Log Analyzer: Uma Proposta de Software Livre Para o Correlacionamento de Eventos em Arquivos de Log

Weverton Cordeiro[1], Vanner Vasconcellos[2], Antônio Abelém[1]

[1]Departamento de Informática – Universidade Federal do Pará (UFPA)
Belém, Brasil

[2]Ponto de Presença da Rede Nacional de Pesquisa – Pará (PoP-PA)
Belém, Brasil

wevertoncordeiro@acm.org, vanner@pop-pa.rnp.br, abelem@ufpa.br

Abstract. *In this paper a novel software is proposed based on querying system logs and correlating the events registered in those logs. The purpose of this approach is to identify the hosts causing violations to predefined security policies in computer networks.*

Resumo. *Neste artigo é proposto um novo software para a detecção de atividades maliciosas na rede, através de consultas aos arquivos de log do sistema e do correlacionamento de eventos registrados nesses arquivos com o objetivo de identificar os responsáveis por violações a políticas de segurança em redes de computadores.*

1. Introdução

O uso de software livre na gerência e administração de redes de computadores é uma constante no atual contexto da maioria das redes de computadores institucionais e acadêmicas. As empresas investem na contratação e no aperfeiçoamento de recursos humanos especializados nas mais novas tendências em software livre, e na migração de plataformas proprietárias para plataformas livres, com o principal objetivo de redução de custos, visando à obtenção de software que melhor se adapte a requisitos de eficiência e confiabilidade, entre outros.

Este artigo tem como objetivo endereçar um dos problemas enfrentados pelos administradores de redes – o rastreamento de atividades maliciosas que ocorreram internamente a seu domínio administrativo – ao propor uma tecnologia alternativa, baseada em software livre, para auxiliar de forma eficiente o processo de identificação de responsáveis por anomalias de segurança detectadas em redes de computadores, ou mesmo no rastreamento de atividades nas mesmas, através do correlacionamento de eventos registrados nos arquivos de log gerados pelo Netfilter [Netfilter/iptables 2006] e pelo servidor DNS da rede. Para os fins desta proposta, são considerados os arquivos de log gerados pelo servidor *Bind* [Albitz and Liu 2001], principal referência em termos de servidor DNS. O restante do artigo está dividido em 4 seções. A Seção 2 discute alguns dos problemas tipicamente enfrentados por administradores de redes. A Seção 3 apresenta abordagens existentes no sentido de auxiliar os administradores de redes na solução dos problemas de segurança enfrentados. A Seção 4 apresenta o Log Analyzer, um software para correlacionamento de eventos em arquivos de log. A Seção 5 apresenta uma breve discussão sobre o processo de validação do software, enquanto que a Seção 6 conclui o artigo e apresenta possíveis trabalhos futuros.

2. Os Problemas de Segurança em Redes de Computadores

No atual contexto das redes de computadores institucionais e acadêmicas, são inúmeros os alertas recebidos pelos seus administradores comunicando atividades maliciosas originadas em suas redes. Entre os assuntos mais comuns dos alertas estão a disseminação de vírus, ataques de varreduras de portas (*portscan*), entre outros [Estatísticas do CERT.Br 2005]. Nesse contexto, a tarefa dos administradores é a de tomar providências com o objetivo de garantir a integridade e o bom uso dos recursos destas (de acordo com políticas de privacidade e segurança adotadas pelas respectivas instituições). Respostas a certas atividades maliciosas devem ser tomadas em tempo hábil, uma vez que as mesmas podem acarretar em danos morais e/ou materiais.

Há grupos na Internet que são responsáveis por reportar incidentes de segurança em redes, conhecidos genericamente por CSIRT (*Computer Security Incident Response Team*), entre os quais o CERT/CC foi o primeiro e é o mais importante centro [CERT 2005]. No Brasil, a Rede Nacional de Ensino e Pesquisa (RNP) [RNP 2005], responsável por fornecer conectividade às instituições de ensino e pesquisa federais (IEPs), mantém o Centro de Atendimento a Incidentes de Segurança (CAIS) [CAIS 2006], responsável por identificar e notificar problemas de segurança nas redes das IEPs clientes, bem como cobrar que ações corretivas sejam tomadas em tempo hábil.

À medida que cresce o número de usuários das redes gerenciadas, aumenta a dificuldade em identificar responsáveis por atividades maliciosas reportadas nessas redes. Essa dificuldade aumenta significativamente se a atividade maliciosa foi identificada externamente a uma rede que utiliza o mecanismo de *Network Address Translation – NAT* [Network Working Group 1999]. Após identificar o IP de origem da atividade suspeita, também será necessário identificar qual IP interno foi traduzido para o IP identificado, no exato momento em que a atividade suspeita foi reportada.

