Implantação de um *Firewall* em OpenBSD com PF

Tiago Gerke[1]

[1]Departamento de Computação – Universidade Estadual do Centro-Oeste (UNICENTRO)
Guarapuava – PR – Brasil
tiagogerke@yahoo.com.br

Abstract: *Receive notifications of unhappy users, customers constantly complaining of the unavailability of services or to suffer from malicious users who seek access to confidential information. This is a common scenario in the life of network administrators who did not design their firewall as they should. Therefore, this study will present a firewall solution with software open source that was deployed in the Almix Internet provider in Guarapuava, with the goal of providing more security to the provider.*

Resumo. *Receber notificações de usuários insatisfeitos, clientes reclamando incessantemente da indisponibilidade de serviços ou até sofrer com usuários mal intencionados que buscam obter acesso a informações confidenciais, é um cenário comum na vida dos administradores de rede que não projetaram seus firewalls como deveriam. Portanto, neste trabalho, será apresentada uma solução de firewall em software livre que foi implantada no provedor de Internet Almix na cidade Guarapuava, com o objetivo de fornecer mais segurança ao provedor.*

1. Introdução

Com o crescimento da Internet, da quantidade de serviços *on-line* disponíveis e usuários acessando esses serviços, cresceu também a necessidade de maior segurança. Isto se deve ao fato de muitos usuários mal intencionados tentarem interromper os serviços e obter acesso a informações sigilosas ou apenas causar prejuízos [Tanenbaum 2003].

Segundo [Castro 2007], a Estônia, um dos países mais informatizados do mundo e um dos pioneiros da tecnologia do "governo eletrônico", sofreu ataques de DoS (*Denial Of Service* – negação de serviços) em maio de 2007. Ataques estes sem precedentes na história, com cerca de um milhão de computadores participando da ofensiva, perdurando por cerca de três semanas. Os alvos foram sites governamentais e grandes companhias como bancos. Ainda, afirma-se que estes ataques virtuais vêm ocorrendo desde 1990, porém com maior frequência depois do atentado terrorista aos Estados Unidos em 11 de setembro de 2001. Sendo assim, percebe-se a necessidade de mecanismos de segurança cada vez melhores para amenizar ou até evitar as conseqüências desses ataques [Tanenbaum 2003].

Portanto, esse artigo tem por principal objetivo apresentar a implantação de um servidor de *firewall* em OpenBSD com PF (*Packet Filter* – filtro de pacotes do OpenBSD). Esse servidor foi implantado na empresa Almix Internet[17] no segundo semestre de 2009. A empresa provê acesso à Internet via rádio para cidade de Guarapuava, possuindo vários servidores e serviços que necessitam de segurança adicional.

[17] www.almix.com.br

Este documento está estruturado da seguinte maneira: descrição do problema, seguido da metodologia e proposta de trabalho. Na sequência encontram-se detalhes da implantação do servidor e por fim as conclusões e trabalhos futuros sugeridos.

2. Problema

O provedor de Internet Almix caracteriza-se principalmente por fornecer Internet via rádio para cidade de Guarapuava. Tem grande aceitação por parte dos clientes, pois consegue levar acesso à Internet onde as operadoras de telefonia fixa não conseguem.

O provedor possui uma rede relativamente grande, com oito torres de transmissão em pontos específicos da cidade, visando a melhor qualidade de sinal para seus clientes. Possui também cerca de dez servidores próprios, que provêem alguns serviços como e-mail, DNS externo, hospedagem, entre outros, além da Internet sem fio.

A rede de empresa foi várias vezes alvo de ataques e até então não possuía um mecanismo de segurança que fornecesse proteção eficaz para seus servidores. Isso por que cada servidor possuía seu próprio *firewall* integrado, dificultando a manutenção e eficiência dos servidores.

A Figura 1 mostra a rede da Almix antes da implantação do servidor de *firewall* OpenBSD, cabe ressaltar que a figura não mostra todos os detalhes da rede do provedor. Pode-se perceber que cada servidor (FTP, E-mail, etc.) está ligado diretamente em um *switch*, que por sua vez está conectado à Internet e a rede dos clientes.

