Procedimentos e Ferramentas para Manutenção de Honeynets

Lucio Henrique Franco e Antonio Montes
Laboratório Associado de Computacão e Matemática Aplicada
Instituto Nacional de Pesquisas Espaciais
{lucio,montes}@lac.inpe.br

Resumo

Como parte do processo de manutenção de honeynets, para a monitoração de atividades hostis na Internet, vários procedimentos e ferramentas que automatizam as tarefas de gerenciamento de honeypots de alta interatividade vem sendo desenvolvidos. Este artigo descreve suscintamente estes mecanismos e seu uso nas honeynets.

Introdução

Honeynets são ferramentas de pesquisa que consistem de uma rede projetada especificamente para ser comprometida, elas contêm mecanismos de captura, análise e contenção de tráfego e partem do princípio que todo este tráfego é considerado malicioso [1, 2]. Essas redes são compostas de vários *hosts* chamados de *honeypots* [3], que são um recurso de segurança preparado com a finalidade de ser sondado, atacado ou comprometido e registrar essas atividades [3].

Como parte da operação de um projeto de pesquisa na área de *honeynets* é necessário projetar e implementar diversos procedimentos para ativar ou desativar um *honeypot* em uma *honeynet*. Estes procedimentos variam conforme a taxonomia dos *honeypots* e vão desde zerar os discos da máquina, à escolha do sistema operacional que será instalado e os serviços que serão disponibilizados, até a configuração de mecanismos para coleta e preservação dos artefatos[1] deixados nos *honeypots* pelos invasores.

A padronização destes procedimentos é muito importante, principalmente quando se trabalha com grande número de *honeypots*. Os procedimentos ajudam a evitar erros ou esquecimentos ao longo do processo de manutenção dos *honeypots*. Essas falhas poderiam, por exemplo, contaminar as informações capturadas pelos *honeypots* ou levar o invasor a perceber que está em uma *honeynet*. Estes procedimentos vêm sendo desenvolvidos e aplicados ao longo do tempo desde a implementação das *honeynets*, passando por constantes revisões de aperfeiçoamento.

Tipos de Honeypots

Honeypots podem ser classificados como sendo de baixa ou alta interatividade [4]. Sistemas de baixa interatividade limitam as ações dos atacantes e coletam poucas informações sobre um ataque, porém, são mais simples de se gerenciar e introduzem pequeno risco ao ambiente de rede visto que o atacante não tem acesso total ao sistema. *Honeypots* de baixa interatividade podem emular serviços de rede como, por exemplo, `ftp`, `http`, entre outros.

Honeypots de alta interatividade são capazes de executar as versões reais dos serviços de rede e permitem que o atacante tenha acesso total à máquina comprometida [4]. Esta categoria de *honeypots*, geralmente empregada em *honeynets*, é usada como um recurso para auxiliar no aperfeiçoamento das formas de proteção do ambiente de rede. Estes *honeypots* coletam mais informações que os *honeypots* de baixa interatividade, permitem o acompanhamento dos passos dos invasores e a coleta de artefatos.

Sistemas de baixa interatividade, devido à própria funcionalidade, não devem possibilitar que o atacante tenha controle total do sistema alvo dos ataques e, geralmente, esses sistemas não foram preparados para permitir que o invasor tenha esse controle. Entretanto, falhas nestas aplicações podem levar o invasor a ter acesso ao *honeypot* e comprometer todo o funcionamento da *honeynet* ou da rede de produção em que ele se encontra [5]. Nos *honeypots* de alta interatividade a finalidade é que o invasor possa ter acesso total ao sistema invadido, possibilitando que ele baixe ferramentas e execute comandos. Deste modo, um *honeypot* pode ser utilizado para investigações, para monitoração de acessos não autorizados ou atividades ilícitas.

Como *honeypots* podem ser utilizados pelos atacantes para desfechar ataques a outras redes, faz-se necessário que as *honeynets* tenham mecanismos de contenção de tráfego de alta granularidade para deter estas atividades.

