SuRFE – Sub-Rede de Filtragens Específicas

Ricardo Kléber Martins Galvão, PPGEE, UFRN
Sergio Vianna Fialho, PPGEE, UFRN

***Resumo*—O aumento do número de ataques a redes de corporativas tem sido combatido com o incremento nos recursos aplicados diretamente nos roteadores destas redes. Nesse contexto, os firewalls consolidaram-se como elementos essenciais no processo de controle de entrada e saída de pacotes em uma rede. Estes mecanismos de filtragem têm evoluído conforme evoluem as técnicas de ataques, chegando ao topo da pilha TCP/IP ao incorporar filtragens em nível de aplicação. Esta solução embora eficiente do ponto de vista do nível de filtragem, além de provocar um retardo natural nos pacotes analisados, compromete o desempenho da máquina na filtragem dos demais pacotes pela natural demanda por recursos da máquina para este nível de filtragem. Este artigo apresenta os resultados de um estudo de modelos de tratamento deste problema baseados no reroteamento de pacotes específicos para análise em uma sub-rede de filtragens específicas.**

***Index Terms*—Network Security**

I. Introdução

Acompanhando a evolução histórica dos *firewalls*, observa-se a rápida incorporação de novos mecanismos de filtragem, flexibilização de parâmetros de implementação e modularização de seus componentes, buscando dotar estes elementos periféricos de segurança de um maior grau de controle e bloqueio de ataques aos servidores e às estações por ele protegidos.

A necessidade de filtros específicos para determinados serviços que analisassem não só dados de encaminhamento de pacotes em nível de rede e transporte, mas que identificassem e bloqueassem ataques direcionados à própria aplicação, deram origem aos proxies.

A adição um *proxy* para cada porta relacionada a um serviço específico em execução tornou-se insuficiente, contudo, com o surgimento de aplicativos *peer-to-peer* para troca de arquivos entre máquinas de usuários conectadas à Internet. Esses aplicativos, embora inicialmente padronizados para acesso a partir de portas específicas, e assim poderiam ter seu tráfego bloqueado na filtragem em nível de transporte, passaram a utilizar portas aleatórias, demandando uma solução que investigasse os pacotes em nível de aplicação para identificar o tráfego gerado por este tipo de aplicação.

A incorporação dos proxies à máquina do *firewall*, por si só, representa um aumento natural do retardo no repasse dos pacotes, comprometendo em alguns casos, dependendo do volume de informações analisadas, a disponibilidade da máquina pelo aumento do uso dos recursos da máquina. O risco de comprometimento da máquina em que o *firewall* está em execução aumenta consideravelmente com a incorporação de proxies P2P, tornando-se uma decisão questionável a sua implementação em detrimento das implicações a ela inerentes.

Este artigo apresenta modelos para tratamento de tráfegos específicos baseando-se na utilização de uma sub-rede de filtragem e, assim, aliviando o volume e o nível de informações analisadas pelo *firewall* principal da rede.

II. Firewalls

O *firewall* é uma barreira inteligente entre a rede local da corporação e a Internet, através da qual só passa tráfego autorizado [6].

O motivo principal da instalação de *firewalls* é o controle de acesso em nível de *kernel* [5], realizando a filtragem antes, durante e/ou após o processo de roteamento dos pacotes.

A. Evolução dos Firewalls

1) Primeira Geração – Filtragem de Pacotes

O papel do *firewall* na filtragem de pacotes tradicional era o de assumir as regras de filtragem dos roteadores (*Access Lists* - ACLs), de modo a aliviar o volume de processamento nesses roteadores, isentando-os da responsabilidade pela análise e bloqueio de determinados pacotes.

A utilização de *firewalls* desta geração também se justificava em função das limitações encontradas no uso de ACLs em roteadores: interface de configuração pouco amigável, impossibilidade de registro local de *logs* de acesso/bloqueio, além de questões administrativas envolvendo interesses distintos entre corporações. No cenário mostrado na Figura 1, um roteador serve a duas redes com administradores diferentes e, conseqüentemente, o acesso às regras do roteador implicaria em compartilhamento da sua senha de administração. Caso esse acesso não fosse possível, o administrador em questão não poderia inserir regras de filtragem específicas para sua rede.

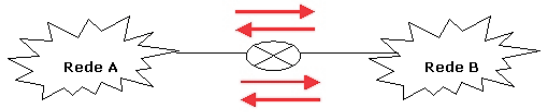

Fig. 1. Conexão de redes com roteador e sem *firewalls*

Uma alternativa a esse cenário seria uso de dois *firewalls* (entre as redes internas e o roteador) sob responsabilidade da administração local de cada uma destas redes. Essa solução além de "desafogar" o processamento do roteador, tornaria mais seguro e controlado o acesso ao equipamento de segurança e permitiria a inserção de regras específicas para cada rede no respectivo *firewall* local (Figura 2).