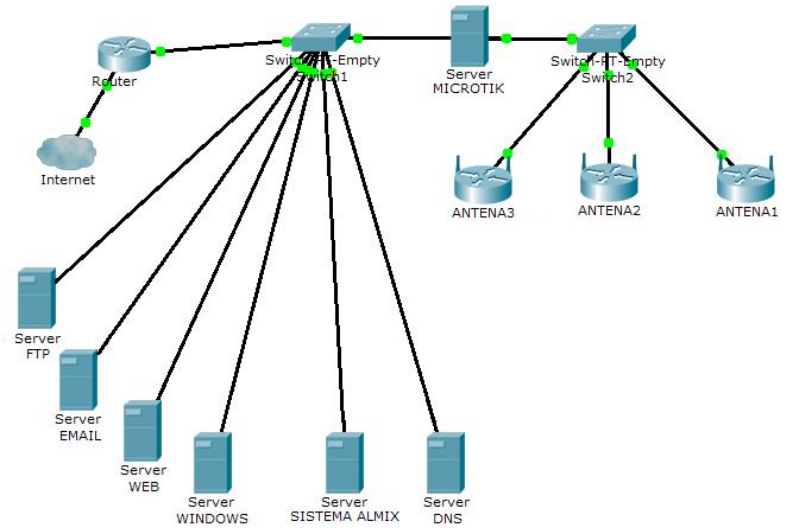

Figura 2. Rede da Almix Antes do *Firewall* OpenBSD.

Uma grande desvantagem era que cada servidor possuía seu próprio *firewall*, sendo assim havia vários pontos de contato tanto com a Internet quanto com a rede dos clientes, resultando em vários pontos vulneráveis a ataques. Outro problema era quanto à manutenção dos servidores, pois qualquer mudança em seu *firewall* acarretaria mudanças em todos os outros, ou seja, cada *firewall* precisaria ser tratado em separado. Os *firewalls* antes usados nos servidores eram construídos em *iptables* e os servidores trabalhavam com o sistema operacional Debian.

Sendo assim, constatou-se que não havia um tratamento de segurança adequado, que fornecesse o menor número possível de pontos de contato com a Internet e também a maior eficiência dos servidores.

Alguns trabalhos correlatos podem ser encontrados em [Helleis 2008] e [Luz 2009]. Onde o primeiro trata da implementação de um *firewall* em OpenBSD em uma rede de grande porte, na CELEPAR (Companhia de Tecnologia da Informação e Comunicação do Paraná). E o segundo trabalho apresenta os principais conceitos do OpenBSD e do PF e também estudo de caso real da implementação de um *firewall* utilizando esse sistema.

3. Metodologia

Segundo [Infowester 2009] o *firewall* pode ser considerado como uma barreira, que protege e controla os dados que trafegam entre uma rede privada e a Internet, entre redes privadas ou somente entre dois computadores. Ainda, pode-se evitar que uma rede seja acessada sem autorização, bem como evitar que informações confidenciais sejam acessadas por pessoas não autorizadas.

Pode-se evitar também vírus e cavalos de tróia que por ventura tentem prejudicar a rede, bloqueando portas que normalmente são usadas por esses intrusos. Ainda, em redes coorporativas pode-se evitar que usuários de um determinado setor acessem serviços indevidos, obtendo-se assim um controle interno maior [Infowester 2009].

Para a implantação do *firewall* proposto, foi utilizado o sistema operacional OpenBSD, que segundo [Koch 2009] é um sistema *Unix-Like* focado em segurança, muito utilizado nos dias de hoje para a construção de *firewalls*. BSD é sigla para *Berkeley Software Distribution*, um sistema que derivou diretamente do código do Unix original, desenvolvido em Berkeley na Universidade da Califórnia nos anos 70. Ele foi o primeiro sistema operacional a implementar o protocolo TCP/IP e a realizar testes com a Internet [Koch 2009].

Ainda, do BSD surgiram duas variações: O FreeBSD, que foi destinado a diversos usos, tais como computadores pessoais e servidores e o NetBSD focado em portabilidade. Koch ainda afirma que existem torradeiras com capacidade para rodar NetBSD, mostrando assim como o sistema é portável.

