Faculdade de Tecnologia de Americana
Curso de Segurança da Informação

Arquiteturas de *Firewall*: Propostas para ambientes empresariais

Maurício Rafael Possari

Americana, SP
2011

Faculdade de Tecnologia de Americana
Curso de Segurança da Informação

Arquiteturas de Firewall: Propostas para ambientes empresariais

Maurício Rafael Possari

mauricio.possari@gmail.com

Trabalho de conclusão de curso – desenvolvido em cumprimento à exigência curricular do Curso de Segurança da Informação - ASTI, sob orientação do Prof. Dr. José Luís Zem.

Área: Segurança da Informação

Americana, SP
2011

FICHA CATALOGRÁFICA elaborada pela
BIBLIOTECA – FATEC Americana – CEETPS

Possari, Maurício Rafael
P889a Arquiteturas de firewall: propostas para ambientes empresariais. / Maurício Rafael Possari. – Americana: 2011.
63f.

Monografia (Graduação em Análise de Sistemas e Tecnologia da Informação). - - Faculdade de Tecnologia de Americana – Centro Estadual de Educação Tecnológica Paula Souza.
Orientador: Prof. Dr. José Luís Zem

1. Segurança em sistemas de informação I. Zem, José Luis II. Centro Estadual de Educação Tecnológica Paula Souza – Faculdade de Tecnologia de Americana.

CDU: 681.518.5

Bibliotecária responsável Ana Valquiria Niaradi – CRB-8 região 6203

BANCA EXAMINADORA

Prof. Dr. José Luís Zem (orientador)

Prof. Marcus Vinicius Lahr Giraldi (convidado)

Prof. Antonio Alfredo Lacerda (presidente)

AGRADECIMENTOS

Em primeiro lugar a Deus que sempre ajuda aqueles que se esforçam, mesmo que não sejam muito talentosos. Em segundo lugar ao meu orientador que mostrou o norte quando eu estava à deriva no mar de livros e artigos nos quais procurava um tema para esse trabalho e que teve paciência para me atender aos sábados, sempre com bom humor, que, aliás, é marca registrada de suas aulas. Gostaria de agradecer também aos meus colegas de trabalho da Unicamp, que me ajudaram muito com a experiência, conhecimento e boa vontade de ensinar e responder a todas minhas dúvidas técnicas, além dos excelentes cursos que pude realizar naquela instituição, com instrutores bastante preparados e atenciosos. À minha noiva que sempre me apóia e que ajudou a corrigir este trabalho. E finalmente a todos os professores e funcionários da FATEC, que mesmo com as limitações orçamentárias, sempre fizeram o melhor possível para que nós, alunos, tivéssemos um ensino de qualidade.

DEDICATÓRIA

Aos professores da FATEC de Americana.

RESUMO

O *firewall* é um instrumento de segurança de rede bastante eficiente e que apresenta uma boa relação custo-benefício, além de ser possível encontrar no mercado soluções bastante confiáveis e, ainda, com *softwares* livres.

Apesar disso, nem sempre ele é usado da maneira correta, pois, geralmente, é configurado apenas para separar a Internet da rede interna da empresa, subutilizando, assim, essa ferramenta.

Este estudo busca demonstrar como uma Arquitetura de *Firewall* eficiente pode evitar problemas de segurança e limitar o campo de ação de um eventual invasor.

Palavras Chave: *firewall* segurança perímetro

ABSTRACT

The firewall is a tool for network security that provides an efficient and cost-benefit relationship and be able to find solutions on the market and very reliable, even with free software.

Nevertheless it's not always used the right way, because usually it's configured just to separate Internet from the company's internal network, underdoing this powerful tool.

This paper seeks to demonstrate how an efficient Firewall Architecture can avoid security problems and limit the purview of an attacker.

Keywords: firewall security perimeter

LISTA DE FIGURAS

LISTA DE SIGLAS

AC: Autoridade Certificadora

DMZ*: De-Militarized Zone* ou Zona Desmilitarizada

FTP: *File Transfer Protocol*

HD*: Hard Disk*

HIDS: *Host-based Intrusion Detection System*

HTTP: *Hypertext Transfer Protocol*

HTTPS: *HyperText Transfer Protocol Secure*

ICP: Infra-estrutura de Chaves Públicas

IDS: *Intrusion Detection* System ou Sistema de Detecção de Intrusões

IP: *Internet Protocol*

IPS: Intrusion Prevention Systems ou Sistema de Prevenção de Intrusões

LAN: *Local Area Network*

NIDS: *Network-based Intrusion Detection System*

PIN: *Personal Identification Number*

SSO: *Single Sign-On*

TI*: *Tecnologia da Informação

VPN: *Virtual Private Network*

WEB: *World Wide Web*

SUMÁRIO

1 INTRODUÇÃO

O *firewall* é um instrumento de segurança de rede bastante eficiente e que apresenta uma boa relação custo-benefício, além de ser possível encontrar no mercado soluções bastante confiáveis e, ainda, com *softwares* livres.

Apesar disso, nem sempre ele é usado da maneira correta, pois, geralmente, é configurado apenas para separar a Internet da rede interna da empresa, subutilizando, assim, essa ferramenta.

Este estudo busca demonstrar como uma Arquitetura de *Firewall* eficiente pode evitar problemas de segurança e limitar o campo de ação de um eventual invasor.

O **objetivo geral** deste estudo é o de mostrar o ganho, em termos de segurança, obtidos quando se adota uma arquitetura de *Firewall* eficiente e adequada ao ambiente que deverá ser protegido.

Como **objetivos específicos** podem ser destacados o de apresentar as definições conceituais dos assuntos tratados, explicar o motivo da escolha de um Sistema Operacional seguro para a configuração dos *bastion hosts*, descrever as principais características das arquiteturas de *firewall* mais conhecidas, mostrar quais são as maneiras mais comuns de configuração de *firewall* nas empresas, mostrar exemplos de configurações seguras de rede.

