Uma Proposta de um Sistema para a Prevenção de Intrusões em Redes de Computadores

Guilherme Linck, Diego Luís Kreutz

[1]Núcleo de Ciência da Computação — NCC
Universidade Federal de Santa Maria — UFSM

{linck,kreutz}@inf.ufsm.br

Abstract. *This article presents the architecture of a simple system, that can easily be distributed, for intrusions prevention on computer networks. The basic idea is to adapt monitoring modules, to resources as package filters, that are able to identify and block probable attackers.*

Resumo. *Este artigo apresenta a proposta de um sistema simples, que pode ser facilmente distribuído, para a prevenção de intrusões em redes de computadores. A idéia básica é adaptar módulos de monitoramento, a recursos como filtros de pacotes, que sejam capazes de identificar e barrar prováveis atacantes.*

1. Introdução

Atualmente existem diversas ferramentas que auxiliam na detecção de intrusão, porém, a área de prevenção de intrusões é relativamente nova e carece de soluções [Desai 2003, Holland 2004]. Nexte contexto, este trabalho apresenta uma proposta de um sistema que busca auxiliar de forma simples e prática a prevenção de intrusões.

A idéia básica é fazer uso dos filtros de pacotes existentes, acrescentando monitores de *log* que irão analisar os registros armazenados e tomar as medidas preventivas cabíveis para interromper uma ação suspeita. Outro objetivo é permitir um controle integrado de diversos sistemas de uma rede local ou corporativa, possibilitando a identificação e prevenção distribuída de prováveis tentativas de ataque contra sistemas.

O artigo está estruturado da seguinte forma: a seção seguinte contém alguns trabalhos relacionados. A arquitetura do proposta é apresentada na seção 3. Na seção 4. é abordado o estado atual. Finalizando, seguem a conclusão e os trabalhos futuros.

2. Trabalhos Relacionados

Existem basicamente duas grandes classes de sistemas para a segurança em uma rede de computadores. A primeira delas, mais antiga, é composta pelos sistemas de detecção de intrusão (IDS[1]). Estes sistemas permitem um monitoramento e análise de uma vasta gama de dados e ações em sistemas ou contra a própria rede. Os dados coletados permitem a detecção de anomalias na rede ou em sistemas locais.

Uma segunda técnica que vem crescendo nos últimos anos é a prevenção de intrusões [Holland 2004]. Segundo uma das definições encontradas para essa classificação mais recente [Desai 2003], um sistema de prevenção de intrusões (IPS [2]) pode ser a combinação de um *firewall* e um IDS especializado na análise de pacotes de rede. A

[1]Intrusion Detection Systems
[2]Intrusion Prevention System

idéia básica é prever ataques conhecidos ou desconhecidos e tentar impedir a realização dos mesmos.

Ferramentas como o *firestarter* [Junnonen 2000], e *portsentry* [Rowland 2002] costumam ser utilizadas para a proteção de estações de trabalho contra possíveis ataques pela rede. Algumas destas ferramentas possuem interface gráfica, facilitando a manipulação de regras e controle geral de acesso por parte do usuário. Outros sistemas como o *firestorm* [Leach and Tedesco 2004], *Snort* [Caswell and Roesch 2004] e *prelude-nids* [Prelude Trac 2005] são destinados ao monitoramento de serviços e detecção de intrusões e anomalias em máquinas ou na rede. A funcionalidade básica é identificar situações conhecidas e desconhecidas (anormais/suspeitas), gerando relatórios e alertas de estado da rede ou tomando medidas preventivas baseadas em políticas pré-definidas. Dentro deste contexto, algumas dessas ferramentas possuem funcionalidades como bloqueio e restrição de acesso, impedindo ações posteriores de máquinas suspeitas. Esse é o caso do *portsentry* e o *Snort-inline*.

O sistema aqui proposto tem como abordagem principal a prevenção, de forma centralizada ou distribuída/colaborativa, de intrusões à redes de computadores ou sistemas locais.