Do NetBSD surgiu o OpenBSD, e seu projeto também é o responsável por criar a biblioteca de encriptação OpenSSL (*Security Sockets Layer*), capaz de criar túneis encriptados de transmissão de dados por TCP/IP como o *https* e o *ssh*, muito utilizados no Linux e Windows [Koch 2009]. Sabe-se ainda que o OpenBSD visa prover aos seus usuários portabilidade, padronização, correção, segurança e criptografia, e está atualmente na versão 5.1. [OpenBSD 2012]. Mais sobre esse sistema pode ser encontrado em [OpenBSD 2012].

Em conjunto com o OpenBSD, foi utilizado o PF (*Packet Filter*), que é o sistema usado pelo OpenBSD para filtrar tráfego TCP/IP e fazer tradução do endereço de rede (NAT – *Network Address Translation*). O PF também é capaz de fazer normalização e condicionamento do tráfego TCP/IP e providenciar o controle de banda e priorização de pacotes. O subsistema de filtro de pacotes do OpenBSD [Hansteen 2009] foi escrito originalmente em um esforço de desenvolvimento extremamente rápido. Foi desenvolvido por Daniel Hartmeier e um número de desenvolvedores do OpenBSD, e foi lançada como parte de um padrão do OpenBSD 3.0, em dezembro de

2001. Mais informações sobre esse filtro de pacotes encontram-se em [OpenBSD2 2012].

Com esse sistema é possível fazer a filtragem de pacotes, que segundo [OpenBSD3 2009] é o bloqueio ou a liberação de maneira seletiva da passagem de pacotes de dados, conforme esses pacotes trafegam pela interface de rede. Os critérios que o PF usa quando inspeciona pacotes são baseados na Camada de rede (IPv4 e IPv6) e Camada de transporte (TCP, UDP, ICMP e ICMPv6) do modelo OSI. Os critérios mais usados são os endereços de origem e destino, porta de origem e destino e protocolo.

Ainda, para essa filtragem são utilizadas as regras, que especificam critérios que os pacotes devem enquadrar-se e ações que devem ser tomadas caso os pacotes se enquadrem em determinadas regras. A sintaxe das regras, bem como sua criação podem ser encontradas em [OpenBSD3 2009].

3.1 Proposta para o Resolução do Problema

Tendo em vista os problemas constatados e descritos na seção 2, foi proposta a criação de dois *firewalls* redundantes, como é mostrado pelos servidores destacados em vermelho na figura a seguir, sendo que a figura não mostra todos os detalhes da rede do provedor.

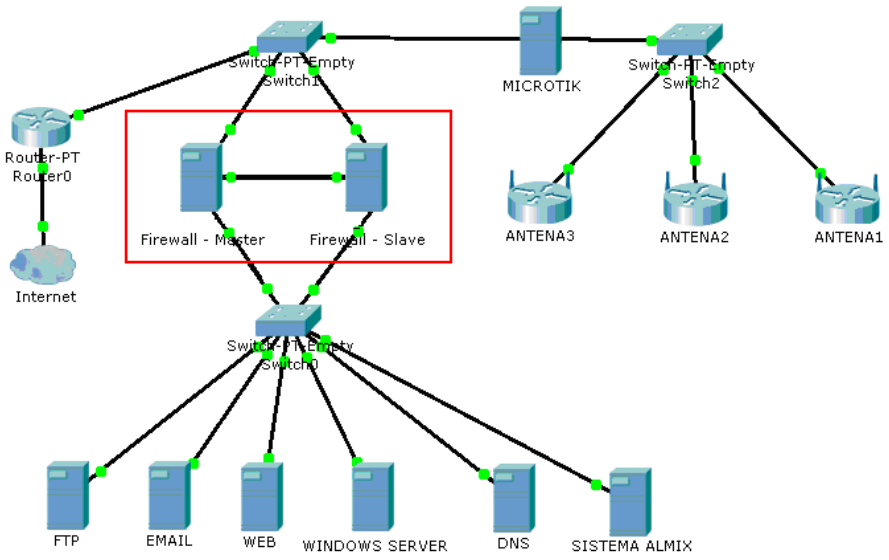

Figura 2. Solução com *firewalls* redundantes.