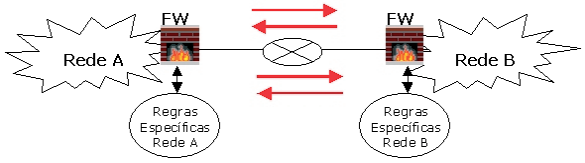

Fig. 2. Conexão de redes com roteador e com *firewalls*

2) Segunda Geração – Incorporação de NAT (*Network Address Translation*)

A segunda geração dos *firewalls* caracterizou-se pela incorporação de uma técnica de conversão de endereços (NAT) à máquina do *firewall*. Na implementação de alguns sistemas operacionais, as tarefas de NAT e filtragem de pacotes, embora na mesma máquina, eram realizados por ferramentas distintas, enquanto em outras implementações, uma mesma ferramenta realizava ambas as tarefas.

A partir de então, o uso de mascaramento (*masquerade*) de endereços IP privados para acesso à rede externa, utilizando temporariamente um único endereço externo (NAT N:1) passou a ser uma nova funcionalidade dos *firewalls*.

A conversão direta e fixa de endereços públicos em privados (NAT 1:1), em que determinadas máquinas da rede interna (geralmente servidores de aplicação) poderiam ser acessadas a partir da rede externa através de seu endereço público (mapeado para seu endereço privado) também fazia parte desta solução, e se encontrava disponibilizada a partir de então.

3) Terceira Geração – Checagem de Estados

Dentre as soluções existentes até então, a dificuldade dos *firewalls* era diferenciar os pacotes que entravam na rede como resposta a solicitações internas, dos pacotes que, partindo da rede externa, buscavam iniciar conexões em máquinas da Intranet.

A inspeção do estado dos pacotes (*stateful inspection*) marcou uma nova era para os *firewalls*. Sua terceira geração, com a possibilidade de restringir o acesso de pacotes vindos da rede externa, liberando aqueles relacionados a conexões estabelecidas a partir de máquinas internas e bloqueando os demais. Dessa forma, tornou-se possível evitar vários tipos de ataques conhecidos até então, aumentando consideravelmente a segurança da rede corporativa.

4) Quarta Geração – Filtragens Específicas em Nível de Aplicação

Antes do surgimento desta modalidade de filtragem, uma das maiores limitações para os *firewalls* na detecção e bloqueio de ataques contra redes corporativas eram os ataques contra as implementações de serviços liberados pelo *firewall*, ou seja, a exploração de vulnerabilidades nas aplicações em execução acessadas a partir de portas válidas (serviços tradicionais), utilizadas para prover acesso a partir de máquinas externas a informações da instituição.

Nestes casos específicos, informações como endereços IP, portas, protocolos e estados de conexão não eram suficientes para identificar e eventualmente bloquear a exploração das vulnerabilidades dos programas.

O "mito" de que os dados da camada de aplicação só deveriam ser manipulados pelos equipamentos das extremidades da conexão (cliente e servidor) caiu por terra, diante da necessidade de filtragem das informações transportadas nesta camada, de modo a identificar ataques em andamento contra a corporação.

A quarta geração de *firewalls* é marcada, portanto, por implementações que disponibilizam parâmetros para configuração de filtragem neste nível específico.

Proxies de Aplicação

A utilização dos proxies de aplicação consiste em dotar a rede de um elemento intermediário entre os usuários e os servidores de determinada(s) aplicação(ões). Este elemento recebe a solicitação de conexão a uma máquina externa e, ao invés de repassar o pacote, assume a condição de cliente iniciando uma nova conexão ao destino e repassando os pacotes de retorno ao cliente original.

A utilização deste tipo de serviço, além de proteger os endereços reais das máquinas internas (clientes), permite a filtragem e eventual necessidade de bloqueio de pacotes baseando-se em informações de seu cabeçalho IP, ou mesmo no conteúdo dos pacotes.

A grande desvantagem na adoção deste modelo é a necessidade de utilização de proxies específicos para cada serviço (http, smtp, ftp, etc.).

Firewalls de Aplicações

Os *firewalls* de aplicações têm funcionalidades semelhantes aos proxies, já que analisam e eventualmente filtram/bloqueiam conexões para determinados serviços. Porém, este elemento de segurança não intermedia as conexões, apenas aplicando regras de filtragem baseadas no conteúdo dos pacotes.

Comparando-o com os *firewalls* tradicionais, os *firewalls* de aplicações diferem no objeto da análise, incorporando o nível de filtragem de cabeçalho (endereçamento e portas de origem e destino), adicionadas da análise do nível da camada de aplicação (conteúdo dos pacotes).

