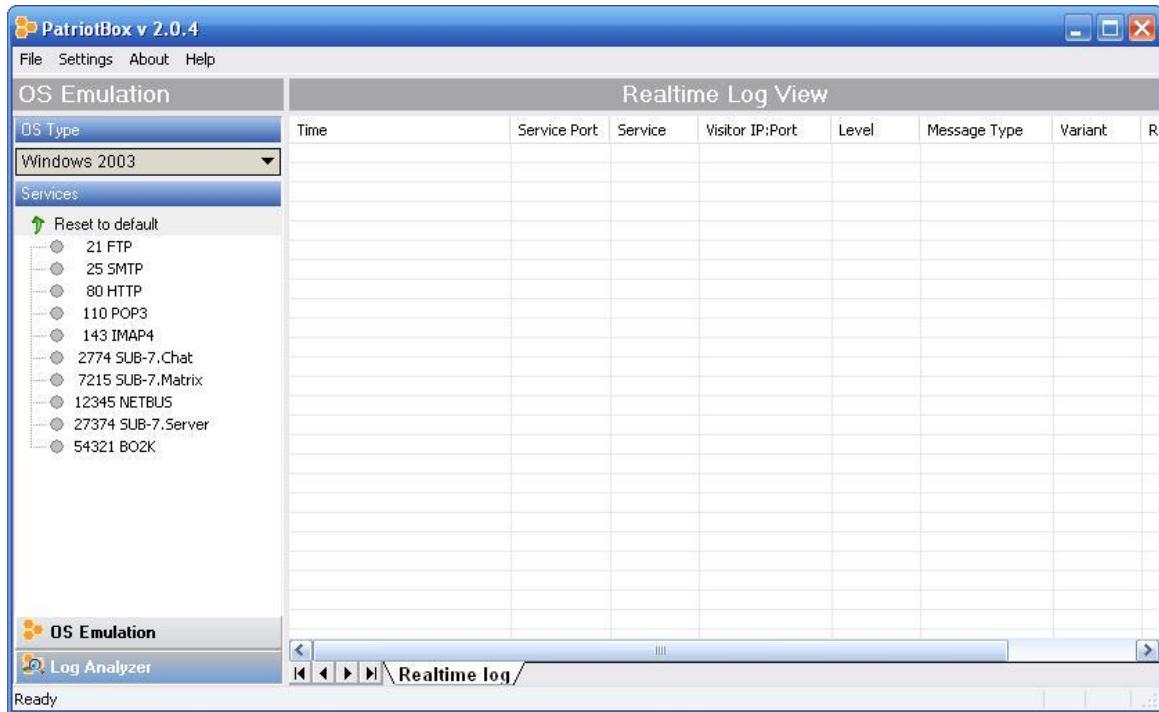
Vamos aos principais campos:

OS Type : Permite selecionar um sistema operacional para “simular”. Possui suporte a diversas versões do Windows, entre outros.

Services: Após selecionar um sistema operacional, os serviços disponíveis irão aparecer para serem configurados. É possível realizar uma configuração individual (com a possível inclusão de scripts) para cada um deles.

Realtime Log view: É onde os logs de acesso serão capturados. Todas as tentativas de acesso serão listadas aqui.