Nesse contexto, o administrador precisa de uma ferramenta que forneça informações estratégicas e de forma prática sobre o problema que está ocorrendo em sua rede. A próxima seção tem o objetivo de discutir algumas abordagens que estão disponíveis para os administradores de redes, e que visam a obtenção de informações relacionadas à segurança em redes de computadores.

3. Softwares Relacionados

A utilização de Sistemas de Detecção de Intrusão (do inglês *Intrusion Detection System, IDS*) para a identificação de atividades maliciosas é uma tradição entre os administradores de rede, sendo o *Snort* [Snort 2005], uma ferramenta que permite a análise (*sniffing*) de tráfego e registro de pacotes IP em tempo real, a mais utilizada.

Existem poucas ferramentas que trabalham diretamente com as informações geradas pelo Netfilter e DNS. O *Hatchet - pf Firewall Log Parser* [Hatchet 2005], por exemplo, é uma ferramenta para reformatação de arquivos de log escrita para o formato de arquivo de log do p*acket filter* (*pf*), a ferramenta que implementa políticas de *firewall* no sistema operacional OpenBSD. Um projeto que visa o aproveitamento das informações geradas pelo Netfilter é o *Wallfire* [Wallfire 2005]. As ferramentas mantidas por esse projeto, como o *wflogs*, estão relacionadas a reformatação dos registros do arquivo de log, de acordo com parâmetros estabelecidos pelo usuário.

Entre os projetos que visam a geração de estatísticas a partir das informações geradas pelo Netfilter destaca-se o *DShield.org* [Dshield.org 2005]. O intuito desse projeto é o de coletar dados sobre atividades *crackers* no mundo inteiro para a geração de estatísticas sobre os mesmos. Os dados sobre ataques são levantados a partir dos arquivos de log do Netfilter enviados por administradores de *firewall* ao redor do mundo. Ao final do processo de catalogação dos dados são obtidas estatísticas globais sobre *hosts* que mais atacam, portas que mais são atacadas, entre outros.

4. O Software Log Analyzer

O Log Analyzer é uma ferramenta para o correlacionamento de eventos contidos nos arquivos de log gerados pelo Netfilter e pelo DNS, e que está sendo distribuído sob a licença GPL [GPL 2006]. O software é composto de três partes: um servidor, escrito em C originalmente para a plataforma GNU/Linux, que é responsável por receber consultas e repassar ao módulo responsável pelo tratamento dessa consulta; um cliente, escrito em PHP, que irá receber dados fornecidos pelo usuário, construir uma consulta contendo os parâmetros, repassá-la ao servidor, receber o documento contendo o resultado da consulta, extrair informações e gerar relatórios sobre o resultado da mesma; e módulos acopláveis, escritos em C, que fazem o processamento de uma consulta e constroem um documento de resposta baseados nas informações coletadas dos arquivos de log e nos filtros da consulta. Parte do código utilizado neste software foi obtido do livro Advanced Linux Programming [Mitchell, Oldham, and Samuel, 2001].

Toda a comunicação entre as partes do software é feita utilizando XML [Deitel 2003]. Foram elaborados DTDs (*Document Type Definition*) com as regras específicas para a elaboração dos documentos de consulta e de resposta, que serão trocados entre servidor e cliente, durante um processo de consulta.

O DTD que define a estrutura XML das consultas é mostrado na Tabela 1, enquanto que o DTD que contém as regras para a elaboração de documentos XML de resposta é mostrado na Tabela 2.

Tabela 1. DTD de consulta

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT query (option*, field+)>
<!ATTLIST query type CDATA #REQUIRED>
<!ELEMENT field (#PCDATA)>
<!ATTLIST field name ID #REQUIRED>
<!ELEMENT option (#PCDATA)>
<!ATTLIST option name ID #REQUIRED>
```

O servidor, quando recebe uma consulta (*query*), identifica o tipo da mesma (através do atributo *type*) e repassa ao módulo responsável, o qual contém o mesmo nome do valor deste atributo, mais a extensão *.so*. Competirá então ao módulo incumbido de processar a consulta selecionada executar o seguinte algoritmo: validar a consulta, de acordo com o DTD que define a estrutura de uma consulta; processar todos os filtros especificados; selecionar as entradas do arquivo de log que combinam com os filtros; criar um documento de resposta – de acordo com o DTD de resposta; e encaminhar o documento XML de resposta ao cliente que solicitou a consulta.

Cada módulo interpreta um tipo de consulta e arquivo de log específicos. Até o momento foram implementados os módulos *firewall.so*, o qual processa consultas relativas ao arquivo de log do Netfilter, e o módulo *dns.so*, responsável por processar

consultas referentes ao arquivo de log do DNS. Com a implementação de novos módulos, outros formatos de arquivos de log também poderão ser analisados.

Tabela 2. DTD de resposta

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT result (entry)*>
<!ATTLIST result type (firewall|dns|error) #REQUIRED>
<!ELEMENT entry (#PCDATA)>
<!ATTLIST entry date CDATA #IMPLIED>
<!ATTLIST entry time CDATA #IMPLIED>
<!ATTLIST entry src CDATA #IMPLIED>
<!ATTLIST entry dst CDATA #IMPLIED>
<!ATTLIST entry proto CDATA #IMPLIED>
<!ATTLIST entry spt CDATA #IMPLIED>
<!ATTLIST entry dpt CDATA #IMPLIED>
<!ATTLIST entry client CDATA #IMPLIED>
<!ATTLIST entry query CDATA #IMPLIED>
<!ATTLIST entry error CDATA #IMPLIED>
<!ATTLIST entry message CDATA #IMPLIED>
```

