Snap - Um Monitor de Redes com Banco de Dados de Conexões para o Sistema Operacional Linux

Rudinei José Cantarelli
cantarel@zaz.com.br

Roland Teodorowitsch (orientador)
roland@ulbra.tche.br

Universidade Luterana do Brasil - Campus Gravataí
Centro de Ciências Naturais e Exatas - Departamento de Informática
Estrada Itacolomi, 3.600 - Bairro São Vicente - CEP 94170-240 - Gravataí-RS

Resumo

Este trabalho apresenta um monitor de redes que mantém um banco de dados de conexões feitas através de terminais remotos (usando comandos como TELNET). O sistema implementado é composto por um processo residente e por uma ferramenta de visualização. O processo residente executa sobre uma interface de rede em modo promíscuo capturando os pacotes das conexões de TELNET. Estes pacotes são classificados por conexão, montados e salvos em um banco de dados de conexões. A ferramenta de visualização é utilizada para percorrer o banco de dados, visualizando e eliminando conexões antigas. O objetivo principal deste trabalho é o desenvolvimento de uma ferramenta que possa ser utilizada para identificar como um *hacker* está explorando as vulnerabilidades do sistema. Se uma invasão for detectada, o administrador pode verificar como o *hacker* a executou. O sistema, chamado Snap, foi desenvolvido para o sistema operacional Linux, mas pode ser executado em outros sistemas Unix com poucas modificações.

Abstract

This work presents a network monitor that maintains a database of connections made through remote terminals (using commands like TELNET). The implemented system is composed by a daemon process and a visualization tool. The daemon process runs over a promiscuous network interface capturing packets from TELNET connections. These packets are classified by connection, assembled and saved on a database of connections. The visualization tool is used to browse the database, to visualize and to eliminate old connections. The main objective of this work is the development of a tool that can be used to identify how a hacker is exploiting the vulnerabilities of the system. If an invasion is detected, the administrator can verify how the hacker did it. The system, called Snap, was developed for the Linux operating system, but it can be executed on other Unix systems with a few changes.

1 Introdução

O crescimento acelerado da tecnologia de redes de computadores, impulsionado inicialmente pela necessidade de compartilhar informações e mais recentemente pela possibilidade de exploração do comércio via Internet, tem revelado em muitos casos a fragilidade dos sistemas computacionais. Os sistemas operacionais, na medida em que disponibilizam cada vez mais serviços para os seus usuários, tornam-se também cada vez maiores e mais complexos, e, portanto, mais sujeitos a falhas. São justamente essas falhas que são exploradas por *hackers* na invasão de sistemas.

Hoje em dia é possível encontrar centenas de programas e escriptes com os quais se pode iniciar tanto ataques internos, quanto externos. Enquanto *hackers*, de um lado, tem à sua disposição ferramentas para facilitar a invasão, administradores, do outro lado, devem se aparelhar com todo tipo de ferramenta para evitar invasões, ou no mínimo detectá-las.

O presente trabalho apresenta uma ferramenta cujo principal objetivo é ajudar o administrador de sistemas a identificar como ocorreu a invasão e que ferramentas o invasor está utilizando para executá-la. A ferramenta, chamada Snap, é um monitor de redes que captura todos os pacotes que trafegam na rede local, isolando e classificando os pacotes para o protocolo de aplicação TELNET [5] com o objetivo criar um banco de dados de conexões. Desta forma, no caso de ocorrer alguma invasão, mesmo que o *hacker* tenha removido todas as marcas da sua presença, todo o conteúdo da sua conexão estará armazenado no banco de dados, permitindo que o administrado identifique: através de que conta se processou o ataque, o que o invasor fez durante a sua permanência no sistema, que ferramentas foram utilizadas, etc.

O trabalho está organizado em cinco seções. Na seção 2, é definido o ambiente sobre o qual o Snap deverá ser executado, bem como os requisitos necessários. A seguir, a seção 3 descreve o funcionamento do Snap e a seção 4 descreve os arquivos que compõem o pacote de distribuição do Snap. Por fim, são apresentadas as conclusões do trabalho.

2 O Ambiente do Snap

O Snap é um monitor de redes capaz de capturar pacotes que trafegam em uma rede local, classificando estes pacotes e armazenado-os em arquivos que compõem um banco de dados de conexões. Entre as motivações que levaram ao seu desenvolvimento está a criação de uma ferramenta com a qual se pudesse identificar como determinado usuário (em geral um *hacker*) conseguiu explorar as falhas do sistema. Também são objetivos do Snap: ser facilmente instalável e configurável, dispor de um ambiente que pudesse ser facilmente reconfigurado para necessidades específicas e não implicar em maiores custos, podendo ser implementado com recursos disponíveis localmente.