Essa solução visa reduzir para somente dois o número de pontos de contato dos servidores com a Internet, porém somente um dos pontos de contato é ativo, tendo em vista que somente o servidor de *firewall master* (primário) atenderia o tráfego, e o servidor *slave* (secundário) ficaria desativado, e só seria usado em caso de falhas no primário.

Essa solução, se implantada, seria ideal devido à redundância nos servidores de *firewall*, pois é uma garantia a mais de que os servidores protegidos não ficarão indisponíveis em caso de falha em um dos *firewalls.*

Porém, por falta de interesse da parte dos proprietários do provedor Almix em investir em dois novos servidores, não foi possível executar a proposta inicial, sendo

esta então adaptada. A nova proposta restringiu-se à criação de um único servidor de *firewall* para a proteção dos servidores, como mostra a figura 3.

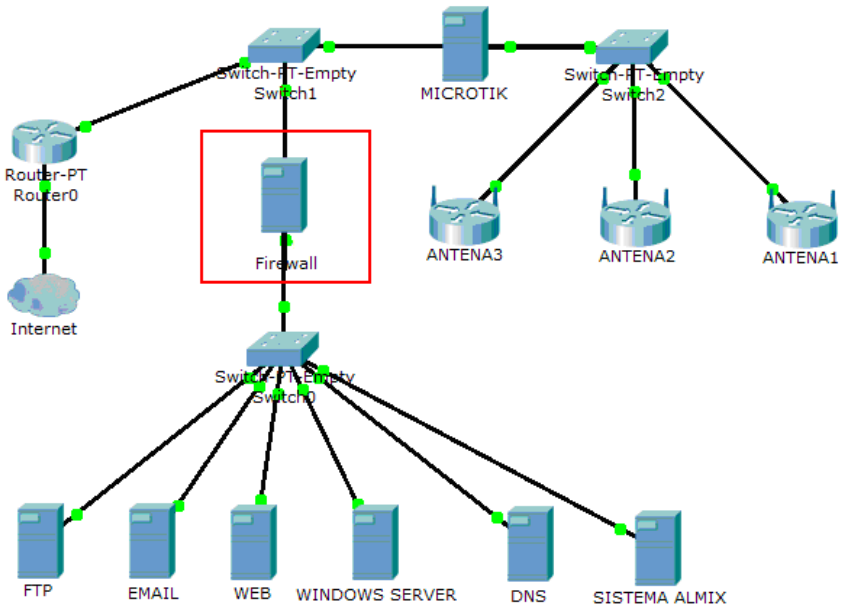

Figura 3. Solução sem redundância.

Essa solução, ainda assim, resolve os problemas descritos na seção 2, pois reduz os pontos de contato dos servidores com a internet e centraliza o *firewall* em somente um servidor. Porém manter somente um servidor de *firewall* na rede pode gerar alguns riscos, pois, caso esse servidor falhe, como não há mecanismo de segurança em caso de falhas, todos os servidores que estão sendo protegidos pelo *firewall* ficarão fora de serviço. Esse risco seria amenizado caso a solução com *firewalls* redundantes fosse implantada.

3.2 Implantação do *Firewall*

Além de todos os pontos referentes ao OpenBSD que foram descritos na seção 3, e também por já possuir um sistema de filtragem de pacotes nativo (PF), específicos para *firewalls*, outro ponto foi também favorável a escolha desse sistema operacional, que é a capacidade trabalhar de modo transparente.

O fato de o servidor trabalhar de modo transparente significa que ele é tratado logicamente pela rede como sendo uma *bridge*, porém fisicamente ele continua sendo um servidor normal. Segundo [OpenBSD4 2009], ao contrário de um roteador, os pacotes são transferidos pela *bridge* de maneira "invisível". Os dois segmentos de rede parecem ser um segmento só, para os nós, em qualquer um dos lados da *bridge*. Sendo assim, além do *firewall* trabalhar de modo transparente há o filtro de pacotes que fará a filtragem de tráfego TCP/IP.