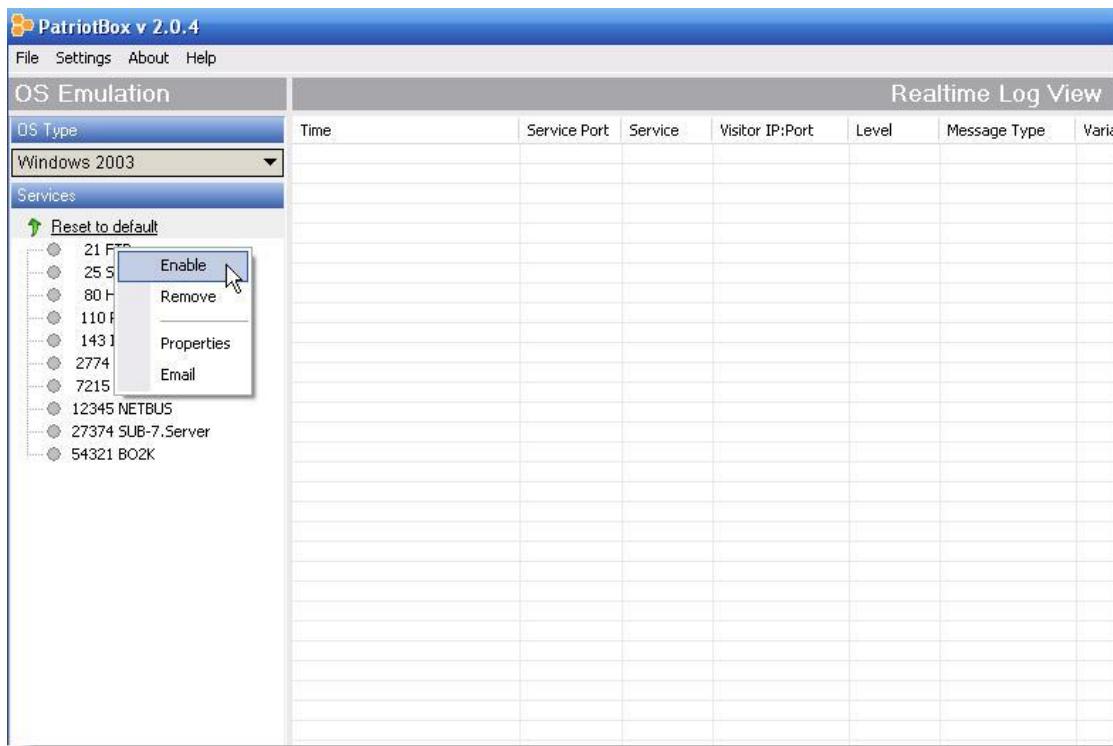
Em OS TYPE, iremos alterar o sistema operacional para Windows 2003 e ver o que acontece:



Perceba que apareceram vários serviços que podem ser utilizados, desde FTP a cavalos de tróia como Netbus e Back Orifice.