O Projeto Honeynet.BR [6] é baseado na Arquitetura GenII descrita por Lance Spitzner [1, 2] e utiliza *honeypots* de alta interatividade. A gerência desses é complexa, pois não se deve deixar nenhum vestígio da preparação dos sistemas e eles devem conter mecanismos para a captura e preservação dos passos dos atacantes, e para a coleta dos artefatos deixados no sistema. Considerando todos estes aspectos foram desenvolvidos alguns procedimentos, apresentados nas próximas seções, para auxiliar na manutenção desses *honeypots*, envolvendo o processo de sua configuração inicial, do seu acompanhamento e da sua restauração.

[1] Artefatos podem ser definidos como todo o material deixado pelo invasor após o comprometimento de um sistema

Procedimentos Desenvolvidos

Honeynets contêm vários *honeypots* de arquiteturas diferentes, rodando diversos sistemas operacionais cada qual provendo vários serviços locais e de rede. Para a manutenção desses sistemas é necessário desenvolver metodologias e elaborar procedimentos. Estes procedimentos podem ser automatizados por meio de *scripts* que reduzem o tempo entre a desativação e ativação do *honeypot*. Foi adotado um procedimento padrão em todo o processo de preparação dos *honeypots* para se tornarem operacionais em uma *honeynet* e para sua desativação visando a preservação dos artefatos deixados pelo invasor.

Os procedimentos e *scripts* auxiliam, quando se trabalha com grande número de *honeypots*, na execução de passos vitais na configuração de um *honeypot*, como zerar um disco, eliminando qualquer dado de instalações anteriores ainda contido nele [7]. Permite também a remoção de vestígios da configuração evitando desconfianças do invasor com relação à máquina invadida, o que poderia fazer com que ele não mais retornasse ao sistema, com a possibilidade de anunciar na Internet a localização de tal rede, afastando os invasores e impedindo a realização de mais pesquisas nesta rede. Outro aspecto importante destes procedimentos é a documentação da configuração do sistema e dos aplicativos de cada *host* que fará parte da *honeynet*.

O Projeto Honeynet.BR [6] tem como um dos seus objetivos, a pesquisa e o desenvolvimento de procedimentos e ferramentas na área de *honeynets*. Neste projeto foram criados vários procedimentos para captura, análise de tráfego e configuração de *honeypots*, alguns destes procedimentos foram automatizados através de *scripts*. Na próxima seção são apresentados alguns procedimentos para a preparação de um *honeypot* de alta interatividade.

Procedimentos de Instalação dos Honeypots

Inicialmente, define-se o sistema operacional, os serviços e suas versões que serão instalados no *honeypot*; em seguida é necessária a execução de diversos passos para a configuração do *honeypot*, conforme a metodologia a ser empregada e que deve ser definida previamente. Abaixo é apresentada uma visão geral do conjunto de procedimentos básicos que são seguidos durante o processo de instalação de cada *honeypot*. São eles:

1. O disco tem seu conteúdo sobrescrito com um padrão constante de dados (usualmente zeros). Este procedimento permite a geração de sua imagem com uma melhor taxa de compressão e facilita a análise pós-invasão, por constarem no disco somente dados referentes à última instalação [7];

2. O sistema operacional escolhido para o *honeypot* é instalado, geralmente, utilizando as opções padrões de instalação de cada sistema;

3. Depois de instalado o sistema operacional no *honeypot*, ele tem seus serviços configurados e inicializados;

4. É executada uma seqüência de *scripts* responsáveis pela configuração final do *honeypot*, eles criam usuários e senhas já predefinidos pelo grupo, configuram serviço de sincronização de tempo, serviço de gerenciamento de *logs* para exportar as informações geradas pelas aplicações para um *loghost*, entre outros;

5. Visando o monitoramento do *honeypot* são configurados sistemas de captura de teclas que enviam os comandos digitados pelo invasor para o *loghost*, através do serviço de `syslog` [1] ou inserindo essas informações diretamente na rede;

6. São gerados os *hashes* `MD5` dos arquivos dos *honeypots* visando armazenar a priori informações de integridade do sistema. Neste passo, são coletadas também informações de status do sistema com a execução de alguns comando, como: `ps`, `netstat`, `lsof`, `socklist`, `df`, entre outros;

7. É feita uma imagem do disco, que é comprimida e armazenada em outra máquina juntamente com o status do *honeypot* e, posteriormente, esses dados são gravados em uma fita magnética;

