Especificação de Agentes de Captura para Sistemas Detectores de Intrusão

Dalton Matsuo Tavares[*]**, Mauro César Bernardes, Edson dos Santos Moreira**
Instituto de Ciências Matemáticas e Computação
Universidade de São Paulo
Av. do Trabalhador São-Carlense, 400
Centro - Cx. Postal 668 CEP 13560-970
São Carlos - São Paulo - Brasil
{dmatsuo,mauro,edson}@icmc.sc.usp.br

Stenio Firmino Pereira Filho
Centro de Computação Eletrônica – CCE
Universidade de São Paulo
Av. Prof. Luciano Gualberto, Travessa 3 nr 71
Cidade Universitária 05508-900 São Paulo – SP
stenio@usp.br

Resumo: A captura de pacotes é usada por administradores de rede para coletar dados relativas ao comportamento do usuário e da rede. Um problema comum enfrentado hoje em sistemas detectores de intrusão (SDIs) *network based*, concerne o uso de sistemas de captura de pacotes em grandes redes segmentadas. O objetivo deste artigo é superar essa limitação através de um sistema composto por agentes de captura estáticos, integrados a especificação de um SDI baseado no paradigma de agentes móveis.

Palavras-chave: captura de pacotes (*sniffers*), sistemas detectores de intrusão (SDIs), *switches*¸ monitoramento de redes.

Abstract: *The packet capture approach is used by network administrators to gather information regarding the user and network behavior. A common problem faced today in intrusion detection systems (IDSs) network based, concerns the use of existing packet capture systems in large segmented networks. The objective of this paper is to overcome this limitation by means of a system composed by static capture agents, integrated to the specification of an IDS based on the paradigm of mobile agents.*

Keywords: *packet capture (sniffers), intrusion detection systems (IDSs), switches, network monitoring.*

1. Introdução

A aplicação de um sistema de captura de pacotes (*sniffer*) em uma rede `ethernet` limita-se a um segmento de rede, e é fisicamente representado através da utilização de *switches* e roteadores. Com isso, um problema corriqueiro enfrentado pelos administradores refere-se ao uso de sistemas de captura de pacotes em grandes redes segmentadas em diversos domínios de colisão.

Um domínio de colisão pode ser descrito como o compartilhamento da banda de rede disponível entre todos os equipamentos, ou seja, todas as interfaces de rede recebem todos os frames transmitidos em um segmento onde uma colisão se propaga. Em um *hub* o domínio de

[*] Bolsista CAPES

colisão é o seu barramento, em um *switch* o domínio de colisão está associado à porta, independendo da configuração do equipamento assim como no roteador.

É comum confundir os conceitos envolvidos na definição de domínio de colisão e domínio de *broadcast*. Seus limites dependem do tipo de dispositivo utilizado na segmentação da rede.

Em uma rede segmentada por roteadores e *switches*, o tráfego capturado pelos *sniffers* fica limitado a porta do equipamento, tornando-se necessária a utilização de *sniffers* em diversas portas. Em redes com diversos roteadores e *switches* esta tarefa torna-se nada trivial. Esses equipamentos impõem níveis de hierarquia na rede tornando determinados segmentos inacessíveis, impedindo a troca de informações entre as diversas sondas posicionadas.

É nesse contexto que se encontra inserido o escopo do presente artigo. Seu objetivo é especificar um sistema de captura de pacotes (*sniffer*) aplicável a redes de computadores divididas em diversos segmentos (nível 2 e 3). Isso será viabilizado através da utilização da tecnologia de agentes móveis, seguindo a arquitetura do sistema detector de intrusão (SDI) definida por [BERNARDES & MOREIRA, 2000]. Além disso, serão abordados métodos de captura de pacotes em equipamentos *core*.

2. Arquitetura do sistema

A arquitetura proposta é baseada em uma especificação modular implementada em um ambiente composto por agentes móveis e agentes estáticos. A grosso modo, os agentes móveis da primeira camada desempenham o papel de 'canal de comunicação assíncrono' entre os diversos agentes de captura (estáticos) situados em segmentos de rede distintos. A descrição a seguir utiliza os resultados obtidos no trabalho de mestrado de [BERNARDES, 1999] onde é fornecida a descrição de uma arquitetura modular baseada em agentes móveis para aplicação em sistemas detectores de intrusão (SDIs). Esta arquitetura representa a base para a solução desejada (um sistema de captura móvel e distribuído aplicável a redes chaveadas).

