Aperfeiçoamentos do Modelo para Respostas de Detecção de Intrusão Compatível com o Modelo IDWG

Paulo Fernando da Silva, Carlos Becker Westphall

Laboratório de Redes e Gerência (LRG) – Programa de Pós-Graduação em Ciências da Computação (PPGCC) – Universidade Federal de Santa Cataria (UFSC)
Caixa Postal 476 – 88040-970 – Santa Catarina – SC – Brasil

paulo@lrg.ufsc.br, westphal@lrg.ufsc.br

Resumo. *Este artigo apresenta um modelo para interoperabilidade de respostas entre IDSs, compatível com o modelo para interoperabilidade de alertas desenvolvido pelo grupo IDWG. São também apresentados o desenvolvimento e testes do modelo proposto e de seus componentes.*

1. Introdução

Sistemas de detecção de intrusão (*Intrusion Detection Systems* – IDSs) são ferramentas utilizadas na segurança de redes de computadores, estas ferramentas são utilizadas na tentativa de identificar e rastrear ataques às redes de computadores [NIST 2001].

O IDMEF (*Intrusion Detection Message Exchange Format*), especificado pelo IDWG e detalhado em [CURRY 2003], é um modelo de dados que visa possibilitar a troca de informações de alerta entre IDSs. Porém, este modelo não define o formato das respostas trocadas entre os componentes de um IDS. Em [SILVA 2004] estão descritas algumas vantagens obtidas através da especificação de um modelo de respostas para IDSs.

O objetivo deste artigo é possibilitar a interoperabilidade de respostas entre IDSs através de um modelo de dados para formatação das respostas que seja compatível com o modelo já existente de interoperabilidade de alertas.

A seção 2 expõe detalhes do modelo de respostas. Testes realizados no modelo desenvolvido são apresentados na seção 3. A seção 4 apresenta resultados e a conclusão.

2. O Modelo de Respostas IDREF

Os tipos de resposta suportados pelo modelo IDREF são representados pelas classes *Response*, *React* e *Config*, que especializam a classe base do modelo, classe *IDMEF-Message*. A Figura 1 apresenta as classes derivadas e agregadas às classes *Response*, *React* e *Config*.

A classe *Response* permite que sejam enviadas informações com o objetivo de controlar ou avisar sobre um ataque, possuindo três classes derivadas: *TCP*, *ICMP* e *notify*. A classe *TCP* indica que deve ser enviado um pacote TCP pela rede como resposta a um alerta ocorrido, pode ser utilizada para enviar pacotes com *flags* para resetar ou fechar conexões de transporte. A classe *ICMP* indica que deve ser enviada uma mensagem ICMP pela rede como resposta a um alerta ocorrido, pode ser utilizada para enviar mensagens de rede, *host* ou porta não encontrada para a origem do ataque. E

a classe *notify* é utilizada quando se deseja avisar alguém externo à arquitetura do IDS sobre a ocorrência de um ataque.

O modelo IDREF também permite que um recurso seja bloqueado ou finalizado quando está sob um ataque. Para isto é utilizada a classe *React*, esta classe representa uma reação do ambiente contra o ataque. A classe *React* possui duas classes agregadas: *Block* e *Shutdown*. As classes *Block* e *Shutdown* representam respectivamente o bloqueio e o fechamento de algum recurso. Estas duas classes possuem agregada a si a classe *Resource*, que representa um recurso do ambiente.

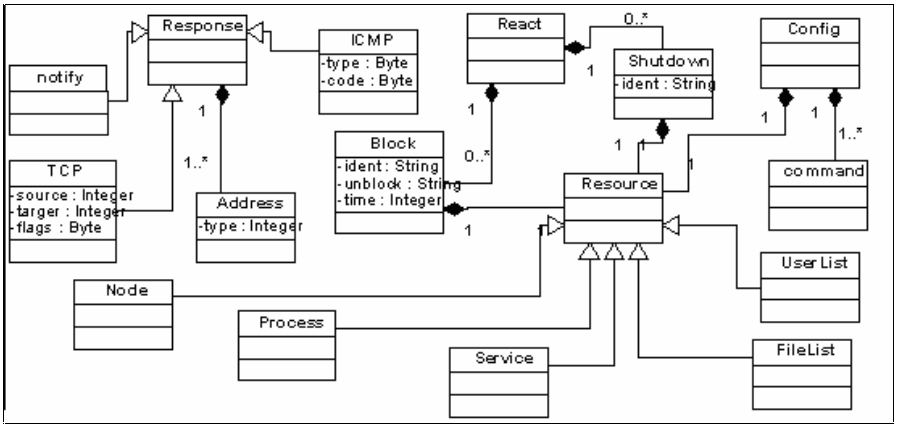

Figura 1 – Detalhamento das classes de resposta.