Análise de Performance

A filtragem de pacotes tradicional, em termos de velocidade de repasse de pacotes, é entre 3 e 10 vezes mais veloz que a utilização de proxies de aplicação [1]. Este retardo é decorrente da filtragem de pacotes no nível de aplicação, característica dos proxies não presente nos *firewalls* tradicionais.

Na filtragem de aplicações, este retardo pode ser ainda maior, já que a análise e comparação dos dados dos pacotes em trânsito com padrões pré-estabelecidos (assinaturas) representarão, neste caso, um maior volume de utilização de recursos de processamento e memória da máquina.

O uso da filtragem em nível de aplicação na mesma máquina em que é realizada a filtragem de pacotes é desaconselhada em função do possível comprometimento de todo o processo de filtragem em decorrência do nível de análise dos pacotes resultando em degradação da performance [3] do *hardware* do *firewall*.

A análise dos dados (camada de aplicação) dos pacotes, portanto, tende a tornar-se inviável com o aumento do volume de informações que passam pelo filtro de aplicações se esta filtragem é feita na mesma máquina em que é realizada a filtragem de pacotes.

III. Sub-Rede de Filtragens Específicas (SuRFE)

A solução para esta situação é manter no *firewall* principal somente a filtragem de pacotes, desviando os pacotes endereçados a máquinas e/ou serviços internos específicos (análise baseada nas informações de endereço IP e porta de destino) para uma sub-rede de filtragem de aplicações. Nesta sub-rede, então, será realizada a filtragem na específica, bloqueando pacotes notadamente maliciosos, isto é, pacotes cujo conteúdo coincida com *strings* listadas na base de assinaturas de ataques carregadas pelo(s) *firewall(s)* de aplicações, conforme ilustrado nas Figuras 3 e 4.

A arquitetura desta sub-rede pode variar, conforme necessidades e/ou disponibilidades de recursos específicos. A seguir são apresentados os modelos básicos destas arquiteturas. A pesquisa em andamento consiste em implementar e determinar os impactos de utilização de cada um destes modelos, analisando a performance de roteamento e eficiência dos mecanismos de filtragem.

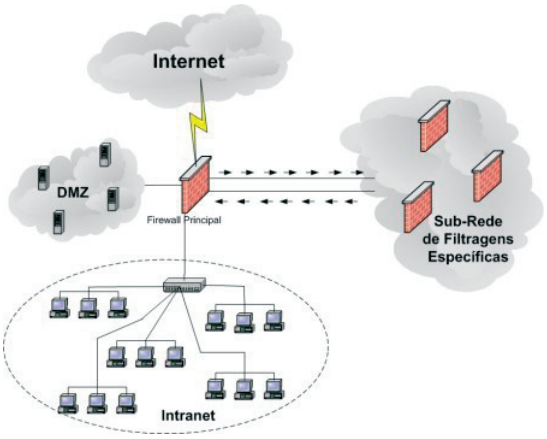

Fig. 3. Sub-Rede de Filtragens Específicas

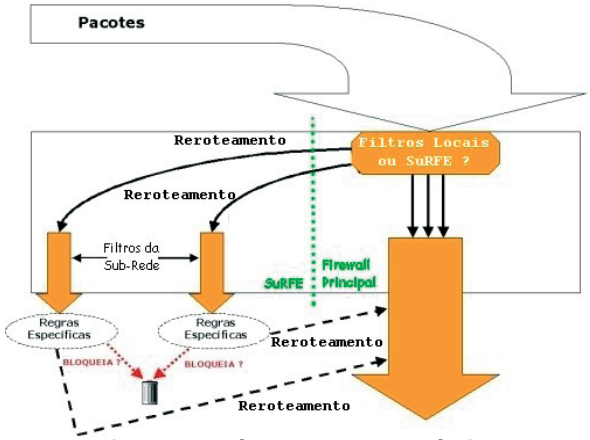

Fig. 4. Esquema de Funcionamento da SuRFE

A. Modelos Propostos

1) SuRFE com uma única máquina

Este é o modelo (Figura 5) de mais fácil implementação e de menor custo, já que envolve somente uma máquina adicional à estrutura pré-existente, e duas placas de rede no *firewall* para o desvio dos pacotes que serão analisados e retorno dos pacotes que não foram bloqueados pelas regras de filtragem. Entretanto, a utilização do re-roteamento deve ser um recurso suportado e implementado no firewall principal, já que somente os pacotes que se deseja analisar serão re-roteados para o filtro de aplicações (desvio baseado no serviço e/ou rede de origem/destino), sem modificação do cabeçalho, enquanto os demais pacotes serão filtrados e/ou repassados para seus destinos sem o re-roteamento.