8. Todo o processo é registrado em um livro de registro seguindo o Apêndice A descrito no *Honeynet Project* [8];

9. O *honeypot* é conectado na *honeynet* e monitorado até o momento da sua retirada da rede.

Outra padronização adotada é quanto à forma de particionamento dos discos dos *honeypots* e o tamanho dessas partições, pois, no momento da análise forense destes *honeypots*, há sistemas de arquivos que não suportam montar imagens de partições que ultrapassam 2 *Gigabytes*. Um exemplo seria o sistema de arquivo *ext2* do Linux, fazendo necessária a utilização de partições inferiores a 2 *Gigabytes*. *Honeypots* que têm diversas partições como, por exemplo, `/`, `/home/`, `/usr/`, `/var/`, `swap`, tornam trabalhoso e demorado o processo de geração da imagem dessas. No entanto, optou-se por trabalhar nesses sistemas, quando possível, somente com duas partições: `/` e `swap`.

Procedimentos e padronizações também foram desenvolvidos para o armazenamento das imagens após uma instalação e das imagens comprometidas dos discos dos *honeypots* juntamente com os status iniciais e finais de cada *honeypot*. O nome do diretório que contém as imagens e o status de cada sistema é composto pelo nome dado ao *honeypot*, por exemplo, `foobar` acrescido da data atual, mais o nome

do sistema operacional do *honeypot*. Dentro desse diretório raiz são criados mais quatro subdiretórios compostos do prefixo `image` (para as imagens), mais a data atual, mais o nome do *honeypot*, mais o sufixo `orig` para as imagens inicias ou o sufixo `hack` para as imagens finais deste *honeypot*. O status inicial e o final são armazenados em subdiretórios do diretório raiz, `status-hack` e `status-orig`, para as informações extraídas de cada *honeypot* na fase inicial e na sua desativação, respectivamente. Abaixo um exemplo da estrutura de diretórios criada:

```
/foobar_15_03_2003_windows2000server/
        image_15_03_2003_foobar_orig/
        image_15_04_2003_foobar_hack/
        status-orig/
        status-hack/
```

Estrutura de diretórios criada para armazenamento das imagens e do status dos *honeypots*

Procedimentos de Acompanhamento dos Honeypots

Depois que um *honeypot* é disponibilizado na *honeynet* ele passa a ser monitorado e quando invadido os comandos digitados pelos atacantes, suas ações no sistema e outras informações geradas pelas aplicações são enviadas para um *loghost* centralizado e encaminhadas na forma de alertas para os membros do Projeto via e-mail, celular ou qualquer meio que utilize à Internet.

Estes sistemas dos *honeypots* podem vir a sofrer falhas, sejam elas, por problemas de *hardware*, problemas no sistema de arquivos ou alguma interrupção de serviço ou do sistema todo devido a um ataque sofrido. Assim, tornou-se necessário implementar um sistema de acompanhamento destes *hosts*. Semanalmente[2], cada *honeypot* passa pela seguinte série de testes locais via *console*, como mostra a Figura 1.

Ao executar os procedimentos para aferir se um *honeypot* está em funcionamento conforme a configuração inicial é necessário ter alguns cuidados para não deixar vestígios no sistema. Preferencialmente são utilizados binários compilados de forma estática dos comandos executados para a obtenção do status do sistema. Esses binários são executados a partir de um CDROM e têm as saídas de seus comandos direcionados para um disquete. Essas informações são armazenadas posteriormente em um documento que é enviado por e-mail para todos do grupo. São empregados binários estáticos, pois, esses não fazem uso de bibliotecas e/ou arquivos do sistema, que podem ter sido alterados por algum *rootkit* [9]. Entretanto, a utilização desses binários estáticos não dará os resultados esperados se o *rootkit*