O processo de verificação se uma entrada do arquivo de log satisfaz os filtros especificados na consulta é feito baseado em um autômato de saída, mais especificamente uma Máquina de Mealy (que possui saídas associadas às transições), a qual é implementada internamente no software. A palavra a ser reconhecida pelo autômato é uma entrada do arquivo de log, e a saída do autômato é uma *tag* XML que irá compor o documento XML de resposta. Em cada estado do autômato, um atributo da entrada processada é esperado. Se o autômato chegar a um estado final para a entrada dada, a mesma é aceita, e a saída acumulada durante a execução do autômato é adicionada ao documento XML de resposta. Dessa forma, a entrada do arquivo de log do Netfilter contida na Tabela 3 seria formatada como mostrado na Tabela 4.

Tabela 3. Entrada obtida no arquivo de log do Netfilter

```
Jul  4 00:00:00 rnp-gw kernel: RULE 1 -- ACCOUNTING IN=eth0 OUT=eth1
SRC=10.92.1.154 DST=84.155.75.71 LEN=48 TOS=0x00 PREC=0x00 TTL=126
ID=56976 DF PROTO=TCP SPT=3514 DPT=11111 WINDOW=16384 RES=0x00 SYN
URGP=0
```

Tabela 4. Tag XML referente à entrada da Tabela 3

```
  <entry date="Jul  4" time="00:00:00" src="10.92.1.154"
dst="84.155.75.71" proto="TCP" spt="3514" dpt="11111"/>
```

Para a Máquina de Mealy associada a entradas do arquivo de log do DNS, a entrada mostrada na Tabela 5 seria formatada como mostrado na Tabela 6.

Tabela 5. Entrada obtida no arquivo de log do DNS

```
Jul 04 00:00:00.005 client 200.17.51.25#1034: query:
exploited.lsass.org IN A
```

Tabela 6. Tag XML referente à entrada da Tabela 5

```
  <entry date="Jul  4" time="00:00:00" client="200.17.51.25"
query="exploited.lsass.org"/>
```

A consulta é feita de forma seqüencial no arquivo de log. Uma estratégia utilizada para acelerar as consultas é a de manter todas as entradas do arquivo de log com datas iguais (ou seja, todas as entradas que foram geradas no mesmo dia) em um

único arquivo, e fazer com que a data de última modificação do arquivo (*timestamp*) seja igual à data de todas as entradas do arquivo de log. Além disso, manter todos os arquivos de log de um mesmo tipo em um diretório exclusivo. Assim, quando uma determinada data é consultada, basta apenas abrir o arquivo de log que possui o mesmo *timestamp* de última modificação que a data especificada na consulta. Com isso, basta iniciar o servidor passando como parâmetro o diretório que contém os arquivos de log de interesse de um determinado módulo.

O software também utiliza a biblioteca de compressão zlib [zlib 2006], pelo fato de os arquivos de log não correntes serem compactados e *rotacionados* pelo utilitário *logrotate*. Desta forma, não é necessário que os arquivos de log sejam mantidos todos em texto plano, ou então que seja especificado o tipo de arquivo que está sendo processado, uma vez que esta biblioteca é bastante flexível, permitindo que arquivos compactados ou não possam ser processados utilizando as mesmas chamadas.

O correlacionamento de eventos contidos nos arquivos de log do Netfilter e do servidor DNS está implementado da seguinte maneira: quando o usuário efetua uma consulta em um dos dois arquivos de log, os dados desta consulta são incorporados a cada entrada exibida no relatório, substituindo os filtros do usuário pelos valores que foram obtidos da entrada. No momento que o usuário clica em uma linha do relatório exibido, uma nova consulta é efetuada, de tipo análogo ao tipo da consulta exibida, e com filtros iguais ao da linha selecionada. Neste momento é feita uma correlação de campos, e campos que não podem ser relacionados um ao outro entre os dois tipos de consultas são ignorados, sendo neste caso qualquer valor aceito.