O **método científico** de pesquisa utilizado foi o estudo teórico dos modelos clássicos de *firewall* e alguns outros modelos eficientes propostos por profissionais de segurança da informação. Será feita uma análise sobre alguns modelos de *firewall* comumente instalados nas empresas, de forma a verificar se há falhas de configuração que possam ocasionar problemas de segurança. Após a verificação dos possíveis erros serão propostos modelos de *firewall* que possam eliminar ou diminuir ou até mesmo eliminar tais problemas de segurança.

O estudo será puramente teórico e as arquiteturas propostas serão extraídas de livros, teses, dissertações, artigos e sites especializados em segurança da informação.

O trabalho foi estruturado em seis capítulos, sendo o primeiro esta introdução. O segundo apresenta definições e conceitos importantes para o entendimento dos capítulos posteriores. Já o terceiro capítulo foca na importância da configuração correta de um dos componentes apresentados no segundo capítulo: o *bastion host*.

O quarto capítulo apresenta as configurações comumente encontradas em ambientes empresariais. Já o quinto capítulo há uma discussão sobre os riscos de segurança encontrados nas configurações mencionadas no quarto capítulo.

Finalmente, o sexto capítulo conclui o trabalho recapitulando alguns tópicos e propondo alguns assuntos para serem tratados no futuro.

2 Definições e conceitos importantes

Antes de iniciar o estudo sobre arquiteturas de *firewall*, é necessário que sejam abordadas as definições de termos e a apresentação de conceitos importantes para o entendimento dos assuntos tratados por este trabalho.

2.1 Política de Segurança da Informação

Para se proteger algo, é necessário que se identifique o que, como e por que protegê-lo. Abaixo há uma visão geral sobre a Política de Segurança da Informação tendo como base a NBR ISO/IEC 27002 (2005).

2.1.1 Objetivo

De acordo com a NBR ISO/IEC 27002 (2005:8), o objetivo da Política de Segurança da Informação consiste em:

> Prover uma orientação e apoio da direção para a segurança da informação de acordo com os requisitos do negócio e com as leis e regulamentações pertinentes.

E complementa com:

> Convém que a direção estabeleça uma clara orientação da política, alinhada com os objetivos do negócio e demonstre apoio e comprometimento com a segurança da informação por meio da publicação e manutenção de uma política de segurança da informação para toda organização.

2.1.2 A informação e sua segurança

A informação é um ativo importante e essencial para os negócios da organização e precisa ser protegida de maneira apropriada. Como a interconectividade no ambiente dos negócios está crescendo dia a dia, a informação acaba por ficar exposta à muitas ameaças e, conseqüentemente, muito mais vulnerável.

A informação pode ser armazenada sob diversas formas, como, por exemplo, impressa ou escrita em papel ou mesmo em algum dispositivo de armazenamento eletrônico. Também pode ser enviada por correio tradicional ou meios eletrônicos, falada ou apresentada em filmes. De qualquer maneira, a informação precisa ser protegida apropriadamente.

A segurança da informação visa proteger a informação contra os mais variados tipos de ameaças, garantindo assim a continuidade do negócio, minimizando-se o risco ao mesmo e maximizando o retorno sobre os investimentos, além das oportunidades de negócio.

2.1.3 A necessidade da Política de Segurança?

Assegurar a competitividade, o fluxo de caixa, a lucratividade, o atendimento aos requisitos legais e a imagem da organização junto ao mercado estão entre os principais objetivos das organizações. Para que as ameaças à segurança não interfiram no bom funcionamento das organizações é preciso que suas redes de computadores e sistemas de informação estejam bem protegidas.

Entre as ameaças à segurança da informação encontram-se as fraudes eletrônicas, sabotagem, espionagem, vandalismo, incêndio e inundação.

2.1.4 A obtenção da Política de Segurança?

A NBR ISO/IEC 27002 (2005:X) apresenta uma visão geral sobre como a segurança da informação pode ser obtida:

> A segurança da informação é obtida a partir da implementação de um conjunto de controles adequados, incluindo políticas, processos, procedimentos, estruturas organizacionais e funções de *software* e hardware. Estes controles precisam ser estabelecidos, implementados, monitorados, analisados criticamente e melhorados, onde necessário, para garantir que os objetivos do negócio e de segurança da organização sejam atendidos. Convém que isto seja feito em conjunto com outros processos de gestão de negócio.

A NBR ISO/IEC 27002 é um guia para a organização que deseja iniciar, manter ou melhorar as diretrizes e princípios gerais da gestão de segurança da informação de uma organização e, com isso, criar um documento sobre a política de segurança da informação mais adequado à realidade da empresa.

2.2 *Firewall*

Há diversas definições para *firewall* disponíveis na literatura, mas a definição de Nakamura (2000:105-106) baseada, entre outros autores, na definição de Chapman (1995) é bastante explicativa:

> (...) *firewall* é um ponto entre duas ou mais redes, ponto este que pode ser um componente ou um conjunto de componentes, por onde passa todo o tráfego, permitindo que o controle e/ou autenticação e registro de todo o tráfego seja realizado. Assim, esse ponto único constitui um mecanismo utilizado para proteger, geralmente, uma rede confiável de uma rede pública, não-confiável. Um *firewall* pode ser utilizado também para separar diferentes sub-redes, grupos de trabalhos ou LANs dentro de uma organização (...)

Ele ainda oferece uma outra:

> (...) é um sistema ou um grupo de sistemas que reforça a política de controle de acesso entre duas redes, e portanto pode ser visto como uma implementação da política de segurança.

Logo, o *firewall* não é necessariamente composto por um único componente, mas pode ser constituído por diversos itens de *hardware* e *software* que tem por finalidade o controle, a autenticação e o registro do tráfego, visando à proteção da rede, mas, para que essa solução seja bem-sucedida, é necessário que a política de segurança seja definida de maneira clara. Isso deve ser feito para que o *firewall* possa, tecnicamente, fazer valer aquilo que foi definido de forma textual no documento de Política de Segurança.

A Política de Segurança de uma empresa é tão importante que Nakamura (2000:106) enfatiza dizendo:

> O *firewall* é tão seguro quanto à política de segurança que ele suporta (...)

2.2.1 Componentes do *Firewall* e termos utilizados com freqüência

É necessário apresentar algumas definições sobre os termos e componentes encontrados em um *firewall* para um bom entendimento sobre o seu funcionamento.