3. A Arquitetura Proposta

A idéia básica é criar um sistema prático e útil para a prevenção de intrusões em redes de computadores. Uma das metas é manter a arquitetura tradicional de sistemas de segurança, como filtros de pacotes, e acrescentar módulos auxiliares que irão incrementar funcionalidade nesses sistemas, de modo a impedir, protelar, ou desviar o trabalho (que pode ser uma tentativa de ataque) de endereços suspeitos.

A figura 1 apresenta uma visão geral da base do sistema. Esta é composta por: (1) um filtro de pacotes; (2) um arquivo de registros de atividade sobre as políticas de acesso ou tráfego na rede; e (3) um monitor de atividades registradas no arquivo de *log* e gerenciador de políticas de controle de acesso e roteamento do filtro de pacotes.

Os três componentes básicos executam na mesma máquina. Esta pode ser uma simples estação de trabalho, um servidor ou um *firewall* da rede.

O objetivo do sistema é tentar identificar e bloquear máquinas suspeitas, que possam ser prováveis atacantes à rede. A primeira função cabe ao monitor de registros do filtro de pacotes. Em uma primeira instância, através da análise das tentativas de conexões não

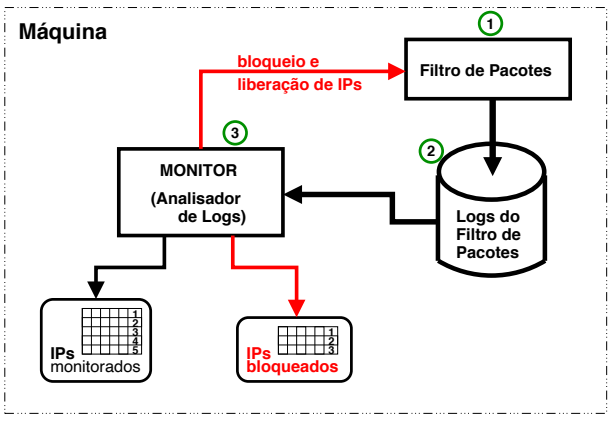

Figure 1. Visão geral do sistema em uma máquina

autorizadas [Kreutz 2004] o monitor irá classificar os computadores, tanto da rede externa quanto da rede interna. Todos os IPs das máquinas que infringiram alguma regra do filtro de pacotes são colocadas em uma tabela geral de monitoramento. A partir desse ponto o agente monitor começa a analisar o número e a freqüência de conexões não autorizadas de cada endereço de origem. A meta é identificar endereços com uma alta probabilidade de serem fontes de futuros ataques e bloqueá-los, ou desviá-los a um *honeypot*.

Uma máquina é considerada suspeita quando atingir o limite máximo de infrações permitidas [Kreutz 2004]. Neste caso, infrações estão relacionadas às entradas do arquivo de *logs* que registram tentativas de conexão, a serviços ou máquinas, negadas (não autorizadas). Outra forma de um IP ser enquadrado como uma possível ameaça à rede é quando a probabilidade é bastante grande de o endereço de origem estar realizando uma varredura de portas ou máquinas da rede. Uma terceira maneira de um endereço ser considerado suspeito é quando sua atividade denota freqüências ou comportamentos de tempos de conexão (seguindo regras autorizadas) constantes, anômalos, ou reconhecidas como táticas de invasão (a partir da análise de casos anteriores).

Quando uma máquina entra para a lista de prováveis atacantes (ou lista negra do sistema) ela é automaticamente colocada em estado de observação, que pode ser: bloqueio no filtro de pacotes, ou redirecionamento dos pacotes para um *honeypot*. Com isso, espera-se impedir que muitos ataques, em relação a redes sem nenhum mecanismo de prevenção desse gênero, sejam iniciados, prossigam ou tenham sucesso.