[2]Este período é definido pelo grupo

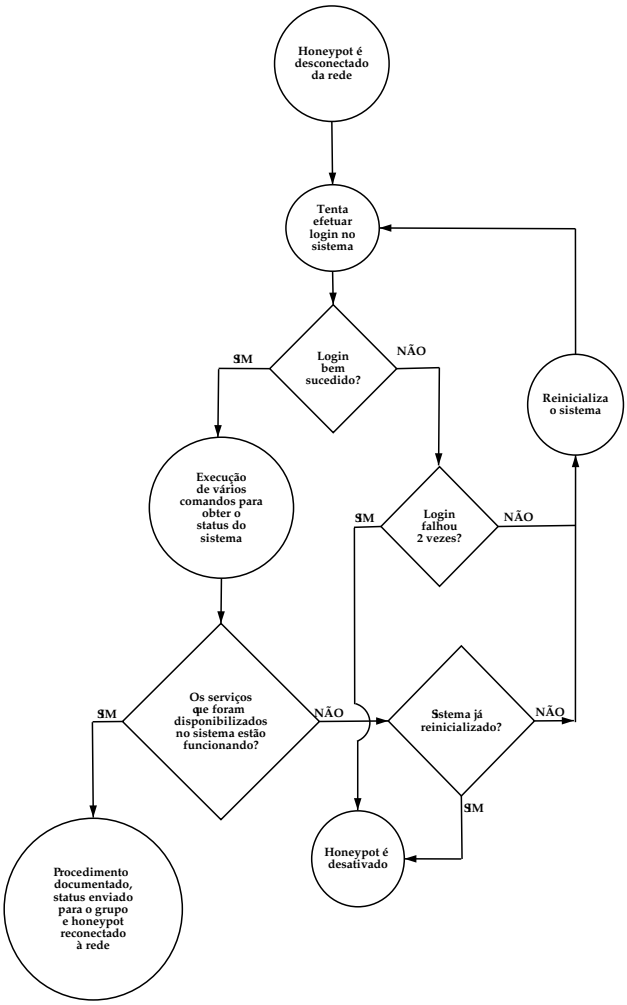

Figura 1: Procedimentos utilizados para o acompanhamento semanal dos *honeypots*.

ou outra aplicação instalada no *honeypot*, pelo invasor, fizer uso de módulos de *kernel* para tentar ocultar a invasão. Por outro lado, isso já é conhecido de antemão, uma vez que as atividades do atacante vem sendo monitoradas e a instalação do *rootkit* e seu resultado já foi registrado. Neste caso, o status é extraído do sistema e comparado com anteriores, dependendo da diferença entre eles, devida a influência da instalação de ferramentas no sistema pelo invasor, o grupo pode definir pela retirada do *honeypot* da *honeynet*.

Na próxima seção são apresentados os procedimentos desenvolvidos e empregados na desativação dos *honeypots*.

Procedimentos de Desativação dos Honeypots

Depois do *honeypot* ser invadido é tomada a decisão pelo grupo, no momento mais oportuno, de sua retirada da rede. Isso normalmente acontece quando o invasor danifica com-

ponentes críticos do sistema ou deixa o *honeypot* num estado inutilizável em decorrência das alterações realizadas no sistema após uma invasão. Para a desativação de cada *honeypot* os procedimentos adotados são semelhantes aos procedimentos de acompanhamento descrito anteriormente. Porém, após a coleta do status do sistema, os seguintes passos são executados:

1. É feita a imagem da máquina comprometida em outra máquina, armazenando este arquivo junto com a imagem inicial e os status iniciais e finais extraídos do *honeypot*, seguindo a estrutura de diretórios apresentados na seção ;

2. O *host* neste momento é liberado para ser preparado e configurado novamente como um *honeypot*, passando novamente por todos os procedimentos descritos na seção .

Nos *honeypots* em que não é possível realizar *login* no sistema através do *console* para a geração da imagem de suas partições, é realizado o *boot* no *host* através do CDROM ou disquete, utilizando uma distribuição Linux como a `tomsrtbt`[3] ou pelo sistema operacional OpenBSD[4], a partir deste ponto, com o auxílio desse outro sistema, as imagens dos discos do *honeypot* são geradas e exportadas para outra máquina.

Estas imagens juntamente com todas as outras informações de status e acompanhamento do sistema são comprimidas, armazenadas em outra máquina e posteriormente transferidas para uma fita magnética.

Na próxima seção são descritas algumas ferramentas desenvolvidas com a finalidade de automatizar os procedimentos e metodologias utilizadas no gerenciamento de *honeypots*.