2.2.1.1 Host

É um computador que faz parte da rede, seja ele um servidor, computador pessoal, notebook, celular e etc., geralmente é conhecido como nó final.

2.2.1.2 *Bastion Hosts*

Segundo Chapman (1995:91-92) e Nakamura (2000:107), *bastion hosts* são equipamentos que disponibilizam serviços que podem ser acessados diretamente a partir da Internet. Essas máquinas possuem IP público, definido como categoria 3 na RFC 1918 REKHTER (1996:3), o que significa que elas são conhecidas e endereçáveis via Internet.

Contudo, possuir um IP público pode ser problemático, já que os *bastion hosts* poderão sofrer a ataques externos. Portanto, é necessário que essas máquinas sejam configuradas com muito cuidado.

No terceiro capítulo (Sistemas Operacionais e *softwares* de firewall) haverá uma discussão mais detalhada sobre a configuração dos *bastion hosts*.

2.2.1.3 Dual-homed host

É um sistema de computador (segundo Chapman (1995:58)) de uso geral que possui pelo menos duas interfaces de rede.

2.2.1.4 Pacote

Unidade fundamental de comunicação na Internet, conforme Chapman (1995:58).

2.2.1.5 Filtro de Pacotes

Ação de um dispositivo que controla seletivamente o fluxo de dados de uma rede, aceitando ou descartando pacotes mediante regras pré-determinadas. A filtragem de pacotes pode ser feita por qualquer equipamento que possua um *software* capaz realizar o filtro de pacotes IP, como um roteador ou um *host* que tenha instalado um *software* de filtragem de pacotes como o iptables, AYUSO *(*http://www.netfilter.org/projects/iptables/index.html).

2.2.1.6 Zona Desmilitarizada ou *Perimeter Network*

De acordo com Nakamura (2000:107) e Chapman (1995:58) por sua vez, zona desmilitarizada (DMZ) é sinônimo de *perimeter network*, que seria uma rede que separa a rede protegida da rede externa, mas também pode haver perímetros dentro de uma LAN.

2.2.1.7 Roteador interno ou c*hoke router*

É o roteador que contém um filtro de pacotes e protege a rede interna tanto de ameaças da Internet quanto da DMZ. Esse roteador fica entre a *perimeter network* e a rede interna. Nesse trabalho o mesmo será chamado de roteador interno.

2.2.1.8 Roteador externo ou a*ccess router*

Protege a rede interna e a DMZ da Internet. O roteador externo localiza-se entre a DMZ e a Internet. Neste texto ele será chamado de roteador externo.

2.2.1.9 *Proxy*

O ambiente configurado com o serviço de *Proxy* é composto por clientes *proxies* e por servidores *proxies*. Em sua configuração convencional, o servidor *Proxy* recebe as requisições dos clientes, que estão na rede interna, e as retransmitem para servidores externos. O servidor pode atuar apenas como um

relay, retransmitindo os pedidos dos clientes, ou pode realizar uma filtragem mais profunda dos pacotes.

Um dos *softwares* de *proxy* mais conhecidos e utilizados na comunidade de *software* livre é o SQUID (http://www.squid-cache.org/). O squid é um *caching proxy* para a Web que suporta, entre outros, os protocolos HTTP, HTTPS, FTP. Ele armazena as páginas mais visitadas em um dispositivo de armazenamento local. Sendo assim, se um usuário faz a requisição de uma página que já está armazenada pelo servidor squid, essa página será imediatamente enviada ao usuário, sem a necessidade de uma nova consulta ao servidor de origem. Uma vantagem dessa funcionalidade do *software* é a melhora do desempenho da rede, pois a página será acessada mais rapidamente pelo usuário. Outro benefício é o melhor aproveitamento da banda de acesso à Internet, já que, como foi dito, não haverá a necessidade de uma nova consulta ao servidor de origem.

Segundo Jucá (2005:267-268) o squid tem três modos de operação: *Proxy* convencional, *Proxy* transparente e *Proxy* reverso. No modo convencional, o cliente precisa configurar manualmente o endereço IP e a porta do *Proxy* para a solução funcionar corretamente. Já no *Proxy* transparente, não há a necessidade de configurações por parte do cliente, pois as conexões Web serão redirecionadas para o servidor *Proxy* automaticamente. E finalmente, o *Proxy* reverso permite que os clientes da rede externa, Internet, por exemplo, acessem os servidores da organização via *proxy*, ou seja, os *bastion hosts* serão acessados indiretamente. Esse último modo de operação visa evitar diversos tipos de ataques, além de controlar os acessos aos servidores, como pode ser visto na Figura 2-1.

O *proxy* é um componente importante nas configurações de *firewall* atualmente, como poderá ser visto no capítulo 5.