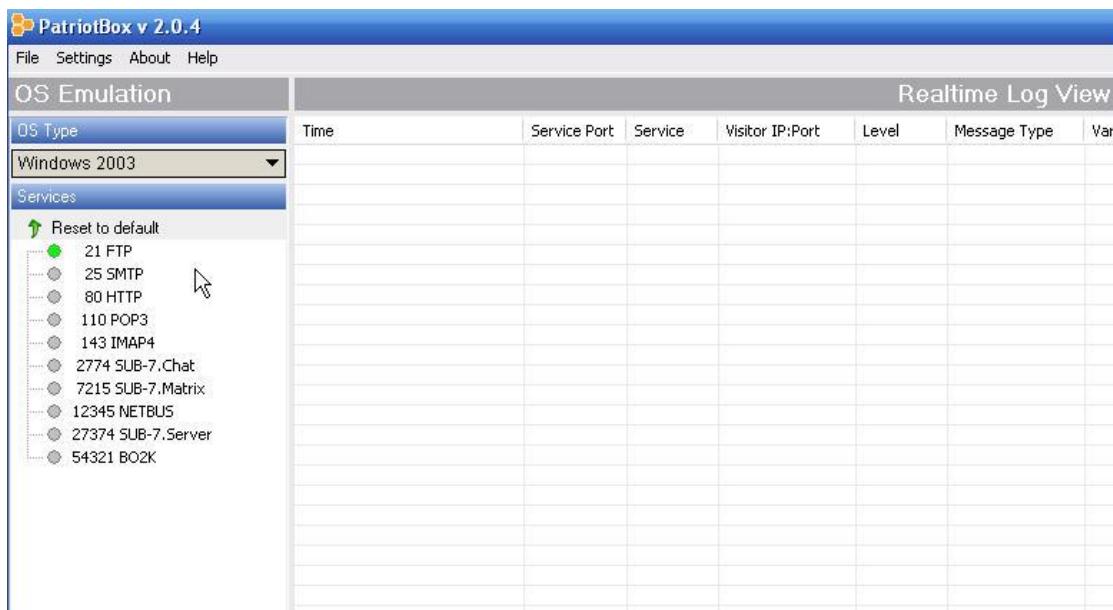
Ativando os serviços

Ativar um serviço é relativamente bem simples, basta clicar nele e selecionar a opção “Enable”. Isso pode ser feito individualmente com cada um dos serviços disponíveis., como visto a seguir:

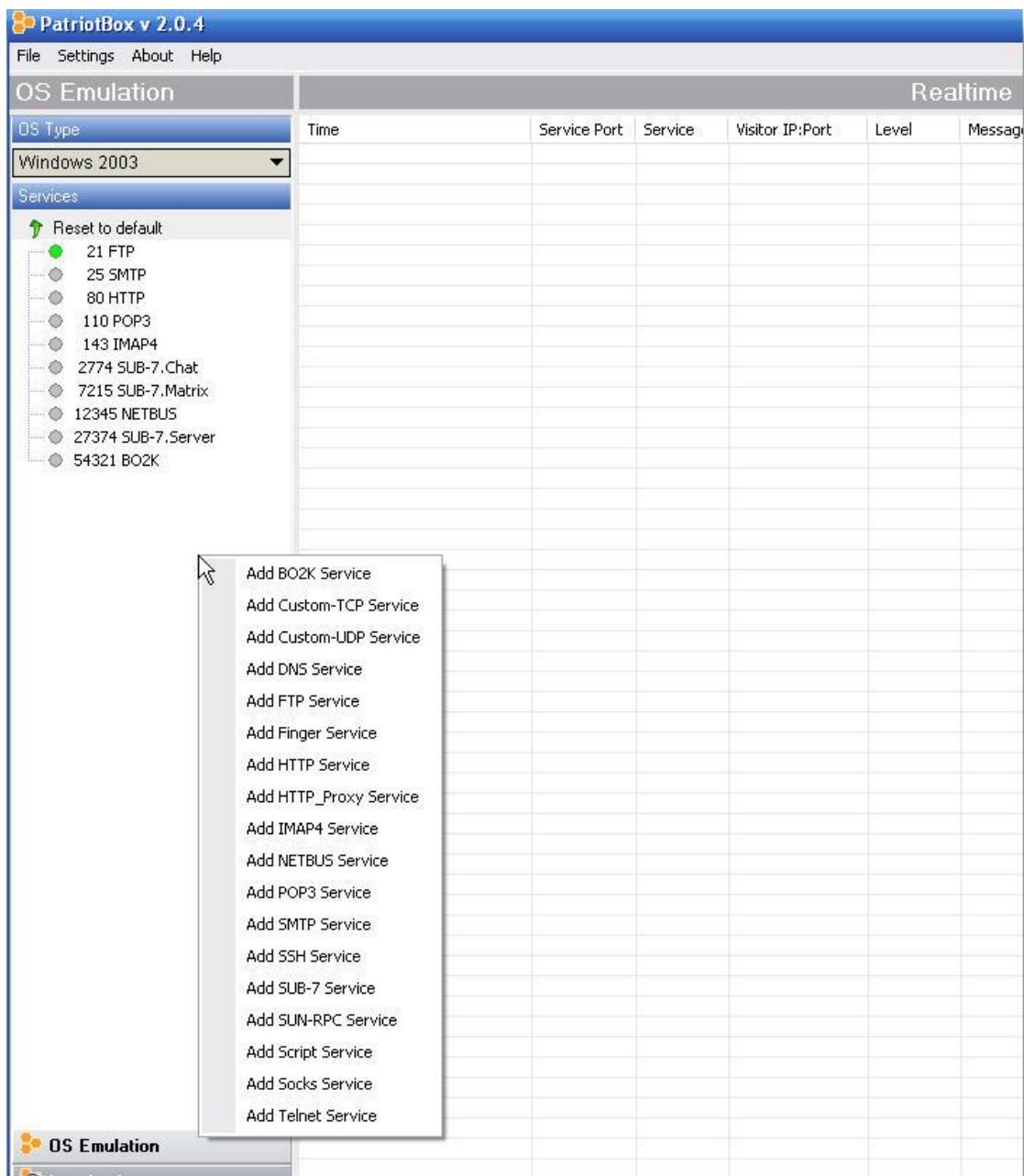


A opção “Remove” apaga o serviço selecionado da lista. A opção Properties abre um menu de opções do serviço. E por último, a opção e-mail permite enviar tentativas de ataque àquele serviço para um endereço de e-mail específico.

Veja como a cor do serviço fica verde, após ser habilitado:



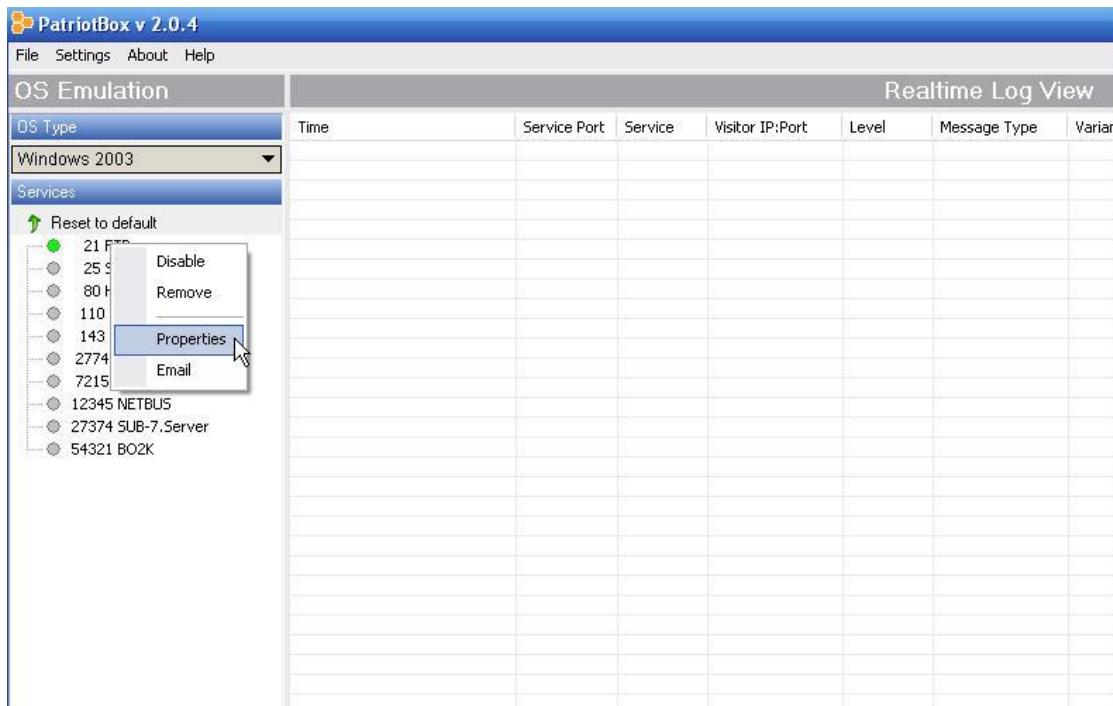
Outra coisa que vale a pena ser comentada, é o fato de que os serviços listados à esquerda não são definitivos. Você pode facilmente adicionar novos serviços se clicar com o botão direito do mouse na parte vazia da aba “Services”. Veja na imagem a seguir:



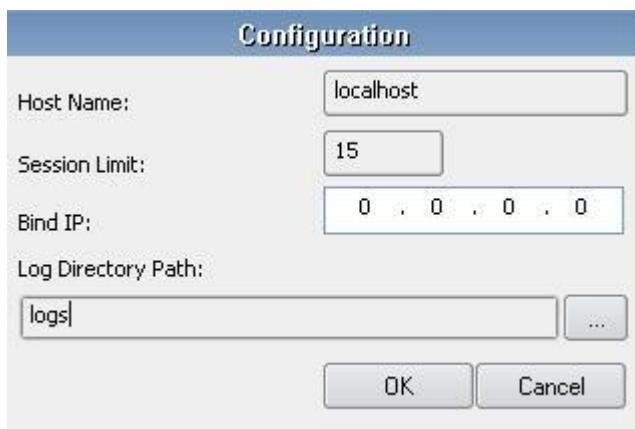
Seja telnet, SSH, DNS, Finger, vários tipos de serviços serão adicionados à lista. Basta apenas você clicar neles para selecioná-los.

Configuração dos serviços

Para configurar um serviço, clique no serviço desejado e seleciona a opção “Properties” (propriedades), como demonstrado a seguir:



A seguinte tela de configuração irá aparecer:



As opções são bem simples, vamos à elas:

Host Name: É o aparente nome da máquina que está rodando o serviço.

Session Limit: É o limite de sessão, ou em outras palavras, quantos usuários podem se conectar simultaneamente.

Bind IP: É qual o endereço IP que será utilizado no serviço. Como assim, você pergunta. Bem, em um computador com múltiplas interfaces de rede você pode querer utilizar o Honeypot em apenas um endereço específico. Basta colocá-lo aqui.

Log Directory Path: É o diretório onde serão armazenados os logs (as tentativas de ataque) deste servidor. O legal do PatriotBox é que você pode ter um diretório de log para cada um dos serviços

Também há as configurações relacionadas ao envio de e-mail, quando você seleciona essa opção no menu principal (ao clicar em um serviço e selecionar “Email”). Vejamos:



As opções são:

Message Title: O título da mensagem que aparecerá no e-mail.

Mail message level: O nível do alerta. Você pode configurar para o PatriotBox dar apenas um alerta ou um alerta e um alarme, por exemplo.

Address: O endereço do servidor SMTP para o envio do e-mail

Auth type: O tipo de autenticação que o servidor de envio de e-mails irá utilizar. Se não utilizar nenhuma, deixe em NONE (que é o padrão).

User login: O nome do usuário do servidor SMTP, caso seja necessário.

User password: A senha do usuário configurado anteriormente

Send to: E-mail de destino que receberá os logs de acesso

Send from: E-mail de origem que enviará os logs de acesso

Após configurar essa opção, você irá configurar o horário de acesso diário para envio dos logs. Isso é feito através dessa caixa de diálogo:

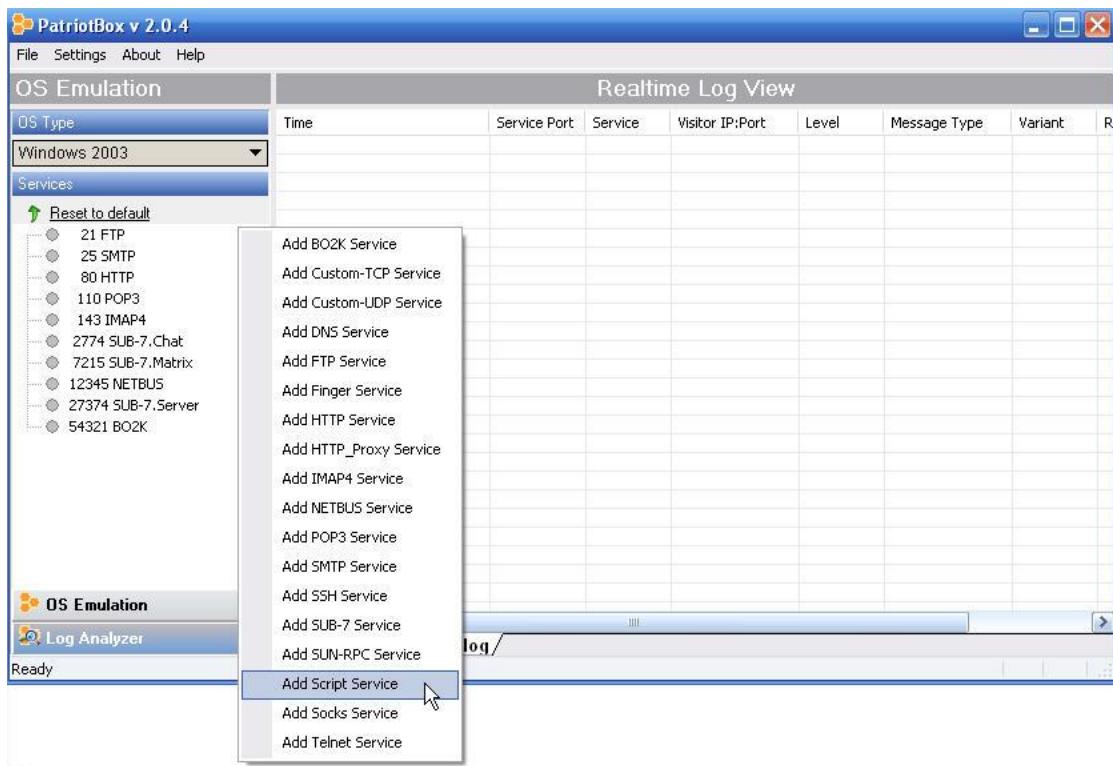


Assim diariamente no horário configurado, os logs serão enviados para o e-mail. Isso evita que você receba excessivamente mensagens na sua caixa postal, o que poderia ser considerado como spam pelo seu servidor.