Um endereço IP é retirado da lista negra quando satisfizer a seguinte condição: não ter cometido novas infrações nos últimos X intervalos de tempo. Esses intervalos de tempo são parâmetros de configuração do sistema, determinados pelo administrador da rede.

Para o funcionamento dessa arquitetura base da proposta são necessários os seguintes requisitos: **(1)** número mínimo de infrações para que um endereço seja enquadrado como suspeito; **(2)** intervalo de tempo de monitoramento[3]; **(3)** número mínimo de conexões negadas à n diferentes máquinas ou portas em um determinado intervalo tempo (normalmente reduzido); **(4)** valor de n contido no requisito anterior; **(5)** intervalo de tempo contido no requisito 3; **(6)** tempo mínimo de bloqueio[4];

O primeiro requisito define o número mínimo de infrações para considerar uma máquina como uma possível ameaça à rede. Todas as máquinas que venham a cometer mais infrações que o número mínimo definido, no intervalo de tempo estipulado, entrarão para o estado de observação. Além disso, o administrador do domínio do endereço de origem (quando definido no DNS) será notificado por e-mail do fato de uma máquina sob sua jurisdição ter sido considerada suspeita e estar em observação. Isso possibilita que administradores tomem providências e verifiquem o fato.

A definição do número mínimo de infrações pode ser feita de acordo com análises do tráfego e ações contra a rede local ou sistema em monitoramento[Kreutz 2004]. Cada grupo de máquinas pode possuir seu próprio número mínimo de infrações. Onde, por exemplo, servidores da rede tem um nível de tolerância maior, evitando bloqueios por problemas de má configuração ou estado de teste dos mesmos.

As infrações de cada endereço IP são cumulativas e verificadas a cada intervalo de

[3]Tempo durante o qual o número mí nimo de infrações deve acontecer

[4]Tempo em que o endereç o fi cará bloqueado ou terá seu tráfego desviado a uma máquina isca

monitoramento. A definição destes intervalos pode ser determinada e programada pelo nível de atividade do sistema [Kreutz 2004]. Com excessão dos casos que se enquadram nos requisitos 3 e 4. Estes casos são verificados com uma periodicidade maior, dinâmica.

O terceiro requisito remete ao número mínimo de conexões negadas oriundas de um endereço IP qualquer, cujo destino é um determinado número de máquinas ou portas diferentes em um espaço de tempo reduzido. É o caso comum de escaneamentos de endereços da rede ou portas de máquinas. Um escaneamento é normalmente encarado como uma primeira premissa de um possível ataque ou ação mal intencionada na rede.

O quarto requisito consiste em definir o número de portas ou máquinas diferentes necessário para verificar se um endereço IP infringe ou não o terceiro requisito. Esta variável poderá ter um valor reduzido, visto que um escaneamento é comumentemente realizado em um período curto de tempo. Por exemplo, é estabelecido que 10 tentativas de conexão não autorizadas com portas ou máquinas distintas em um período de 0 à 60 segundos é enquadrado como uma infração às políticas do sistema. Assim que as 10 tentativas forem detectadas, em um período de no máximo 1 minuto, o endereço irá para o estado de observação. O intervalo de 1 minuto é definido pelo próximo requisito do sistema, que pode também ser infinito.

O último requisito remete ao tempo mínimo que um endereço IP ficará em observação no sistema. Após transcorrido esse tempo e nenhuma nova infração ter sido registrada, o endereço é liberado. No entanto, entra para um segundo estado de observação. Neste estado, o limite de infrações mínimas, e o período de tempo para verificação destas, é reduzido em $Y\%$ (definidos pelo usuário). Essa medida tem como objetivo evitar casos em que o endereço em questão esta mais cauteloso, tentando reduzir sua periodicidade, número de varreduras ou tentativas de ataque a sistemas da rede. A cada período de tempo (requisito 2) essa maior sensibilidade de monitoramento irá sendo reduzida $Z\%$ (parâmetro de configuração do sistema), até atingir o estado normal.