2.1 Modelo em camadas

Segundo o modelo de camadas, o sistema é composto por um conjunto de pequenos processos (agentes) que podem agir independentemente no sistema em construção. São projetados para se moverem pelo ambiente no qual estão inseridos, observando o comportamento do sistema computacional alvo. Além disso, os agentes cooperam entre si via

passagem de mensagens, notificando quando uma ação é considerada suspeita, podendo executar ações reativas.

Uma vez que os agentes são independentes, eles podem ser adicionados e removidos do sistema dinamicamente, dispensando a necessidade de um processo de adaptação penoso de todo o sistema ou, a interrupção de sua atividade. Assim, a qualquer sinal de identificação de uma nova forma de ataque, novos agentes especializados podem ser desenvolvidos, acoplados ao sistema e configurados para atender uma política de segurança específica.

O principal conceito relacionado ao sistema é a simplicidade. Cada agente é uma entidade simples que irá desempenhar uma atividade específica e cooperar com outros agentes da forma mais eficiente e flexível possível. Quando uma atividade for considerada suspeita por um agente, ele irá comunicar aos demais agentes do sistema sua suspeita de uma possível intrusão. Neste momento, será acionado um agente (ou um conjunto de agentes) com um maior grau de especialização naquele tipo de suspeita.

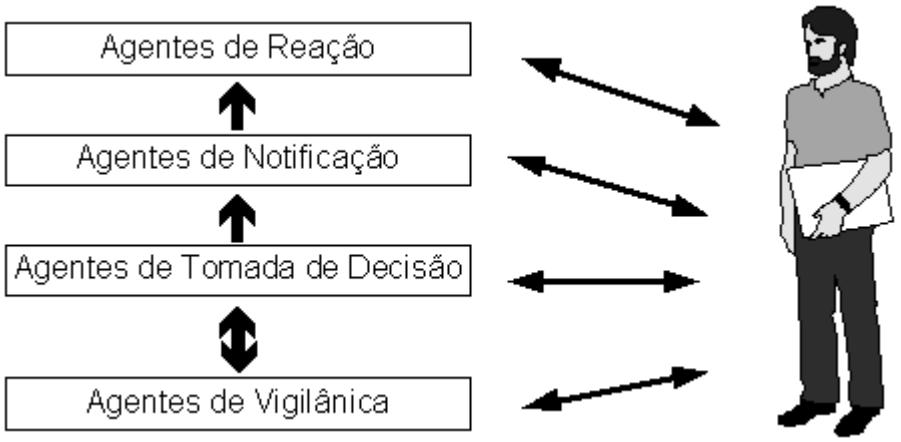

Figura 2.1 - Modelagem em Camadas para o Sistema Proposto [BERNARDES & MOREIRA, 2000].

A Figura 2.1 apresenta a arquitetura (modelo em camadas) para o SDI proposto [BERNARDES & MOREIRA, 2000]. Uma descrição parcial do sistema será fornecida a seguir.

Camada 1 – Agentes de Vigilância

Esta camada representa o primeiro nível de agentes do sistema proposto. É composta por agentes estáticos, responsáveis pela captura de pacotes e agentes móveis, responsáveis pela busca inicial por comportamento suspeito e transporte dos agentes de captura ao longo da

rede. Caso exista a identificação parcial de comportamento malicioso, notifica a camada superior e oferece a opção de informar o administrador.

Camada 2 – Agentes de Tomada de Decisão

Nesta camada, encontram-se os agentes que exercem todas as funções de tomada de decisão no sistema (motor de análise), constituindo a 'inteligência' do sistema. Um agente desta camada irá receber uma mensagem ou um conjunto de dados dos agentes da camada inferior (a camada de vigilância) e, com base em uma análise criteriosa dessas informações, poderá identificar uma intrusão (ou tentativa) no momento de sua ocorrência. Também pode acionar novos agentes de vigilância para a coleta de informações complementares.

Maiores informações sobre a terceira e quarta camadas (agentes de notificação e agentes de reação) podem ser encontradas em [BERNARDES & MOREIRA, 2000]. A tecnologia utilizada será descrita em maiores detalhes nas próximas seções..