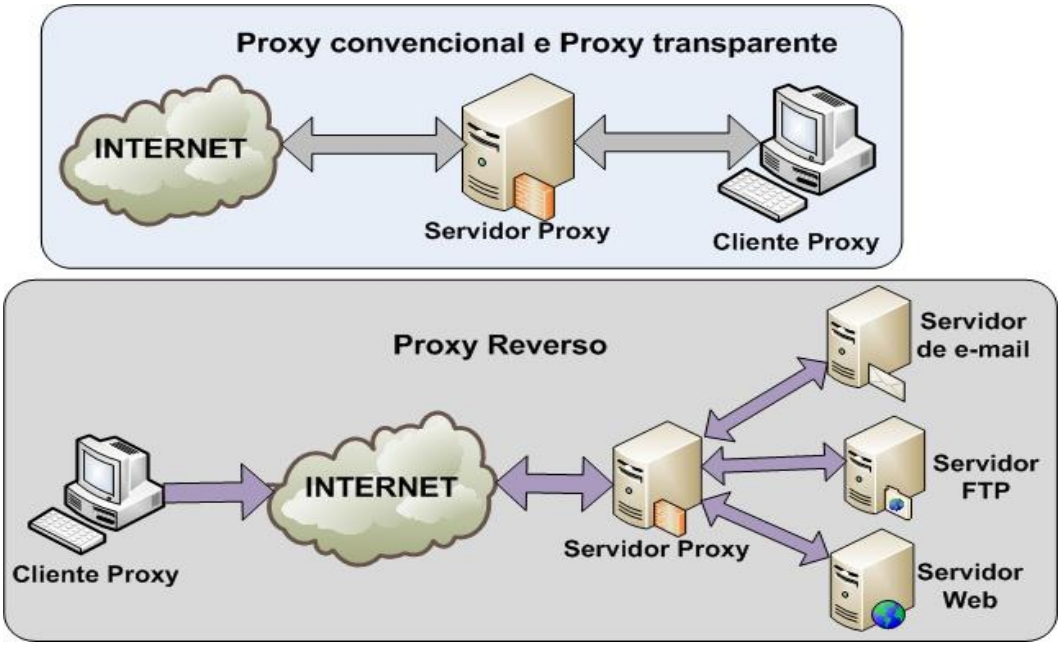

Figura 2-1. Proxy convencional, Proxy transparente e Proxy reverso

Fonte: Autor

2.2.1.10 Screening router

Screening router é um roteador que possui um filtro de pacotes.

2.3 Arquiteturas de *Firewall*

Um projeto de Arquitetura de *Firewall* consiste em dispor os componentes de *firewall* pela rede de forma a criar uma seqüência de barreiras e controles que têm por finalidade dificultar o tráfego malicioso nessa rede.

As arquiteturas mostradas abaixo são as combinações dos componentes descritos na sessão 2.2.

2.3.1 Dual-Homed Host Architecture

Para criar a *dual-homed host architecture* é necessário um *dual-homed host*, (conforme foi definido na sessão 2.2.1.3). Cada interface de rede dessa máquina se

conectará a uma rede diferente. No caso mais simples, uma interface estará ligada à rede interna, protegida, e a outra à rede externa.

Nesse tipo de arquitetura, representada pela Figura 2-2, a rede interna só se comunica com a rede externa através do *dual-homed host*, e vice-versa. As comunicações são geralmente feitas via *proxy*.

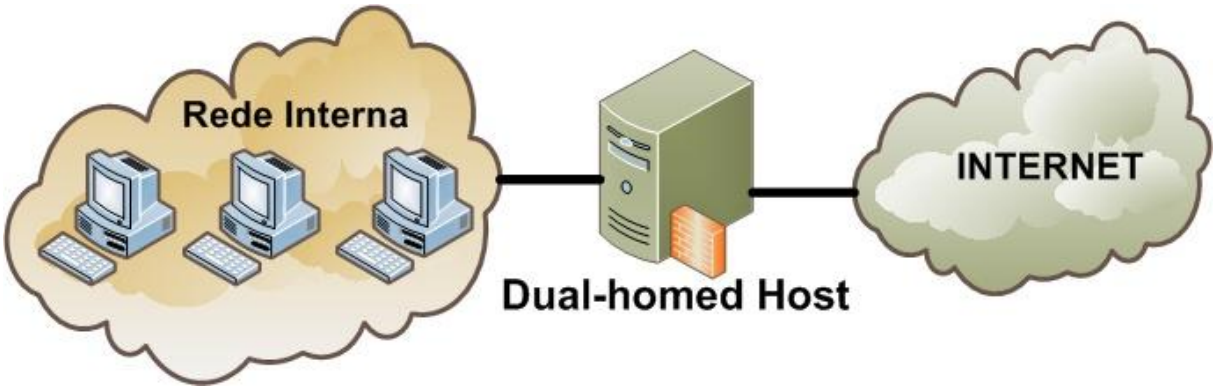

Figura 2-2. *Dual-homed host architecture*

Fonte: Autor

2.3.2 Screened Host Architecture

Como pode ser visto na Figura 2-3, essa arquitetura é composta por um *screening router*, que contém um filtro de pacotes, e um *bastion host*. O primeiro nível de segurança é feito pelo filtro de pacotes, que aplicará as regras pré-determinadas pela Política de Segurança e que, por exemplo, evitará que usuários da rede interna acessem a Internet diretamente. O acesso externo só será feito através do *bastion host,* que estará conectado somente à rede interna. O filtro de pacotes precisa garantir que o *bastion host* será o único computador da rede interna autorizado a abrir conexões diretamente com a rede externa, e também precisa definir quais conexões serão permitidas.

Essa arquitetura, do modo como foi apresentada na Figura 2-3, tem uma vantagem importante sobre a *dual-homed host architecture*, pois é mais fácil, segundo Chapman (1995:66), defender um roteador que um *host*.