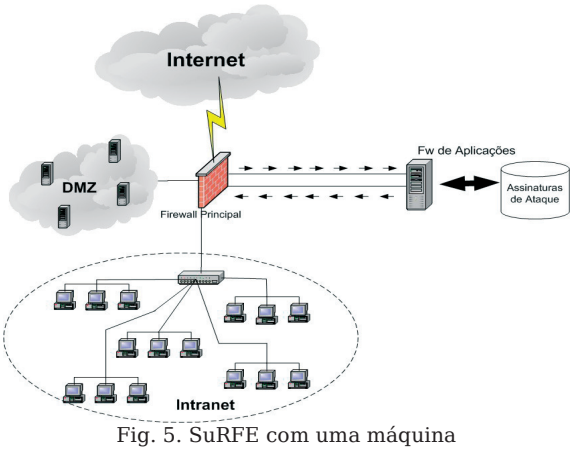

Fig. 5. SuRFE com uma máquina

2) SuRFE com Balanceamento de Carga

Neste modelo (Figura 6) será realizado o balanceamento de carga entre os firewalls da SuRFE (máquinas com o mesmo perfil) oferecendo redundância (alta disponibilidade) e escalabilidade para a solução.

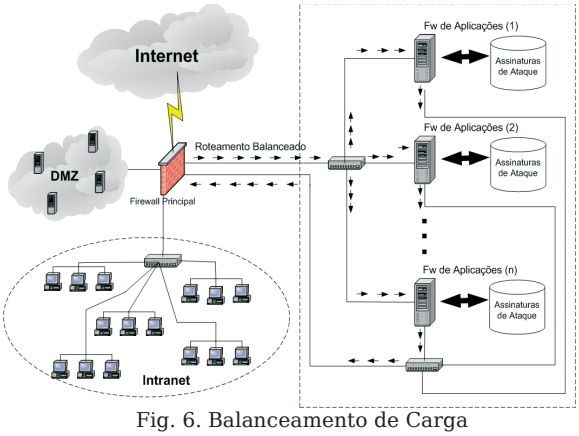

Fig. 6. Balanceamento de Carga

3) SuRFE com Separação por Aplicação

Neste modelo (Figura 7) a sub-rede de filtragem de aplicação é formada por *firewalls* com bases de assinaturas específicas para cada aplicação (porta ou conjunto de portas). O *firewall* principal redireciona os pacotes ao *firewall* de aplicação específico, de acordo com a aplicação destino de cada um deles.

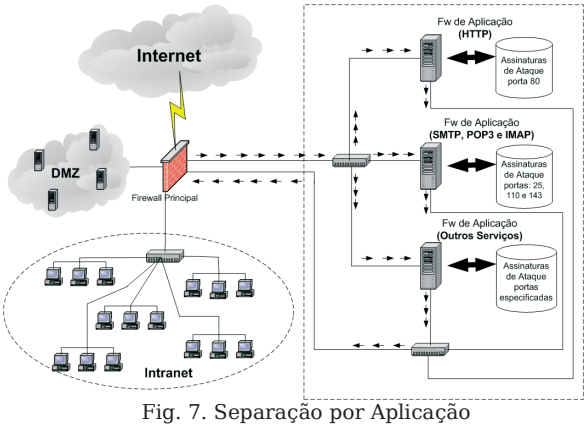

Fig. 7. Separação por Aplicação

IV. Conclusão

A insegurança das redes de computadores diante dos novos tipos de ataque que surgem a todo momento demanda soluções nem sempre satisfatórias do ponto de vista da usabilidade. Tornar um sistema minimamente seguro depende de decisões que podem resultar em problemas de indisponibilidade e até inviabilidade de determinadas aplicações cujo desempenho varia de acordo com o tempo de resposta.

O estudo de soluções de implementação simples que minimizem os impactos dos mecanismos de filtragem é uma necessidade tão crítica quanto as próprias soluções.

O objetivo final do estudo parcialmente detalhado neste artigo busca ratificar a viabilidade dos novos elementos de filtragem, minimizando o impacto de sua implementação pelo o tratamento específico de cada tipo de tráfego com a seleção adequada dos filtros a que serão submetidos.

References

[1] CHUVAKIN, Anton, *IPTables Linux firewall with packet string-matching support*: SecurityFocus , 2001.
[2] HUMES, Jeff. *Filtering packets based on string matching:* LinuxGuru.net, 2001.
[3] SILVA, Artur e PEIXOTO, Jarbas. *Iptables: Uma solução de baixo custo para implementação de firewalls p.102*: São Paulo.GTS, 2003.
[4] GONÇALVES, M. Firewalls – Guia Completo. Rio de Janeiro: Ed. Ciência Moderna, 2000.
[5] HATCH, B., LEE, J., KURTZ, G. Hackers Expostos – Linux. São Paulo: Makron Books, 2002.
[6] MARCIO, A Internet e os Hackers – Ataques e Defesas.São Paulo: Chantal, 2000.