Como já foi mencionado anteriormente, um dos maiores problemas nas redes é a complexidade crescente das mesmas. Sua arquitetura tem se tornado mais hierarquizada resultando na conseqüente separação dos domínios de colisão. Apesar dos benefícios inerentes ao emprego de novos equipamentos (como *switches*, roteadores, etc.), a aplicação de técnicas de captura para o monitoramento de redes tem se tornado cada vez mais ineficaz.

A maior polêmica em torno da tecnologia de *sniffing* gira em torno de sua aplicação em dispositivos de chaveameto (*switches*). Ao contrário de um HUB, um *switch* verifica a quem o pacote é endereçado e estabelece uma conexão particular entre os participantes da sessão. O receptor recebe o *frame* que lhe é destinado e examina o endereço para determinar se, de fato, aquele pacote lhe pertence.

Embora essa dupla verificação cause uma pequena sobrecarga no *host*, esse é um processo necessário para evitar a migração não autorizada de máquinas de um segmento de rede a outro. Com isso, conclui-se que uma rede chaveada não pode ser monitorada através do emprego de mecanismos de *software* tradicionais utilizados em redes *broadcast*, pois não ocorre compartilhamento de tráfego no meio. Perceba também que, dentre os benefícios supracitados, o acréscimo ao nível de segurança da rede não é a característica mais importante. Caso o *switch* não seja configurado de maneira apropriada, ele se torna um meio tão vulnerável quanto uma rede compartilhada.

É possível explorar as vulnerabilidades[1] de um dispositivo mal configurado e, obter como resultado um efeito similar ao obtido por um *sniffer* em ambiente *broadcast*. Essa possibilidade não será discutida no escopo deste artigo, pois em um ambiente de produção (i.e. uma rede com um tráfego intenso), essa solução perde sua validade[2].

3.1 Mecanismo de captura desenvolvido para *switches*

Percebe-se que não é possível recorrer à mesma metodologia usada em ambientes *broadcast* para o desenvolvimento de sondas de captura de pacotes para uso efetivo em *switches*. Uma solução puramente implementada em *software* seria ineficiente para tratar o volume de tráfego presente em uma rede chaveada.

Uma solução mista que utilize mecanismos de *hardware* (inerente ao dispositivo) e *software* seria o mais recomendado. Portas de captura (port mirroring) são mecanismos implementados em *hardware* com o objetivo de redirecionar um subconjunto do tráfego do *switch* a um determinada porta. Acrescentando um mecanismo de comunicação baseado em agentes móveis (como descrito na seção anterior) é possível construir um sistema flexível contendo uma descrição fiel do *status* de segurança dos diversos segmentos da rede.

- **Portas de captura**

Atualmente o grau de gerenciamento dos *switches* chegou a um patamar no qual atividades complexas de gerenciamento podem ser implementadas no próprio dispositivo ou através da instalação de módulos dedicados[3] (módulos de *hardware* acopláveis).

De um modo geral, o mecanismo de *software* para análise de tráfego é baseado nas técnicas utilizadas em redes *broadcast*. Um sensor, ajustado em modo promíscuo, é acoplado a uma porta de captura para a qual todo o tráfego (ou parte dele) é redirecionado.

O método de portas de monitoramento consiste em espelhar o tráfego do *backplane* ou um subconjunto dele e enviá-lo a uma porta específica. O sensor acoplado a porta de captura possui um ambiente de agentes com o objetivo de efetuar a sintonia fina do tráfego sendo analisado.

O mecanismo de porta de captura pode utilizar um protocolo proprietário para o encapsulamento dos *frames*[4] ou o padrão `802.1Q`. Independente da forma de encapsulamento,

---

[1] Alguns exemplos de ataques realizados em *switches* são: ARP *spoofing*, MAC *flooding* e MAC *duplicating* [SIPES, 2000].
[2] Para maiores detalhes consulte http://naughty.monkey.org/~dugsong/dsniff/ (visitado em 28/01/2002).
[3] Como exemplo: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids5/csidscog/overview.pdf (visitado em 28/02/2002)
[4] Como o padrão ISL (*Inter-Switch l ink*). Mecanismo proprietário da Cisco para encapsular pacotes com informações de VLAN e prioridade quando transmitidos pelo *backbone*.

este mecanismo torna possível o envio de pacotes pertencentes a múltiplas VLANs multiplexados em uma única conexão do *switch* para envio a uma sonda acoplada a porta de monitoramento. Este artifício também pode ser usado para expandir a influência de uma VLAN a outros *switches*[5]. Isso permite que dois dispositivos se comportem como uma única entidade lógica.