Como exemplo de respostas do tipo *React* temos o bloqueio de um arquivo do sistema operacional, o bloqueio de um equipamento da rede, o fechamento de uma sessão de usuário ou o fechamento de um processo do sistema operacional.

A resposta do tipo *Config* permite alterar a configuração de um recurso, de modo a conter um ataque. A classe *Config* possui duas classes agregadas: *Command* e *Resource*. A classe *Command* contém o comando a ser executado pelo recurso que será configurado, e a classe *Resource* representa o recurso a ser configurado.

Como exemplo de respostas de configuração temos a alteração de permissões de usuários ou arquivos, a reconfiguração de *firewalls* ou serviços e a ativação de dispositivos auxiliares de segurança.

A classe *Resource* representa um recurso ao qual será aplicada a resposta. Esta classe possui cinco classes derivadas: *Node*, *Process*, *Service*, *UserList* e *FileList*. Isto demonstra que um recurso pode ser um nó da rede, um processo do sistema operacional, um serviço de rede, uma lista de usuários ou uma lista de arquivos.

3. Teste do Modelo Proposto

A Figura 2 apresenta o ambiente criado para o teste do modelo IDREF, onde podemos observar o relacionamento entre os componentes desenvolvidos (IDSMan, IDSAna e IDSRes) e o IDS (IDS Snort) utilizado para realização dos testes. Além dos componentes desenvolvidos, também foi desenvolvida uma biblioteca para geração e leitura de mensagens IDREF em XML.

Para realizar testes nos componentes desenvolvidos foi necessário utilizar um IDS com a capacidade de gerar alertas no formato IDMEF. Para tal tarefa foi utilizado o

IDS Snort. Para que o Snort gerasse alertas no formato IDMEF foi utilizado o "*Snort IDMEF XML plugin*", disponível em *http://www.silicondefense.com*.

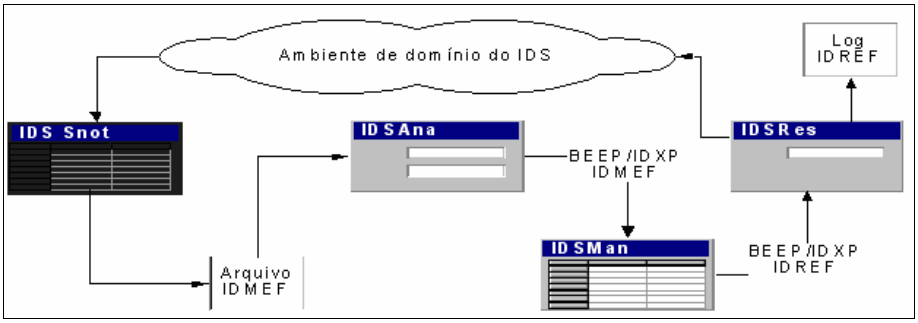

Figura 2 – Componentes desenvolvidos e ambiente de testes.

Com o plugin instalado, quando o Snort gerar um alerta IDMEF no arquivo texto o componente IDSAna irá enviar este alerta ao IDSMan. No gerenciador IDSMan o operador terá a oportunidade de configurar uma resposta IDREF para o alerta recebido e enviar esta resposta ao IDSRes, o qual aplicará no ambiente as ações definidas na resposta IDREF, interrompendo o ataque.

Inicialmente o IDS Snort foi configurado de modo a gerar alertas em caso de ocorrência de qualquer comunicação na rede através do protocolo ICMP. Foi então executado um comando *ping* e o IDS Snort produziu vários alertas, os quais foram capturados pelo IDSAna e enviados ao gerenciador IDSMan.