Ferramentas Desenvolvidas

Ferramentas empregadas na automatização do processo de configuração de honeypots vem sendo desenvolvidas para sistemas Unix, Windows e, principalmente, Linux; elas são desenvolvidas em sua maioria em shell script devido a portabilidade e de não necessitar de instalação adicional de outras aplicações no honeypot. Alguns desses scripts utilizados na fase de instalação dos honeypots são descritos abaixo:

Gerador de Hash MD5 de Arquivos e Compactação dos Arquivos Gerados

A função do *script* apresentado nesta seção é gerar o *hash* `MD5`[5] de todos os arquivos num dado diretório; incluindo arquivos de subdiretórios. As informações resultantes são armazenadas em um arquivo de saída. No diretório `/dev/` é realizado o hash do arquivo que contém a saída do comando `/bin/ls -lac` que apresenta a data de alteração do diretório, já que é comum a utilização deste diretório para armazenamento de artefatos.

Este *script* foi desenvolvido para os sistemas Linux, FreeBSD, OpenBSD, Solaris e Windows, esse último utilizando o software `cygwin`[6].

O *script* também suporta vários diretórios como entrada do sistema, neste caso, ao término de sua execução, é criado um arquivo, para cada diretório, contendo as informações coletadas nesse diretório, e por fim, é realizada uma compressão do arquivo gerado. A formação do nome dos arquivos já comprimidos é composta do nome do *honeypot*, acrescentado do `path` do diretório e do sufixo `md5sum`. Dentre os diretórios dos sistemas Unix em que esse *script* é executado, pode-se citar: `/dev/`, `/usr/local/etc/`, `/usr/sbin/`, `/usr/local/bin/`, `/usr/bin/`, `/usr/local/sbin/`, `/etc/`, entre outros.

Ferramenta para Remoção Segura dos Arquivos Copiados

Uma ferramenta desenvolvida utilizando os comandos `awk`, `dd` e `rm`, é responsável por sobrescrever o conteúdo de todos os arquivos de um diretório, dado como entrada, com um padrão de zeros e posteriormente remover esses arquivos [7]. Isto permite apagar todos os arquivos tranferidos para o *honeypot* após sua instalação e que foram necessários para sua configuração, isso facilita a análise forense desses sistema.

Configurações Gerais dos Honeypots

Os *scripts* descritos até o momento são integrantes de um sistema geral (*framework*) que segue todos os procedimentos criados para o processo de instalação de um *honeypot* e descrito na seção . Abaixo é apresentada a ordem de execução dos diversos *scripts*:

1. **Criação das contas dos usuários:** Todos os *honeypots* têm usuários legítimos que são criados logo após a instalação do sistema. Os usuários têm nomes fictícios e são adicionados através do comando `adduser` com os parâmetros necessários para cada sistema operacional;

2. **Criação das senhas dos usuários:** Nesta função são criadas as senhas dos usuários adicionados no passo anterior. As senhas seguem um padrão forte especificado pelo grupo, o comando utilizado é o `passwd` com seus parâmetros;

[3] `http://www.toms.net/rb/`
[4] `http://www.openbsd.org/`
[5] `http://www.ietf.org/rfc/rfc1321.txt`
[6] `http://www.cygwin.com/`

3. **Instalação dos mecanismos de coleta de *logs*:** Como o principal objetivo de *honeynets* é observar as ações dos atacantes, faz-se necessária a utilização nos *honeypots*, de diversos mecanismos para coleta de *logs*. Esse é responsável por exportar para outra máquina todas as informações sobre o comportamento do sistema. Na arquitetura utilizada, as informações geradas em cada *host* são exportadas para um *loghost*. Este passo estabelece também qual o endereço *IP* do *loghost* e qual a severidade das informações que serão exportadas para ele;

4. **Instalação do mecanismo de sincronização de tempo:** É essencial um sistema para sincronismo de tempo entre os *hosts* para que se possa armazenar os *logs* de forma correta, permitindo a correlação de eventos. Esta função é responsável por instalar no sistema uma aplicação para ajuste de tempo e suas dependências. Outra configuração realizada neste passo visa especificar o endereço *IP* do servidor com o qual o *honeypot* será sincronizado, este endereço pode ou não estar contido dentro da *honeynet*;

5. **Compilação e instalação do sistema de captura de teclas:** As máquinas Unix da *honeynet* possuem um sistema de captura de teclas que envia o histórico dos comandos digitados via o serviço de `syslog` [1] ou módulo de *kernel*. Esta função compila e instala este mecanismo;