Desta forma, a ferramenta foi desenvolvida usando o sistema operacional Linux [3]. Este sistema operacional é um clone do Unix licenciável gratuitamente, oferecendo, desta forma, a mesma versatilidade da família de sistemas operacionais Unix a um custo insignificante. O Snap, apesar de ter o Linux como plataforma nativa, pode ser portado para outros sistemas Unix sem grandes alterações.

Os requisitos mínimos necessários para compilação e uso do Snap são os seguintes:

- computador com placa de rede *Ethernet* rodando o sistema operacional Linux;
- biblioteca Curses [4], que é utilizada para controle do terminal.

Para o desenvolvimento do monitor foi utilizada a linguagem de programação C e chamadas da API (*Application Program Interface*) *Sockets* [4,6], que permite a uma aplicação iniciar uma conexão e enviar ou receber dados da rede através desta conexão.

Sugere-se o uso do Snap em uma máquina dedicada, removendo desta máquina todos os serviços possíveis, com o objetivo de evitar que a mesma possa se tornar alvo de ataques. No entanto, caso não seja possível alocar uma máquina exclusiva para isto, pode-se executar o Snap até mesmo em uma das máquinas que se deseja monitorar.

3 O Snap

O ambiente desenvolvido é composto basicamente por dois módulos: um módulo de captura de pacotes e armazenamento de conexões (monitor) e um módulo para consulta de conexões (visualizador).

3.1 O Monitor

O funcionamento básico do monitor, cujo executável se chama **snap**, consiste em estabelecer uma conexão capaz de ler todos os pacotes que trafegam pela rede, independentemente do seu destino. Este modo de leitura, em que uma aplicação é capaz de ler pacotes não necessariamente destinados à estação onde executa o monitor, chama-se modo "promíscuo".

Após a leitura dos pacotes, o monitor extrai destes pacotes informações que serão utilizadas para identificar protocolo, origem, destino, portas, etc. Inicialmente verifica-se se o pacote pertence ao protocolo TELNET e se o pacote está relacionado a alguma das máquinas que se deseja monitorar. A definição das máquinas que se deseja monitorar deve estar especificada no arquivo **`snap.conf`**. Este arquivo contém simplesmente uma relação de endereços IP (Internet Protocol) e números de portas que devem ser monitorados.

Os pacotes são então classificados por conexão, de acordo com o endereço IP e a porta TCP (Transmission Control Protocol) ao qual se destinam [1,2]. Para cada conexão são armazenados: o número do IP de origem do pacote, o número do IP de destino, o número da porta de origem, o número da porta de destino, a hora em que a conexão foi iniciada e a hora em que a conexão foi finalizada, além dos dados da conexão lidos da rede. Cada conexão em andamento tem a ela associada um arquivo temporário, cujo nome segue o padrão **`db/buffer_*.tmp`**. Quando a conexão é encerrada, as informações são definitivamente transferidas para o banco de dados. Para cada conexão é atribuído um número que será usado para compor o nome do arquivo no banco de dados. O número é obtido a partir do arquivo **`db/snap.fid`** e é incrementado a cada nova conexão. O nome do arquivo definitivo para a conexão segue o padrão **`db/*.snap`**, como, por exemplo, **`db/00000343.snap`**. Todas as conexões finalizadas são ainda registradas no arquivo de índices **`db/snap.index`**, que será utilizado pelo visualizador de conexões.

O monitor utiliza também arquivos especiais para registrar suas ações (**`log/snap.log`**) e eventuais erros ocorridos (**`log/error.log`**).

3.2 O Visualizador

O funcionamento básico do visualizador, cujo executável chama-se **`vsnap`** (*view snap*) consiste em ler um arquivo contendo uma lista de conexões realizadas (**`db/snap.index`**) e apresentá-las na tela. A partir daí, o usuário poderá optar por visualizar os dados da conexão ou eliminar a conexão do banco de dados.

Inicialmente o visualizador apresenta uma listagem com todas as conexões capturadas. Para cada conexão são exibidos os seguintes dados: endereço IP da máquina de destino, endereço IP da máquina de origem, número da porta TCP de destino, número da porta TCP de origem, hora em que foi estabelecida a conexão, hora em que foi finalizada a conexão. A figura 1 mostra uma tela do visualizador.