Podemos observar na Figura 3 a resposta produzida pelo operador no IDSMan e enviada ao IDSRes para conter o ataque em questão.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE IDREF-Message PUBLIC "-//UFSC//DTD IDREF v1//EN"
                               "./idref-message.dtd">
<IDREF-Message ident="1" version="1">
    <description>Resposta teste de validação</description>
    <Manager managerid="MAN1">
        <Node>
            <name>IDSMan</name>
            <Address type="ipv4-addr">192.168.1.3</Address>
        </Node>
        <Process pid="5486">
            <name>IDMan</name>
            <path>c:\IDSMan</path>
        </Process>
    </Manager>
    <CreateTime ntpstamp="0xc3ccc189.0x01400000">2004-02-05T11:20:09Z</CreateTime>
    <alertident>0</alertident>
    <React>
        <Block ident="1" time="30" unblock="time">
            <Node>
                <location/>
                <name/>
                <Address type="ipv4-addr">192.168.1.1</Address>
            </Node>
        </Block>
    </React>
</IDREF-Message>
```

Figura 3 – Resposta recebida pelo IDSRes.

O elemento *React* contém um único elemento *Block*, cujos atributos especificam que o recurso deve ser bloqueado por 30 minutos. O elemento *Node* contém dados do recurso a ser bloqueado, que correspondem ao *host* que estava sendo atacado.

O componente IDSRes, ao receber a resposta, aplica a mesma ao ambiente e a grava no arquivo de *log*. Verifica-se então que todas as informações configuradas na resposta IDREF do gerenciador chegaram ao componente IDSRes corretamente e foram também corretamente aplicadas no ambiente de domínio do IDS.

4. Resultados e Conclusão

Analisando as características do modelo de dados IDREF e da arquitetura de IDSs desenvolvidos, observamos que os mesmos foram cuidadosamente projetados de modo a serem compatíveis com o modelo de alertas IDMEF e a arquitetura de IDSs desenvolvida pelo grupo IDWG.

O modelo de dados IDREF foi projetado de forma a utilizar o máximo de informações possíveis do modelo IDMEF. Por exemplo, uma resposta IDREF tipo *Response* pode ser enviada para conter a origem de um ataque quando o alerta IDMEF contém informações sobre o endereço de origem do ataque.

Outra característica que demonstra o relacionamento entre os modelos é observada entre as classes *Target* do modelo IDMEF e *Resource* do modelo IDREF. A classe *Target* pode conter as classes *Node*, *Process*, *Service*, *User* e *FileList*. Em uma resposta, as informações destas classes poderão ser convertidas em recursos do tipo *Node*, *Process*, *Service*, *UserList* e *FileList*. Neste caso, o alvo de um ataque especificado no alerta IDMEF se transforma em um recurso ao qual será aplicada uma resposta IDREF.

A implementação dos componentes IDSMan, IDSAna, IDSRes e da biblioteca IDREF pode ser utilizada para demonstrar e testar o modelo de dados IDREF e a arquitetura proposta. Também pode ser utilizado para demonstrar e testar os trabalhos desenvolvidos pelo grupo IDWG, pois o contempla o modelo IDMEF.

Como sugestão para trabalhos futuros tem-se: extensão do modelo IDREF proporcionando o suporte a outros tipos de respostas; avaliação do modelo IDREF junto a outros IDSs; e melhoria nos componentes desenvolvidos.

Bibliografia

CURRY, D.; DEBAR, H. **Intrusion Detection Message exchange format data model and Extensible Markup Language (XML) Document Type Definition.** Draft-ietf-idwg-idmef-xml-10, Janeiro de 2005. Disponível por http://www.ietf.org/internet-drafts/draft-ietf-idwg-idmef-xml-14.txt. Acesso em 22 Abr. 2005.

NIST - National Institute for Standards and Technology. **Special Publication on Intrusion Detection Systems**. Nov. 2001. Disponível por http://csrc.nist.gov/ publications/nistpubs/800-31/sp800-31.pdf. Acesso em 10 de Abril 2005.

SILVA, Paulo F.; WESTPHALL, Carlos B. **Um Modelo para Interoperabilidade de Respostas em Sistemas de Detecção de Intrusão.** 23º Simpósio Brasileiro de Redes de Computadores (SBRC 2005). Fortaleza - CE, Maio de 2005.

SILVA, Paulo F.; WESTPHALL, Carlos Becker. **Analysis and Extension of the IDWG Group Intrusion Detection Model.** Proceedings of the International Workshop on Dependable Embedded Systems with 23rd Symposium on Reliable Distributed Systems (SRDS 2004), Florianópolis - SC, Outubro de 2004.