3. Agente móvel de captura

A característica principal do agente móvel de captura proposto [BERNARDES & MOREIRA, 2000] reside na obtenção de informações utilizando um sistema de análise com inteligência limitada, fornecendo apenas um subconjunto da informação capturada (informações de cabeçalho) aos agentes inteligentes situados na camada superior (motor de análise). Esses agentes inteligentes irão avaliar as informações obtidas, funcionando como 'gatilhos' para a ativação de outros agentes de captura com filtros ajustados de forma a refinar suas conclusões[6].

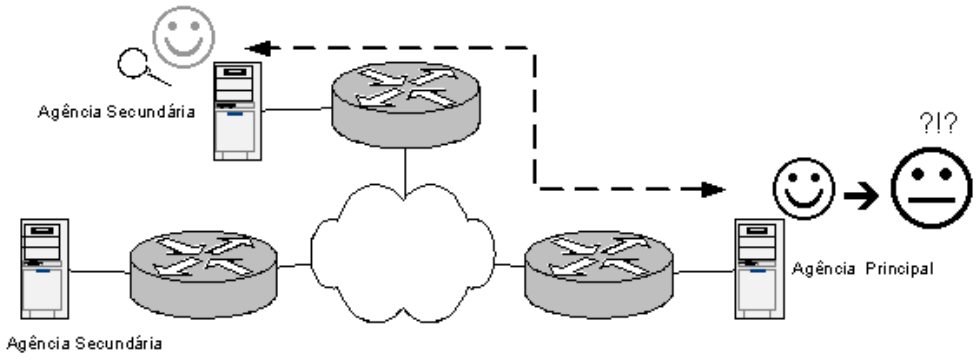

Figura 3.1 - Exemplo de uma execução do agente de captura

O *sniffer* descrito na seção anterior é anexado a um agente móvel e configurado com filtros pré-determinados (que caracterizam o tipo de protocolo monitorado, o serviço utilizado, etc.) pelo administrador do sistema ou pelos agentes da segunda camada. A função desse filtro é selecionar informações essenciais para o motor de análise evitando o exame desnecessário de informação pelo agente móvel, acarretando uma possível lentidão em sua transmissão. À medida que o motor de análise necessitar de informações mais específicas, novos agentes de captura são requisitados.

---

[5] Esta prática é denominada *truncking* pela Cisco. Para maiores informações vide: www-1.cisco.com/warp/public/473/27.html (visitado em 04/02/2002)

[6] No contexto deste artigo serão discutidas apenas as relações entre camadas pertinentes ao agente de captura.

No exemplo da Figura 4.1, o agente é ativado pelo administrador ou por um agente responsável (situado na camada de vigilância ou na camada superior) e enviado a uma agência situada em outro segmento de rede. Após a captura das informações e constatação de algum evento suspeito, o agente retorna à agência de origem e transmite os dados obtidos ao motor de análise. Dependendo da situação e das informações capturadas, o agente inteligente decidirá o curso de ação a ser seguido. Dentre as opções viáveis, pode ocorrer o envio de outros agentes de captura configurados com filtros apropriados, ou ainda, o acionamento de outros agentes inteligentes especializados.

Na implementação destes agentes, foi utilizado o ASDK[7] (*Aglets Software Development Kit*) escrito em Java. Um dos problemas enfrentados refere-se a interação do agente (aglet) com o dispositivo de rede; em outras palavras, o desenvolvimento de um agente captura implementado completamente em Java. Como isso não foi possível[8], a solução foi criar um *sniffer* em C inspirado no tcpdump[9].