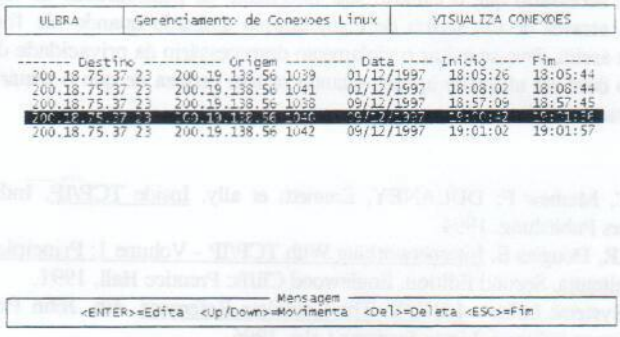

Figura 1 - O Visualizador de Conexões

O visualizador permite que o usuário navegue pelas conexões usando as setas de cursor. Para visualizar o conteúdo de uma conexão, o usuário deverá pressionar a tecla [Enter] sobre a conexão desejada, e para excluir uma conexão, deverá usar a tecla [Delete] - neste caso será solicitada uma confirmação.

4 Os Arquivos da Distribuição

O pacote de distribuição do Snap está disponível no endereço `http://www.ulbra.tche.br/~roland/tcs.html`. Atualmente na versão 1.1, o pacote pode ser obtido gratuitamente nos formatos ".zip" ou ".tar.gz". Os arquivos e diretórios que compõem a distribuição são:

- **`CHANGES`**: contém o histórico de alterações para cada versão;
- **`COPYING`**: contém instruções sobre os direitos autorais da distribuição;
- **`Makefile`**: arquivo usado para compilação dos executáveis;
- **`README`**: contém instruções para instalação e compilação;
- **`VERSION`**: número da versão da distribuição;
- **`bin/`**: neste diretório são colocados os arquivos executáveis do monitor (**`snap`**) e do visualizador (**`vsnap`**);
- **`conf/`**: diretório que contém o arquivo de configuração do Snap (**`snap.conf`**);
- **`db/`**: diretório onde o monitor armazena os dados das conexões capturadas;
- **`log/`**: neste diretório o monitor gera arquivos com informações sobre o funcionamento do monitor (**`snap.log`**) e erros ocorridos (**`error.log`**);
- **`src/`**: diretório com o código-fonte do monitor e do visualizador de conexões;
- **`tmp/`**: diretório para criação de arquivos temporários.

5 Conclusão

Este trabalho, que foi inicialmente desenvolvido como Trabalho de Conclusão do Curso de Bacharelado em Informática na Universidade Luterana do Brasil, apresentou o desenvolvido de um monitor para o sistema operacional Linux. Seu principal objetivo é fornecer uma ferramenta de baixo custo para que o administrador de sistemas possa identificar como um determinado usuário conseguiu burlar os mecanismos de segurança do sistema. O monitor cria um banco de dados de conexões realizadas através do protocolo TELNET. Este banco de dados, por sua vez, pode ser consultado através de um visualizador. O ambiente desenvolvido, distribuído gratuitamente, pode ser facilmente instalado e configurado sem implicar em maiores custos.

Considerando o fato de que todas as informações que o usuário digita ou recebe em uma conexão são armazenadas no banco de dados, esta ferramenta deve ser utilizada com cuidado. Inicialmente, é necessário que o usuário seja informado da possibilidade de monitoramento de suas conexões através de um termo de compromisso assinado quando ele for cadastrado no sistema. Ainda assim, deve-se evitar o violamento desnecessário da privacidade dos usuários. Ou seja, o recurso deve ser utilizado apenas quando se tem certeza de que o usuário está tentando executar um ataque ao sistema.

Referências

[1] ARNETT, Mathew F; DULANEY, Emmett; et ally. Inside TCP/IP. Indianapolis: News Riders Publishing. 1994.

[2] COMMER, Douglas E. Internetworking With TCP/IP - Volume 1: Principles, Protocols and Architetura. Second Edition. Englewood Cliffs: Prentice Hall, 1991.

[3] LINUX Systems Labs. - LINUX: The Complete Reference. 4th. John Purcell e Amanda Robinson (editors). Linux Systems Labs, 1996.

[4] MATTHEW, Neil; STONES, Richard. Beginning Linux Programming. Birmingham: Wrox Press, 1996.

[5] POSTEL, J.B.; REYNOLDS, J.K. RFC854: TELNET Protocol Specification, Network Working Group, Request For Comments: 854, may 1983.

[6] STEVENS, W. Richard. UNIX Network Programming. Englewood Cliffs: Prentice Hall, 1990.