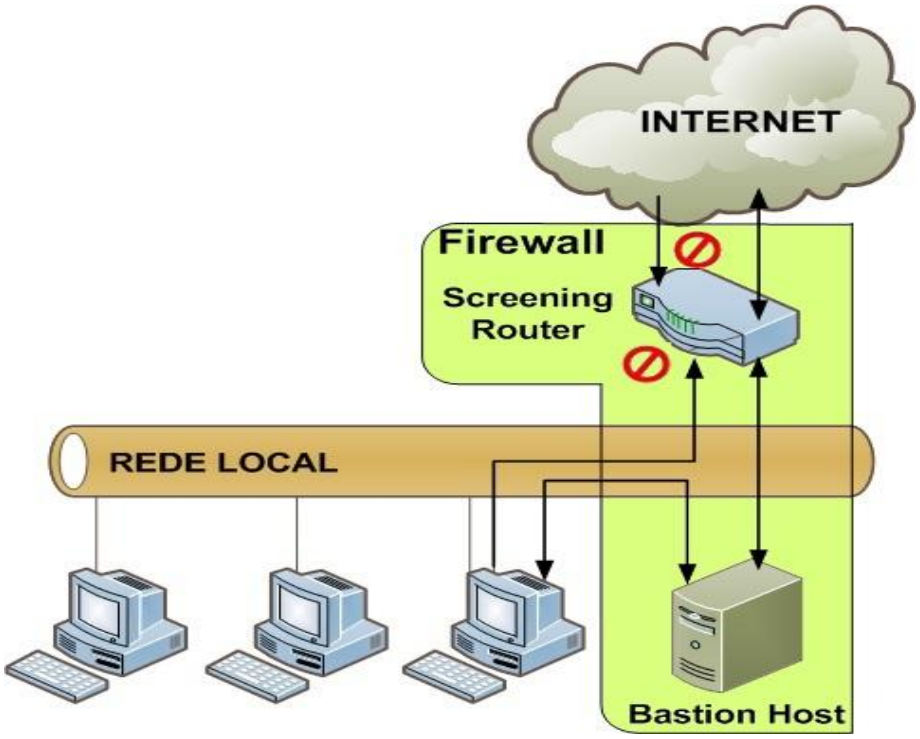

Figura 2-3. *Screened host architecture*

Fonte: Autor

2.3.3 Screened Subnet Architecture

A *screened subnet architecture* diferencia-se da *screened host architecture* devido à adição da DMZ que isola a rede interna da externa. Essa técnica incrementa mais uma camada de proteção à rede, já que o *bastion host*, que é a máquina mais vulnerável da rede – conforme Chapman (1995:66) - localiza-se dentro da DMZ. Isso significa que embora um atacante consiga acesso ao *bastion host*, ele não terá acesso imediato à rede interna.

Na Figura 2-4 pode-se notar que há dois roteadores com filtros de pacotes, ou seja, dois *screening router*s, conectados a DMZ. O roteador externo localiza-se entre a DMZ e a rede externa, nesse caso a Internet, já o roteador interno está entre a DMZ e a rede local. Logo, para um invasor ter acesso a rede interna ele precisará passar por esses dois roteadores, porém, se o roteador externo e interno permitirem os mesmos tipos de conexões, as camadas adicionais não surtirão efeito algum.

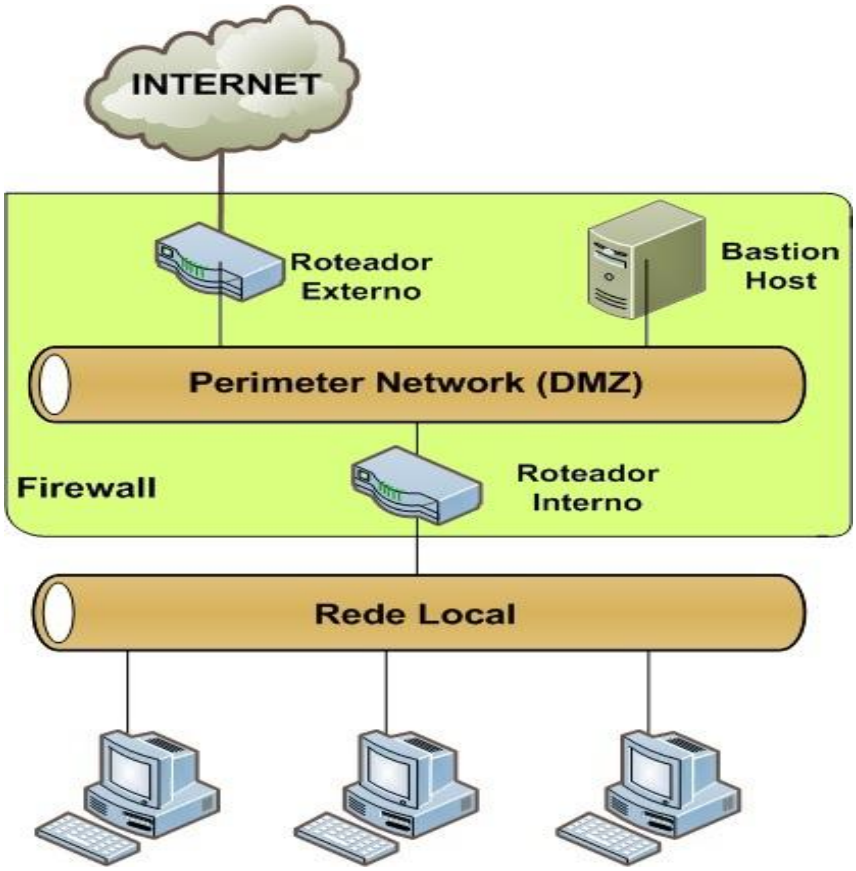

Figura 2-4. *Screened subnet architecture* (usando dois roteadores)

Fonte: Autor

Há casos em que são criados mais perímetros dentro da própria rede interna para dificultar ainda mais o acesso de possíveis atacantes.

2.3.4 Variações nas arquiteturas de *firewall*

Chapman (1995) diz que há diversas variações possíveis nas arquiteturas de *firewall*. Como será visto no quinto capítulo, nem sempre as arquiteturas aplicadas serão exatamente idênticas as apresentadas na sessão 2.3. Algumas variações são: o uso de múltiplos *bastion hosts*, mesclar o roteador interno com o roteador externo, mesclar o *bastion host* com o roteador externo, ter vários roteadores externos. Porém algumas outras configurações podem ser perigosas para a rede como, por exemplo: mesclar o *bastion host* com o roteador interno e ter muitos roteadores internos.

Na medida em que arquiteturas diferentes das apresentadas forem sendo usadas, haverá uma descrição do porquê da adoção da arquitetura em questão.

2.4 *Virtual Private Network* (VPN)

Conforme SCOTT (1999:2) a VPN é uma forma de simular uma rede privada tendo como base uma rede pública, como por exemplo, a Internet. É chamada de virtual porque depende do uso de conexões virtuais, ou seja, conexões temporárias que não são estabelecidas fisicamente.

Consideram-se redes públicas aquelas que possuem um grande número de pontos que trocam informações, mas não possuem relações entre si. A Internet e o sistema público de telefonia são dois bons exemplos de redes públicas.

As redes privadas, por sua vez, são constituídas por computadores de uma mesma organização que trocam informações especificamente entre si. Logo, a organização está segura de que as informações enviadas entre as máquinas, no pior caso, somente serão visualizadas pelos integrantes do grupo. A LAN é um exemplo de rede privada. Geralmente o que divide uma rede privada de uma pública é o *firewall*.

Antes do surgimento da VPN o método mais comum para a interligação entre duas redes privadas era a conexão dedicada. Assim, se uma empresa quisesse transferir dados com segurança entre a matriz e a filial seria necessário contratar um *link* dedicado. O problema é que esse tipo de tecnologia não era, e continua não sendo, barata.