Esse *sniffer* é posicionado na estação de captura e posteriomente disparado por um agente móvel contendo os filtros descritos, compondo o agente de captura. O papel do agente móvel é efetuar a pré-seleção dos dados coletados pelo *sniffer* e procurar no fluxo de pacotes, eventos que caracterizem falhas de segurança potenciais. Isso é feito atualmente através de estruturas de decisão inspiradas no trabalho desenvolvido em [Cansian, 1997]. A comunicação entre o agente de captura estático e o agente móvel é realizada via sockets. Existem outras opções para integração de código C e Java; como por exemplo *Java Native Interface* (JNI). Com a descoberta da jpcap, é um pouco prematuro optar por integrar de maneira mais forte o agente estático e o agente móvel. Esse esforço pode ser desnecessário caso seja comprovado que o uso da jpcap seja mais recomendado que o uso da libpcap.

4. Conclusão

Este trabalho descreveu a especificação de um agente de captura para ambientes de rede segmentados. Para elaborar essa solução, foi utilizada a API para implementação de agentes ASDK em combinação com um *sniffer* de rede para captura de pacotes implementado

[7] Para maiores informações: http://www.trl.ibm.com/aglets/ (visitado em 28/01/2002)
[8] Essa afirmação não pode mais ser considerada verdadeira devido ao surgimento de uma biblioteca de captura de pacotes implementada em Java. Infelizmente, não foi possível efetuar testes de desempenho e integração ao sistema devido a época de sua descoberta pelo autor. Fonte: http://jpcap.sourceforge.net (visitado em 06/04/2002).
[9] Pode ser obtido em ftp://ee.lbl.gov/tcpdump.tar.Z (visitado em 28/01/2002)

em linguagem C. A principal vantagem desse método em relação aos mecanismos de captura existentes é a mobilidade e flexibilidade herdada da tecnologia de agentes móveis e a possibilidade de gerenciamento em múltiplos segmentos.

A questão do filtro implementado no agente de captura estático tem por objetivo minimizar os problemas causados pelo grande fluxo de informações em equipamentos *core*. Além disso, cada agente de captura tem sua contraparte móvel responsável pela seleção preliminar dos pacotes capturados. As tecnologias de p*ort mirroring e* placas *probe* já se encontram disponíveis para a maioria dos equipamentos de grande porte e em alguns *switches* menores. Ainda estão em andamento estudos sobre a utilização de SNMP para a seleção dinâmica de grupos de porta que terão o tráfego espelhado, possibilitando uma maior flexibilidade para o sistema de captura móvel.

Referências Bibliográficas

[Bernardes, 1999] Bernardes, M. C. (1999). Avaliação do uso de agentes móveis em segurança computacional. Dissertação de mestrado, ICMC/USP.

[Bernardes & Moreira, 2000] Bernardes, M.C. & Moreira E.S. A proposal for Intrusion Detection Systems Based on Mobile Agents. Proceedings of Fifth Int Symposium on Parallel and Distributed Engineering, eds Nixon and Ritchie, Pub IEEE CS Pres, 2000.

[Cansian, 1997] Cansian, A. M. (1997). Desenvolvimento de um Sistema Adaptativo de Detecção de Intrusos em redes de Computadores. Tese de doutorado, Instituto de Física de São Carlos - USP.

[Cisco, 2002] Cisco. Cisco Netacad Program – Semester 1 v2.1.1. Cisco Press, 2002.

[Crosbie & Spafford, 1995a] Crosbie, M. & Spafford, E.H. Active Defense of a Computer System using Autonomous Agents. Departament of Computer Science, Purdue University, 1995. (Technical Report CSD-TR-95-008).

[Crosbie & Spafford,1995b] Crosbie, M. & Spafford, E.H. Defending a Computer System using Autonomous Agents. Departament of Computer Science, Purdue University, 1995. (Technical Report CSD-TR-95-022; Coast TR 95-02).

[Lange & Oshima, 1998] Lange, D.B; Oshima, M. Programming And Deploying Java Mobile Agents with Aglets. Addison Wesley Longman, Inc. 1998.

[Zamboni et al., 1998] Zamboni, D., Balasubramaniyan, J., Garcia-Fernandes, J.O., Spafford E.H. An architecture for intrusion detection using autonomous agents. Departament of Computer Science, Purdue University; Coast TR 98-05; 1998.

[Sipes, 2000] Sipes, S. (2000). Intrusion detection faq, why your switched network isn't secure. SANS Institute resouces.