3.1. Controle Distribuído

Outra meta é permitir uma prevenção distribuída e integrada. A simples detecção de um endereço suspeito deverá ser rapidamente propagada para as demais instâncias do sistemas, possibilitando uma ação conjunta e mais efetiva, evitando possíveis futuros ataques e reduzindo a eficácia de ataques em andamento. A figura 2 ilustra essa arquitetura distribuída de prevenção e controle de intrusões.

Em uma rede local, por exemplo, essa estrutura pode ser utilizada para impedir ações maliciosas internas à rede. Sabe-se que os ataques mais perigosos comumentemente são os internos [Holland 2004]. Logo, mecanismos que auxiliam no controle cooperativo das diretivas de segurança de servidores e máquinas da rede podem ser uma boa política para aumentar a efetividade na prevenção e combate desses ataques.

A inclusão de um IP na lista de observação, em um dos pontos de prevenção de intrusões, irá implicar o monitoramento deste endereço em todos os pontos que fazem parte do sistema distribuído de prevenção de ataques. Ao mesmo tempo, a liberação de um determinado endereço será propagada aos demais membros da rede de segurança preventiva. No entanto, a liberação só ocorrerá mediante a satisfação dos pré-requisitos de liberação do sistema, em cada ponto de controle, vistos anteriormente.

3.2. Exemplo de Aplicação

A figura 3 apresenta um caso de uso da arquitetura distribuída. É ilustrada uma empresa com filiais espalhadas geograficamente.

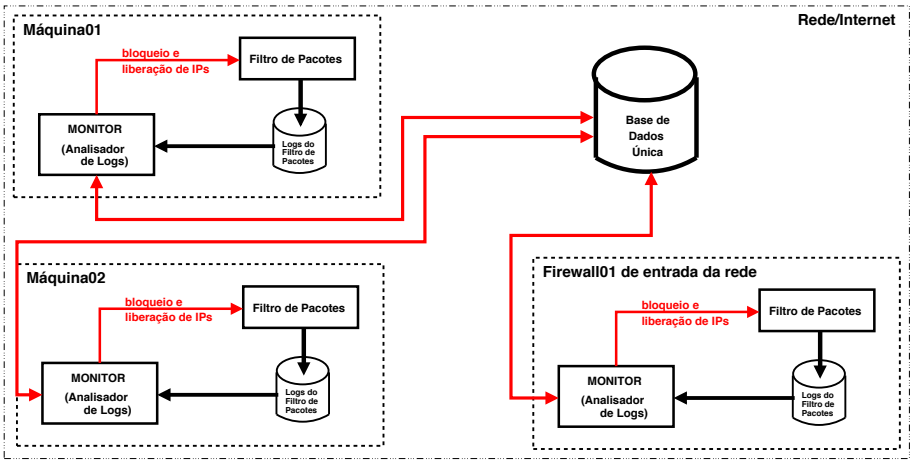

Figure 2. Visão geral do sistema em uma rede

Cada unidade da empresa contém um ponto central de controle de acesso à rede privada. O monitoramento é realizado sobre os *logs* do filtro de pacote dos pontos de acesso. Uma cópia atualizada das tabelas de infratores é mantida na base de dados distribuída, permitindo o compartilhamento de informações entre as filiais.

A detecção de um provável ataque em uma das unidades da empresa irá ser propagada para as demais. Com isso, tentativas de ataques subseqüentes podem ser evitadas. Os sistemas de segurança podem ser facilmente colocados em estado de alerta mesmo antes de qualquer monitoramento suspeito local ter sido identificado. Com isso, um ataque inicial, que visasse a propagação para as demais unidades da rede privada da corporação, seria rapidamente denunciado, tendo suas possibilidades de continuidade ou sucesso bastante reduzidas.

4. Estado Atual

A versão inicial do sistema (segundo a proposta básica - figura 1) foi implementada e testada na rede do Núcleo de Ciência da Computação (NCC) [Kreutz 2004]. Os resultados se mostraram promissores.