A VPN transfere informações de maneira segura pela Internet utilizando diversas tecnologias para proteger os dados durante o tráfego por essa rede pública. As técnicas mais importantes são a autenticação, a criptografia e o tunelamento.

A autenticação garante que as partes comunicantes estão trocando informações com o usuário ou *host* correto. A autenticação geralmente é realizada digitando-se o nome do usuário e a senha, mas também pode ser realizada via

compartilhamento de chaves. Mais detalhes sobre compartilhamento de chaves e métodos de autenticação podem ser vistos nas sessões 2.6 e 2.7, respectivamente.

Todas as VPNs suportam algum tipo de criptografia. Há duas técnicas comuns de criptografia a criptografia de chave privada e a criptografia de chave pública, conforme a Figura 2-5.

Na criptografia de chave privada, há o compartilhamento da chave secreta por todas as partes que precisam do acesso VPN. Essa mesma chave é utilizada para a criptografia e decriptografia da informação.

Já na criptografia de chave pública há duas chaves: a pública e a privada. A chave pública pode ser fornecida para qualquer pessoa, mas à chave privada só o dono pode ter acesso. A mensagem cifrada com a chave privada somente será decifrada pela chave pública e vice-versa.

Muitos tipos de VPN usam o tunelamento para criar uma rede privada. O tunelamento permite que um pacote seja encapsulado dentro de outro pacote, mesmo que os protocolos sejam incompatíveis, por exemplo, IPX encapsulado dentro de pacotes de Internet (TCP/IP).

Dessa forma a VPN ajuda a baratear os custos da rede privada, já que consegue realizar conexões virtuais entre dois pontos sem a necessidade de links dedicados. Além disso, os funcionários de uma organização podem acessar a Intranet da empresa direto de seus computadores pessoais, pois existem *softwares* que realizam essa tarefa. E o mais importante é que tudo isso é feito de maneira segura.

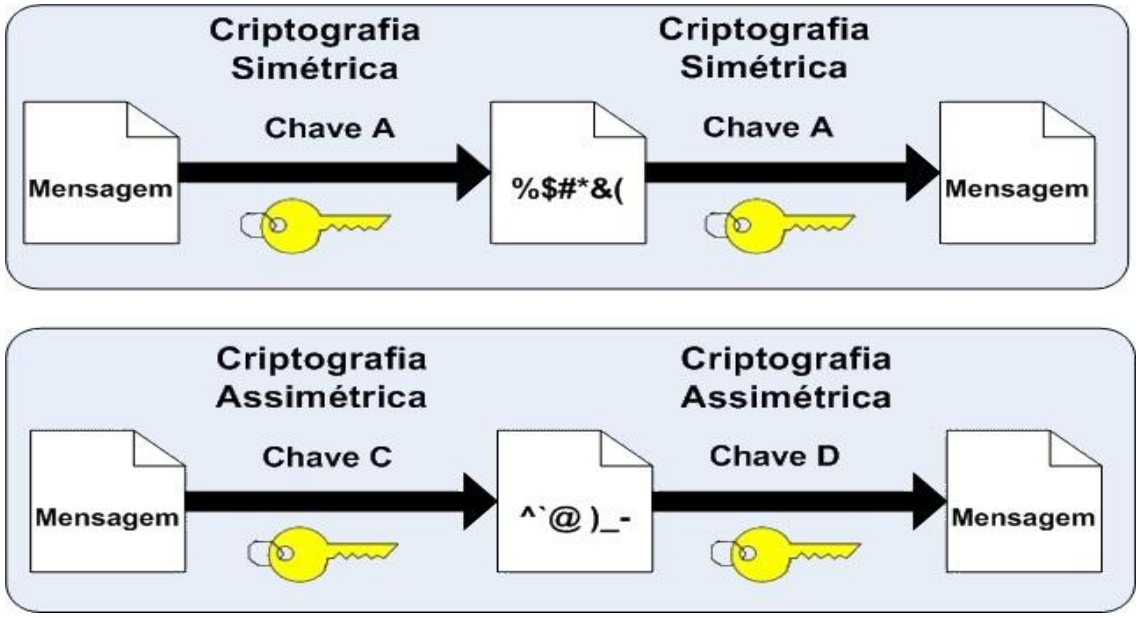

Figura 2-5. Comparação entre a Criptografia Simétrica e Assimétrica

Fonte: Autor

2.5 Sistema de Detecção de Intrusões e Sistema de Prevenção de Intrusões

KORFF (2005:336) apresenta que o sistema de prevenção de Intrusões (IDS) é, em linhas gerais, um programa ou máquina que procura por sinais que indicam que o meio está sendo atacado. O IDS é um sistema que age em modo passivo, monitorando o tráfego e alertando o usuário quando um ataque é detectado. Contudo, ele não realiza nenhuma ação para bloquear o ataque, apenas manda um sinal de alerta. A ação precisa partir do administrador, que deve analisar o alerta e tomar providencias para bloquear o ataque.

Como o IDS pode gerar uma grande quantidade de alertas, é preciso que o equipamento onde esse software será instalado tenha uma configuração avançada e possua bastante espaço de armazenamento.

Uma questão importante antes de implantar um IDS é verificar se a rede onde ele será instalado realmente precisa de um IDS. Conforme KORFF (2005), os sistemas de detecção e prevenção de intrusões só deveriam ser instalados após a implantação total do firewall. É preciso ter em mente que o IDS pode criar mais

problemas que soluções, já que os profissionais que administram a rede precisam ter bons conhecimentos dela para saber responder de maneira coerente aos alertas gerados pelo sistema de detecção de intrusões.

Há vários tipos de IDS, mas duas arquiteturas merecem maior atenção; a baseada no *host-based* IDS (HIDS) e o *network-based* IDS (NIDS)

O *host-based* IDS procura por sinais de ataque em um único computador, verificando se ocorreram modificações em arquivos importantes do sistema. O HIDS é capaz de monitorar ataques em várias interfaces de rede e pode ser customizado para um determinado *host*. Porém pode gerar um grande overhead, consumindo assim muitos recursos da máquina. Dessa forma seria necessário um equipamento com configurações mais avançadas.