6. **Remoção dos arquivos fontes instalados:** Todas as aplicações a serem instaladas são baixadas para o *honeypot*. Neste procedimento são removidos todos os arquivos fontes baixados e instalados no *honeypot*, além dos traços deixados na instalação do sistema em arquivos como `$HOME/.bash_history`. O *script* utilizado neste passo é descrito na seção ;

7. **Geração do *hash* `MD5` do sistema:** São gerados os *hashes* `MD5` dos arquivos do sistema utilizando o *script* descrito na seção ;

8. **Geração do status do sistema:** Neste momento, após o *honeypot* ser configurado, é gerado o status do sistema com informações de saídas dos comandos `ps`, `df`, `rpcinfo`, `netstat`, `lsof`, entre outros. Essas são utilizadas na análise forense e em comparações com o status gerado no momento de retirada do *honeypot* da rede;

9. **Geração da imagem do disco:** Por fim, uma imagem das partições do disco do *honeypot* é transferida para a máquina responsável por armazenar as imagens e status dos *honeypots*. O *script* utiliza os comandos `dd` e `netcat` em conjunto. Inicialmente é executado o comando `nc -l -p 10000` na máquina destino para abrir um `server socket` *TCP* na port 10000 e aguardar conexões do *honeypot*. A saída padrão desse comando é direcionado para um arquivo que será a própria imagem da partição do disco do *honeypot*. No *honeypot* é executado o comando `dd` com os parâmetros para ler blocos de uma dada partição do sistema, em seguida esses são enviados via `pipe` para o comando `netcat`, que atua no *honeypot* como aplicação cliente conectando-se na aplicação servidora na máquina destino, o qual se encarrega de transferí-los para a máquina repositório de imagens;

10. **Compressão das partições:** A partir de agora o *honeypot* pode ser colocado em operação e a imagem juntamente com seu status inicial são organizados na máquina que armazena essas informações seguindo a estrutura de diretórios descrita na seção .

Trabalhos Relacionados

Lance Spitzner [4] discutiu a importância de se desenvolver procedimentos para a configuração dos *honeypots* de forma correta e segura conforme cada taxonomia.

Há várias ferramentas que podem auxiliar no processo de ativação e desativação de *honeypots*, entre elas o `md5deep`[7] que é um programa escrito em linguagem C semelhante ao *script* descrito na seção que gera o *hash* `MD5` de um determinado diretório. Porém, o *script* realiza todas as funções desta ferramenta e apresenta como vantagem sobre essa, sua alta portabilidade e integração com outros *scripts* desenvolvidos em conjunto.

Outra ferramenta auxiliar para a remoção segura de arquivos do sistema é o comando `shred` pertencente ao pacote `fileutils`[8] da GNU[9], porém, com a utilização do *script* desenvolvido com a mesma funcionalidade evita-se a instalação de pacotes extras ou a migração dessas aplicações para outras plataformas.

Trabalhos Futuros

Com a necessidade da realização de testes com diversos sistemas operacionais e aplicações em *honeynets*, os procedimentos elaborados e as ferramentas desenvolvidas para automatizar a configuração e o gerenciamento desses sistemas devem ser revistos para acompanhar as mudanças ou suportar novas características do sistema que está sendo implementado.

Um dos estudos realizados sobre as ferramentas desenvolvidas, conclui-se que é recomendável a migração da ge-

[7] `http://md5deep.sourceforge.net/`
[8] `ftp://ftp.gnu.org/gnu/fileutils/`
[9] `http://www.gnu.org/`

ração do *hash* `MD5`, dos arquivos do sistema, para `SHA1`[10] (*Secure Hash Algorithm 1*), devido à saída de 160 bits que este algoritmo utiliza, ao contrário dos 128 bits empregado pelo `MD5`, evitando colisões nos resultados gerados em cada algoritmo. Porém, o algoritmo `SHA1` não está disponível em todas as distribuições Linux em sua instalação padrão, o que obrigaria a instalação adicional desse software. Como medida de simplificação e compatibilidade das ferramentas com todos os sistemas utilizados na *honeynet*, optou-se por utilizar o algoritmo `MD5`, que pode ser migrado facilmente para `SHA1`, quando esse último estiver amplamente disponível.