5. Validação do Software

O software Log Analyzer foi instalado em um computador de modo a ter acesso a ambos os arquivos de log do servidor DNS e do *firewall* principal da rede. Assim, mensagens de alertas enviadas pelo CAIS aos administradores da rede foram analisadas com o objetivo de identificar os computadores responsáveis pelos alertas. Uma das mensagens utilizada informava que um computador, no dia vinte e nove (29) de novembro de 2005, às 15:31:56 UTC, utilizou um determinado endereço de IP válido na Internet o *worm Win32.Beagle.* Procurando informações sobre o *worm* na Internet, veremos que o mesmo é conhecido por abrir a porta 2535 (TCP) no *host* infectado para escuta e por notificar para alguns *sites* na Internet, através de requisições HTTP *GET /5.php?p=2535&id= HTTP/1*, que o mesmo se encontra ativo. Na mensagem de alerta são emitidas informações sobre as principais características do problema reportado bem como endereços de *sites* nos quais maiores informações podem ser obtidas.

As tarefas a serem realizadas utilizando o Log Analyzer estão definidas: verificar nos arquivos de log do *firewall* todos os computadores que, por volta do horário informado, utilizaram o IP indicado na mensagem (através de tradução de IP por regras de NAT) para acessar algum servidor na porta 80 (www). Em seguida, verificar para cada um dos computadores que aparecerem como resposta, quais acessaram o servidor DNS para pesquisar o endereço IP dos *sites* que são acessados pelo vírus.

6. Conclusões e Trabalhos Futuros

O uso de um software para efetuar o correlacionamento de informações contidas em arquivos de log do sistema pode tornar muito mais simples para o administrador de uma

rede a tarefa de detecção de anomalias e responsáveis por atividades suspeitas. Este artigo apresentou o software Log Analyzer, uma proposta de software para auxiliar administradores de redes em tarefas desse tipo.

Como trabalhos futuros pretendemos expandir o mecanismo proposto para localizar fisicamente um *host*, através da obtenção de informações sobre o mesmo. Uma alternativa é a de utilizar o mesmo mecanismo utilizado pelo software *nbtscan* [Freshmeat.net 2005] para obter mais informações sobre o *host* que está utilizando o endereço, tais como o nome NetBIOS, o usuário que efetuou *logon* no momento e endereço MAC da interface de rede.

Referências

Albitz, P., and Liu, C. (2001) DNS and Bind, Fourth Edition. O'Reilly Media, Inc.

CENTRO DE ATENDIMENTO A INCIDENTES DE SEGURANÇA. Disponível em: http://www.rnp.br/cais. Acesso em: Julho, 2005.

CERT COORDINATION CENTER. Disponível em: http://www.cert.org/. Acesso em: Dezembro, 2005.

Deitel, H. M (2003) XML, como programar, Bookman.

DSHIELD – DISTRIBUTED INTRUSION DETECTION SYSTEM. Disponível em: http://www.dshield.org/. Acesso em: Julho, 2005.

ESTATÍSTICAS DO CERT.BR – INCIDENTES. Disponível em: http://www.cert.br/stats/incidentes/. Acesso em: Dezembro, 2005.

FRESHMEAT.NET: PROJECT DETAILS FOR NBTSCAN. Disponível em: http://freshmeat.net/projects/nbtscan/. Acesso em: Dezembro, 2005.

GNU GENERAL PUBLIC LICENSE. Disponível em: http://www.gnu.org/licenses/gpl.html. Acesso em: Fevereiro, 2006.

HATCHET - PF FIREWALL LOG PARSER. Disponível em: http://www.dixongroup.net/hatchet/. Acesso em: Julho, 2005.

Mitchell, M., Oldham, J., and Samuel, A. (2001) Advanced Linux Programming, New Riders Publishing, First Edition.

NETFILTER/IPTABLES PROJECT HOMEPAGE – THE NETFILTER.ORG PROJECT. Disponível em: http://www.netfilter.org. Acesso em: Fevereiro, 2006.

Network Working Group (1999) IP Network Address Translator (NAT) Terminology and Considerations. Request for Comments: 2663, Agosto.

REDE NACIONAL DE ENSINO E PESQUISA. Disponível em: http://www.rnp.br/. Acesso em: Agosto, 2005.

SNORT. Disponível em: http://www.snort.org. Acesso em: Julho, 2005.

WALLFIRE: WFLOGS. Disponível em: http://www.wallfire.org/wflogs/. Acessado em: Julho, 2005.

ZLIB HOME SITE. Disponível em: http://www.zlib.net. Acesso em: Fevereiro, 2006.