O *network-based* IDS (NIDS) monitora o tráfego da rede em busca de sinais de ataque; Para isso ele coloca as interfaces de rede em modo promíscuo usando um *sniffer*. Há sensores espalhados pela rede e um servidor central que mantém o gerenciamento centralizado, tais sensores são responsáveis por ativar os *sniffers* e detectar os ataques. Quando o ataque é detectado, os sensores mandam os dados para o servidor central. É responsabilidade desse servidor central manter as assinaturas de ataques dos sensores atualizadas.

Como o NIDS consegue monitorar vários *host*s com poucos sensores, ele é mais econômico que o HIDS, porém, o NIDS tem uma desvantagem em relação ao HIDS: ele é menos preciso. Isso acontece porque ele não tem um agente instalado em cada *host*, portanto produz-se mais falso-positivos. O termo falso-positivo é usado para definir a falha do IDS ao interpretar uma atividade como sendo um ataque, quando na verdade não é.

Já o *Intrusion Prevention Systems* (IPS) é a versão ativa do IDS. O IPS não apenas reporta um ataque, mas também bloqueia o tráfego malicioso. O problema dessa solução é que no caso de falso-positivos, o tráfego legítimo pode ser bloqueado, portanto, é preciso avaliar se a instalação de um IPS compensa o risco de bloqueio do tráfego legítimo da rede.

2.6 Certificados Digitais e Infra-estrutura de Chaves Públicas (ICP)

Segundo VACCA (2004), os mecanismos de segurança costumam ser centralizados, devido à tradição da área de tecnologia da informação de concentrar a administração sob o controle de um único domínio de gerenciamento. Porém com o aumento do comércio eletrônico e a necessidade de funcionários trabalharem remotamente (*home office*), esse paradigma de segurança está sendo alterado. Cada vez mais os usuários precisam trocar informações entre domínios diferentes de forma segura, ou ainda, acessar, a partir de uma residência, recursos que estão disponíveis na LAN da empresa.

Para suprir essas novas necessidades, foram criados alguns mecanismos de segurança com o intuito de prover a troca segura de informações entre domínios. Quando uma empresa queria compartilhar dados confidenciais criava-se um segredo compartilhado, como um par de chaves iguais. Isso é conhecido como criptografia simétrica. Porém, havia um problema: como compartilhar essa chave comum de forma segura através de um meio inseguro? Uma solução foi a criação do conceito de criptografia de chave pública, também conhecida como criptografia assimétrica. A chave privada fica em poder do proprietário e a chave pública é disponibilizada ao público. Usando esta técnica é possível criptografar um texto usando a chave pública e apenas o detentor da chave privada poderá decriptografá-lo. Portanto, a chave simétrica poderia ser criptografada usando a chave pública e ser enviada para portador da chave privada de maneira segura.

Outra característica interessante da criptografia de chave pública é a possibilidade de realizar autenticação. Isso é feito quando um texto é criptografado utilizando-se a chave privada e posteriormente decriptografado pelo destinatário que possui a chave pública. O destinatário só conseguirá decriptografar a mensagem se ela tiver sido criptografa pela chave privada correspondente. Dessa forma, tem-se a garantia de que o texto foi enviado pela pessoa que detém a chave privada. Contudo, há um problema nessa técnica. Por exemplo, se o José tiver a chave pública de Pedro ele conseguirá verificar se as mensagens são realmente de Pedro, porém, quem garante que a chave pública que está com José foi realmente enviada por Pedro e não por uma terceira pessoa mal intencionada?

Para resolver o problema de insegurança na distribuição de chaves públicas foi criado o conceito de certificado digital. Esse certificado vincula algumas informações à chave pública (por exemplo, o nome de Pedro) e é assinado digitalmente por uma terceira parte confiável denominada autoridade certificadora (AC). Para verificar um certificado, o portador da chave pública deve obter essa chave de uma AC. Também é possível criar uma cadeia de confiança entre as autoridades certificadoras, ou seja, uma AC confia em outra AC que confia em outra AC e assim sucessivamente.

Para garantir o gerenciamento confiável dos certificados o uso de uma infra-estrutura de chaves públicas (ICP) é muito importante. A ICP permite que os usuários de uma rede pública insegura como a Internet possam de maneira segura e privada trocar dados através dos pares criptográficos de chaves públicas e privadas, compartilhando as chaves públicas através de uma autoridade confiável. A ICP possui sistemas para emitir, armazenar, determinar a autenticidade e revogar certificados cujas chaves foram comprometidas. Além disso, para efetivamente utilizar a criptografia de chave pública e assinaturas digitais, ele também provê o não-repúdio. A ICP garante que esses serviços trabalhem em conjunto e tenham uma compreensão comum de formatos e protocolos necessários para atingir os seus objetivos

Segundo VACCA (2004), a ICP consiste em:

- Uma autoridade de certificação que emite e verifica os certificados digitais

- A autoridade de registro, que atua como um verificador para a autoridade de certificação antes de um certificado digital ser emitido ao solicitante.

- Um ou mais diretórios onde os certificados (com as suas chaves públicas) são guardados.

- Um sistema de gestão de certificados

.

Ainda segundo VACCA (2004), uma boa implementação de ICP precisa satisfazer os seguintes requisitos:

- Não-repúdio: Para uma transação de negócio ser válida, nenhuma das partes pode mais tarde negar a existência ou a execução dessa transação. A ICP usa assinaturas digitais para satisfazer esse requisito.

- Privacidade: A privacidade é obtida através da criptografia de chaves pública e privada.

- Integridade: Na ICP a integridade é obtida através da assinatura digital, que é usada para provar que os dados não foram adulterados durante o transito.

- Responsabilização: A ICP oferece responsabilização, verificando a identidade dos usuários através de assinaturas digitais. Como as assinaturas digitais são mais seguras do que a combinação nome de usuário e senha, os usuários estão mais propensos a ser responsabilizados pelas suas ações.

- Confiança: Todo o conceito de ICP tem como base a confiança. Você confia na autoridade (AC) emissora. Se você não tem fé na AC emissora, então você não pode confiar em nenhum dos certificados emitidos por ela, ou nas organizações que os emitiram. Isso não significa que a organização não é confiável, mas que a AC da organização não o é.