Dentre outras ferramentas que serão desenvolvidas, uma delas será a elaboracão de um comando `rm` modificado para sistemas Unix. Esse terá a finalidade de retirar uma cópia dos arquivos e/ou diretórios que estão sendo removidos do sistema comprometido. Facilitando, desta maneira, a coleta de artefatos e alterações do sistema deixados pelos atacantes.

Muitas vezes a recuperação desses elementos removidos é feita através da análise do tráfego de rede capturado. Essa decodificação se torna praticamente impossível quando o atacante faz uso de criptografia para a transferência de seus dados. Com o novo comando, os arquivos e/ou diretórios que seriam apagados do sistema, passarão a ser movidos para outro diretório oculto no próprio *host* ou enviado para o *loghost*, de modo transparente para o invasor. Esse estará certo que seus artefatos foram removidos do sistema, quando na verdade, este material estará sendo armazenado para posterior análise forense.

Outras ferramentas para coleta do status de acompanhamento dos *honeypots* de forma automatizada estão em desenvolvimento. As ferramentas utilizadas atualmente também poderão ser migradas para outras linguagens como a linguagem `Perl` ou a linguagem C caso haja necessidade.

Conclusão

A criação de metodologias, elaboração de procedimentos e o desenvolvimento de ferramentas para a configuração de um *honeypot* de alta interatividade são extremamente necessários, porque automatizam processos e auxiliam nas coletas de informações. São criados padrões a serem seguidos por todo o grupo, evitando erros como desligar o *honeypot* sem a geração do seu status final, o que prejudicaria a comparação das características do sistema no momento de sua instalação com o seu status depois de uma invasão. Diminuindo também o tempo que o *honeypot* fica fora da rede até sua próxima configuração.

As ferramentas vem sendo desenvolvidas e empregadas ao longo do tempo desde a implantação do Projeto Honeynet.BR em dezembro de 2001, quando tiveram início as definições dos procedimentos adotados para se configurar todos os *hosts* que fariam parte desta rede. O acompanhamento da Honeynet.BR desde então mostrou a utilidade do desenvolvimento destas ferramentas para automatizar a configuração de um *honeypot*, seus ajustes e refinamentos.

[10] `http://www.ietf.org/rfc/rfc3174.txt`

Referências

[1] The Honeynet Project, *Know Your Enemy: Revealing the Security Tools, Tactics, and Motives of the Black-hat Community*. Addison-Wesley, 1st ed., August 2001. ISBN 0-201-74613-1.

[2] L. Spitzner, "Learning the Tools and the Tactics of the Enemy with Honeynets," in *Proceedings of the 12th Annual Computer Security Incident Handling Conference*, (Chicago, IL, USA), June 2000.

[3] L. Spitzner and M. Ranum, "Honeypots: Tracking Hackers," in *SANS 2002 Annual Conference*, (Orlando, Florida, USA), April 2002.

[4] L. Spitzner, "HOSUS(Honeypot Surveillance System)," in *;login: Magazine of Usenix and Sage*, vol. 27, December 2002.

[5] C. Brenton, "Honeynets," in *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 4232, November 2001. ISSN 0277-786X CODEN: PSISDG.

[6] A. B. Filho, A. S. M. S. Amaral, A. Montes, C. Hoepers, K. Steding-Jessen, L. H. Franco, and M. H. P. C. Chaves, "Honeynet.BR: Desenvolvimento e Implantação de um Sistema para Avaliação de Atividades Hostis na Internet Brasileira," in *Anais do IV Simpósio sobre Segurança em Informática (SSI'2002)*, (São José dos Campos, SP), pp. 19–25, Novembro 2002. `http://www.lac.inpe.br/security/honeynet/papers/hnbr-ssi2002.pdf`.

[7] D. Farmer and W. Venema, "Being Prepared for Intrusion," *Dr. Dobb's Journal*, vol. 26, April 2001.

[8] L. Spitzner, "Honeypot Deployment Log." `http://project.honeynet.org/alliance/AppendixA.txt`, 2001.

[9] N. Murilo and K. Steding-Jessen, "Métodos para Detecção Local de Rootkits e Módulos de Kernel Maliciosos em Sistemas Unix," in *Anais do III Simpósio sobre Segurança em Informática (SSI'2001)*, (São José dos Campos, SP), pp. 133–139, Outubro 2001.