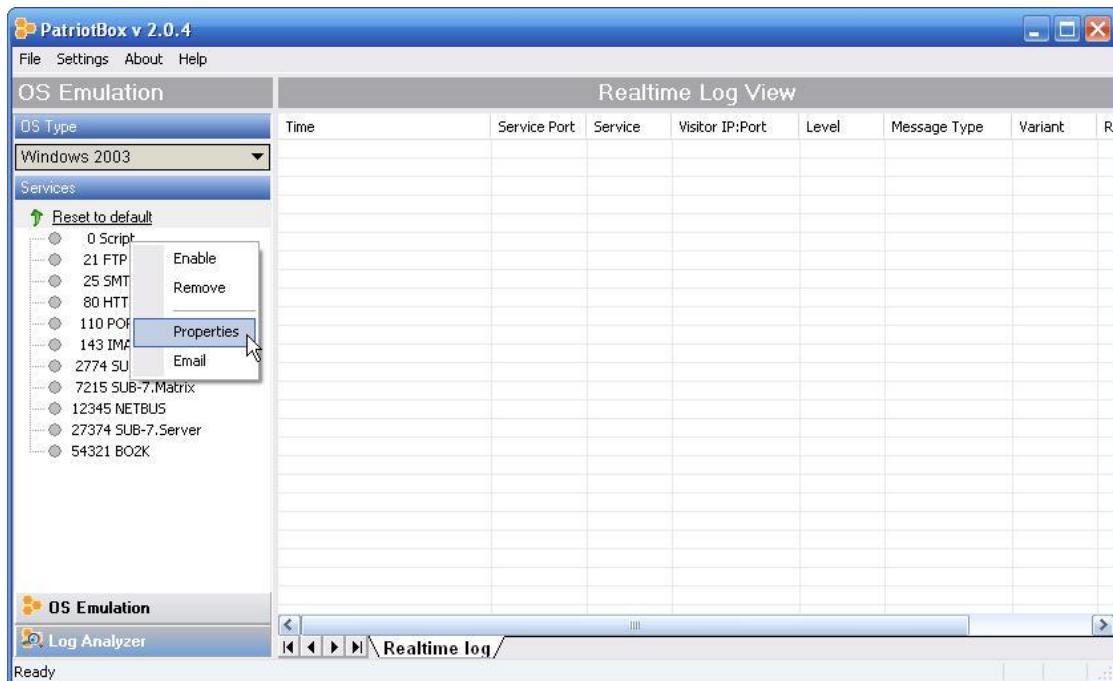
Utilizando scripts personalizados

Como citei anteriormente, o PatriotBox permite que você utilize scripts personalizados, importando por exemplo os scripts do Honeyd. Realizar isso é muito simples. O primeiro passo seria instalar o **ActivePerl** (que pode ser baixado em qualquer site de download, como www.baixaki.com.br, por exemplo).

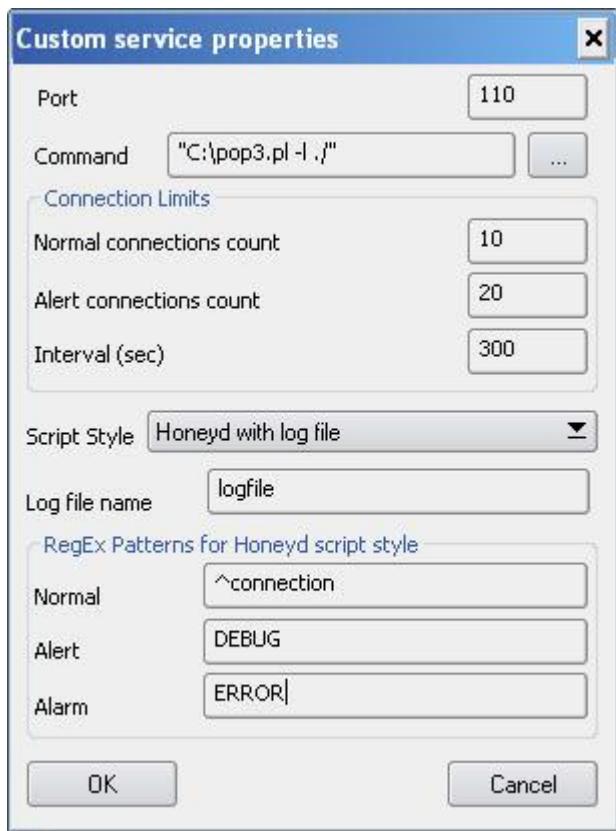
O processo de utilização de tais scripts é bem simples. Primeiramente, você deve mandar adicionar um novo serviço, como mostrado anteriormente. Mas dessa vez deve selecionar a opção “Script”. Vejamos a seguir:



Perceba que a opção agora está disponível na aba scripts. Você deve agora clicar nessa nova opção e selecionar “Properties” (propriedades). Veja:



Teremos então a tela de propriedades para o script, como demonstrada:



As opções são as seguintes:

Porta : A porta na qual o script irá atuar. Não há um padrão, mas depende muito do script. Se o script for de POP3 como neste exemplo, a porta deveria ser a 110 que é a padrão, mas não é obrigatório.

Command : O script em si. No caso do Honeyd, deve ser selecionado arquivo em Perl e passado com a linha de comando correta

As próximas três opções são relacionadas ao limite máximo de conexões.

Normal connections count : O número máximo de conexões “normais” (que não são consideradas como ataques)

Alert connections count: O número máximo de conexões com atividades detectadas como maliciosas

Interval: intervalo em segundos que uma conexão pode ficar ativa

Todas as outras opções são relacionadas aos scripts

Script style: O tipo de script que você irá utilizar. O mais comum é a utilização dos scripts do Honeyd. Você pode fazê-lo com ou sem ter que gravar em log.

Log file name: O arquivo de log que será salvo caso haja conexões ao script

Todas as opções restantes são relacionadas a qual evento externo do Honeyd deve ser disparado em caso de conexão normal, alerta e alarme.