Ainda, para configuração das regras do PF que foram utilizadas no servidor, foi usado o programa *FWBuilder*[18], que é um programa que ajuda na construção das regras de forma gráfica, gerando então o arquivo com as regras em forma de texto para uso no servidor. Optou-se por utilizar esse programa devido à facilidade na criação das

[18] http://www.fwbuilder.org

regras de forma gráfica, e também por haver a possibilidade do programa identificar regras redundantes, ou seja, regras diferentes que possuem o mesmo significado.

Cada servidor que iria ser protegido pelo *firewall* em OpenBSD era tratado em separado para que fossem identificados todos os serviços necessários para o correto funcionamento do servidor. Depois de estudado e os serviços identificados, as regras para tal servidor eram criadas e configuradas no *firewall.*

Colocadas as regras, o servidor era movido fisicamente para proteção no *firewall* e permanecia por cerca de duas semanas para testes, tendo em vista a necessidade de acompanhar o comportamento do servidor, agora protegido pelo novo *firewall.* Alguns testes (portas, acesso, entre outros) eram feitos para verificar se o servidor funcionava corretamente, se as portas necessárias para seu funcionamento estavam corretamente liberadas e também se as portas desnecessárias estavam corretamente bloqueadas.

Verificado o correto funcionamento do servidor, o mesmo processo era seguido para cada um dos servidores que ainda necessitavam de proteção. Os servidores que ficaram sob proteção do *firewall* são: FTP, E-Mail, *Windows Server*, MySQL, HTTP e HTTPS, POP, SMTP e SMTPS, DNS primário e DNS secundário.

Foi utilizada também uma ferramenta chamada *PFStat,* que é um utilitário do OpenBSD usado para coletar informações sobre o PF, como pacotes bloqueados, pacotes liberados, entre outras. Com essas informações o *PFStat* é capaz de gerar gráficos estatísticos (Figura 4) que depois podem ser visualizados pelo *browser* para posteriores análises e auditorias.

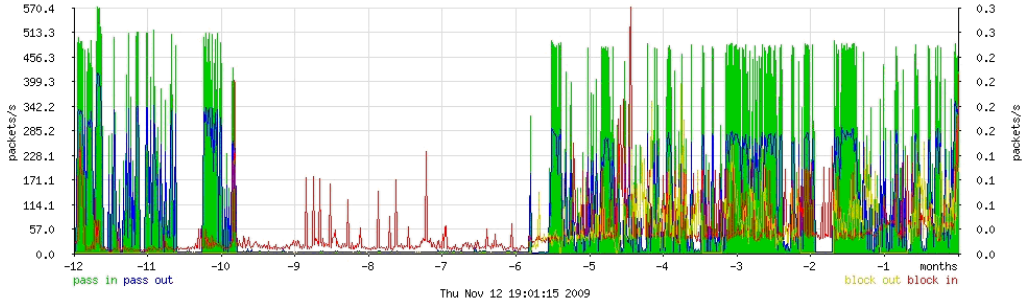

Figura 4. Exemplo de gráfico gerado pelo *PFStat.*

O *PFStat* foi usado para fazer o monitoramento do *firewall* da Almix e com ele foi possível verificar algumas informações que são importantes no funcionamento do servidor, como o tráfego, quantidade de pacotes bloqueados, liberados, fila de pacotes, entre outros.

4. Conclusões

Com a implementação do servidor de *firewall* em OpenBSD no provedor de Internet pode-se detectar que a rede Almix recebe tentativas de acessos indevidos. Isso foi possível observar, pois através do monitoramento do *firewall* e da análise dos *logs* percebeu-se várias vezes que usuários indevidos foram bloqueados ao tentarem obter acesso aos servidores protegidos.

Notou-se também que o sistema operacional OpenBSD é muito estável e com isso seguro, pois em todo o tempo de implantação (cerca de seis meses) só deixou de funcionar uma vez por problemas físicos na placa de rede do computador, que fizeram com que o sistema travasse. Também percebeu-se que o OpenBSD e suas ferramentas,

disponíveis gratuitamente sob a licença BSD, possibilitam uma solução robusta e acessível para roteamento e *firewall* redundante.