A implementação atual funciona apenas com o IPTables, devidamente configurado [Kreutz 2004]. Esta escolha deve-se ao fato deste ser o filtro de pacotes mais difundido e utilizado por administradores de segurança.

A versão completa do sistema, segundo a proposta aqui apresentada, está em

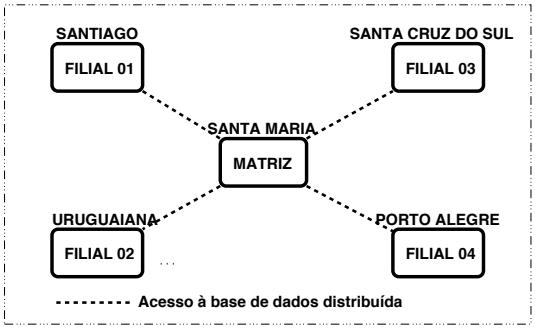

Figure 3. Ilustração de aplicação distribuída do sistema

fase de desenvolvimento. A linguagem de programação Perl está sendo utilizada para a codificação, enquanto que na primeira versão havia sido utilizado *shell scripts* [Kreutz 2004], e o gerenciador de diretórios *online* OpenLDAP para o acesso distribuído e transparente às informações compartilhadas pelas diversas instâncias do sistema. O uso da linguagem Perl tem demonstrado uma melhora significativa no desempenho do sistema, comparado a versão original.

5. Conclusão

A prevenção de intrusões é uma área recente e que ainda demanda pesquisa e soluções. O sistema aqui proposto vem contribuir para o desenvolvimento dessa área.

A arquitetura proposta é simples e prática para o controle preventivo de possíveis invasões. O funcionamento distribuído também torna a ferramenta interessante para uma variedade diversificada de cenários.

Os resultados obtidos até o momento tem se mostrado promissores e satisfatórios. A aplicação e teste na rede do NCC apresentou uma demonstração inicial de sua viabilidade e utilidade. Novos resultados são esperados com os trabalhos futuros.

Trabalhos Futuros. Finalização da implementação da versão completa do sistema, apresentada no decorrer do texto. Inclusão de mecanismos, de identificação de máquinas suspeitas, baseados em freqüência e comportamentos de conexões bem sucedidas (autorizadas). Testes do sistema em um ambiente distribuído. Melhoramentos na arquitetura e implementação. Desenvolvimento de ferramentas para a visualização de estatísticas dos índices do sistema.

References

Caswell, B. and Roesch, M. (2004). Snort - The Open Source Network Intrusion Detection System. `http://www.snort.org/`. Último acesso: maio de 2005.

Desai, N. (2003). Intrusion Prevention Systems: the Next Step in the Evolution of IDS. `http://www.securityfocus.com/infocus/1670`. Último acesso: maio de 2005.

Holland, T. (2004). Understanding IPS and IDS: Using IPS and IDS together for Defense in Depth. `http://www.sans.org/rr/whitepapers/detection/1381.php`. Último acesso: maio de 2005.

Junnonen, T. (2000). Firestarter. `http://www.fs-security.com/`. Último acesso: maio de 2005.

Kreutz, D. (2004). Controle Independente de Restrições de Acesso a Redes Locais em Firewalls que Utilizam o IPTables. In *III Simpósio de Informática da Região Centro do RS*. UNIFRA.

Leach, J. and Tedesco, G. (2004). Firestorm NIDS. `http://www.scaramanga.co.uk/firestorm/`. Último acesso: maio de 2005.

Prelude Trac (2005). Prelude 0.9 Handbook. `https://trac.prelude-ids.org/wiki/PreludeHandbook`. Último acesso realizado em maio de 2005.

Rowland, C. H. (2002). Sentry Tools: portsentry. `http://sourceforge.net/projects/sentrytools/`. Último acesso: maio de 2005.