Ainda, constatou-se que é de grande vantagem a possibilidade do OpenBSD trabalhar de modo transparente, pois em testes externos não foi possível identificar o servidor de *firewall* na rede Almix, dificultando assim o trabalho de possíveis invasores.

Apesar do *firewall* ter sido criado em um computador *Desktop*, não ideal para esse tipo de aplicação, percebeu-se que ele atendeu ao tráfego da rede do provedor, tendo em vista que através do monitoramento constatou-se que o tráfego gira em torno de 6 a 10 *Mega Bytes* por segundo. Sendo assim, conclui-se também que o tráfego que passa no servidor pode crescer em torno de 50%. Acima disso não é possível garantir, pois o servidor nunca foi submetido a tais condições e não se tem informações de como seria o seu comportamento.

O FWBuilder mostrou-se fundamental na construção das regras para o filtro de pacotes, pois através da interface gráfica obteve-se um melhor entendimento do funcionamento de cada regra, bem como da sua ligação com as demais.

O PF se mostrou uma boa solução como *firewall*, tendo ótimas ferramentas para controle e redundância. Essa alternativa de *firewall* pode substituir boa parte dos *firewalls* que hoje em dia são proprietários, por exemplo: *ASA* e *PIX* da Cisco, *Checkpoint* e *Junniper*.

Com base na pesquisa elaborada, é interessante destacar que ainda existe trabalho a ser realizado. Como mostrado na seção 3.1, a proposta inicial era a de dois *firewalls* redundantes, porém não foi possível devido à falta de investimento por parte da Almix. Sendo assim, com o investimento de mais um servidor, pode ser feito a redundância de *firewalls*, o que seria interessante por que limitaria muito mais os riscos de paradas na rede devidos a falhas do *firewall*.

Uma análise comparativa entre *firewalls* é uma proposta que pode ser desenvolvida, abrangendo os *firewalls* PF, IPTables e Checkpoint, afim de verificar as vantagens, desvantagens e também como cada um se comporta em ambientes diferentes.

Outra pesquisa que poderia ser feita é em relação ao comportamento do OpenBSD em diferentes cenários, avaliando o sistema em diferentes quantidades e tipos de tráfego, e também trabalhando com controle de banda e priorização de pacotes.

5. Referências

Castro e Luiz F. D. Moura (2007) "Estônia Sofre Ataque Virtual", http://www.pucminas.br/imagedb/conjuntura/CNO_ARQ_NOTIC20070704113456.pdf?PHPSESSID=40d7b2656db775ea31268bf3df1c04cc, Setembro.

Hansteen, Peter (2009) "*Firewalling with OpenBSD's PF packet filter*", http://www.bgnett.no/~peter/pf/en/intro.html, Setembro.

Helleis, H. Lorenz (2008) "Apresentação de estudo de caso de uma rede de grande porte usando PF com OpenBSD para segurança de perímetro de rede com alta disponibilidade", http://www.ginux.ufla.br/files/mono-LorenzHelleis.pdf, Setembro.

Infowester (2004) "*Firewall*- Conceitos e Tipos", http://www.infowester.com/firewall.php, Setembro.

Koch, Daniel (2009) "Como funcionam os Projetos *Open-Source*", http://informatica.hsw.uol.com.br/projetos-open-source2.htm, Setembro.

Luz, L. Jeferson (2009) "*Firewall* OpenBSD – Conceitos e Estudo de Caso", http://www.polaticus.com.br/downloads/TCC%20Firewall%20OpenBSD.pdf, Setembro.

OpenBSD (2012) "*OpenBSD 5.1, Open, Functional & Secure*", http://www.openbsd.org/, Setembro.

OpenBSD2 (2012), " PF: Começando", http://www.openbsd.org/faq/pf/pt/config.html, Setembro.

OpenBSD3 (2009), "PF: Filtragem de Pacotes", http://www.openbsd.org/faq/pf/pt/filter.html, Setembro.

OpenBSD4 (2009), "Configurando uma Bridge no OpenBSD", http://www.openbsd-br.org/?q=node/14, Setembro.

Tanenbaum, Andrew (2003), *Computer Networks*, Pretence-Hall, 4ª edição.