Para Nakamura (2000:162) a ICP é importante, entre outros motivos, por eliminar a necessidade de armazenamento de um grande número de senhas e de múltiplos processos de autenticação. Além disso, ela pode ser usada para assinar documentos digitais.

2.7 Autenticação

Conforme já mencionado nessa sessão, há três métodos de autenticação. O primeiro é baseado naquilo que o usuário sabe, por exemplo, senha, chave criptográfica ou *Personal Identification Number* (PIN). A mais utilizada desses métodos é a senha.

Para um usuário ter acesso aos recursos da organização ele precisa passar por um processo de verificação. Esse processo tem por finalidade validar a identidade do usuário, para tanto, é necessário verificar a identificação do usuário e realizar a autenticação.

Nakamura (2000:192) diz que a identificação é a função na qual o usuário declara uma determinada identidade para um sistema. Já a autenticação é a função responsável pela validação dessa declaração de identidade do usuário.

Segundo Nakamura (2000:192-203) a autenticação valida a identificação dos usuários e concede a autorização para acesso aos recursos. Há três maneiras de realizar a autenticação: mediante alguma informação que o usuário sabe, ou sobre algo que ele possui, ou ainda alguma característica do usuário. Todos esses métodos possuem um ponto fraco. Dependendo do grau de segurança do sistema, usam-se dois desses métodos para autenticar o usuário.

A senha possui alguns pontos positivos, entre eles a familiaridade dos usuários com esse tipo de autenticação e a facilidade de implementação nos sistemas. Porém há pontos negativos como usuários podendo criar senhas fracas ou as anotarem em pedaços de papel, monitoramento de senhas através de *sniffers*, um atacante pode comprometer o banco de dados que armazena as senhas de todos os usuários, fragilidade contra ataques de força bruta.

O segundo método de autenticação é baseado naquilo que o usuário possui. Esse método consiste em verificar dispositivos que pertencem ao usuário. Entre esses dispositivos estão os dispositivos de memória (*memory token*) e os dispositivos inteligentes (*smart tokens*).

Os dispositivos de memória não processam informações, apenas as armazenam. Eles são quase sempre utilizados em combinação com as senhas. Um exemplo desse tipo de dispositivo é o cartão de banco.

Já os dispositivos inteligentes além de possuírem capacidade de armazenamento, ainda contam com circuitos integrados que atuam no processamento de certas informações. Os *smart cards* e os *smart tokens* são bons exemplos de dispositivos inteligentes. Eles geralmente são mais confiáveis que as senhas comuns, pois, implementam criptografia e podem realizar geração dinâmica de senhas, porém, tanto os dispositivos de memória quanto os inteligentes podem ser roubados, perdidos ou quebrados facilmente.

Finalmente, a autenticação baseada naquilo que o usuário é consiste em verificar características físicas ou comportamentais do usuário - biometria. No caso essas características podem ser faciais, geometria da mão, olho, impressões digitais, entre outras. Segundo Nakamura (2000:197), esse método é mais seguro que os anteriores. Entre suas vantagens estão a maior segurança, não dependem de dispositivos e não precisa lembrar-se de senhas.

Porém, há algumas desvantagens com relação aos equipamentos de biometria. O acúmulo de sujeira nos equipamentos que realizam as leituras de digitais, por exemplo, geram reclamações dos usuários devido à má higiene desses leitores. Também há receio por parte de alguns usuários com relação ao laser usado para a leitura da retina ou da íris do olho. Eles acreditam que essa luz poderia ser prejudicial à saúde.

Com o crescimento na quantidade de sistemas e serviços no ambiente empresarial, a demanda por métodos de autenticação seguros aumentou proporcionalmente. Contudo, alguns problemas surgiram em decorrência disso: a dificuldade de lembrar-se de várias senhas, o risco dos usuários anotarem as senhas em pedaços de papel, o aumento dos pedidos de ajuda ao help-desk (devido ao esquecimento das senhas). Uma das soluções para esses problemas foi o surgimento do conceito de *single sign-on* (SSO).

A idéia do *single sign-on* é possibilitar que os usuários acessem vários sistemas de modo transparente e unificado através de uma única autenticação. Segundo Nakamura (2000:200-201), as principais características de um SSO são:

- Combinação de nome de usuário e senha únicos;
- Único método de administração, centralizado ou descentralizado, onde as mudanças são propagadas através dos diversos sistemas da organização;
- Segurança robusta nas sessões de *logon* e no armazenamento das informações do usuário e da sua senha;
- Integração das regras de autorização nas múltiplas aplicações.

Uma solução eficiente de SSO é a instalação de uma infra-estrutura de chaves públicas (ICP) discutida na sessão 2.6.

3 Sistemas Operacionais e *softwares* de *firewall*

A escolha de um sistema operacional seguro e *softwares* eficientes para a configuração do ambiente de *firewall* é de extrema importância, já que uma escolha equivocada pode significar o comprometimento do equipamento e conseqüentemente da rede.

3.1 Sistemas Operacionais para *Bastion Hosts*

Como já foi dito, o *bastion host* é uma máquina que possui IP público e está sujeita a ataques externos. Por isso é necessário que o Sistema Operacional desse servidor seja configurado de maneira segura.

3.1.1 Princípios gerais

De acordo com Chapman (1995:92) há dois princípios gerais na configuração de um *bastion host*: mantê-lo simples e estar preparado caso ele seja comprometido.

A simplicidade na configuração do *bastion host* torna mais fácil mantê-lo seguro. Isso significa que poucos serviços devem ser instalados e com o menor número de privilégios possíveis, porém não se esquecendo que ele precisa cumprir o seu papel.

Mesmo com toda a atenção e cuidado ao configurar um *bastion host* é preciso estar preparado caso ele seja comprometido. Isso porque um dos principais problemas é que esse equipamento, provavelmente, será o elemento intermediário entre a rede interna e externa da empresa, o que significa que se ele for comprometido será a partir dele que os ataques à rede interna ocorrerão.

Uma das maneiras de diminuir o impacto causado pelo comprometimento do *bastion host* é colocar um *firewall* entre o *bastion host* e a rede interna, usando, por exemplo, a *screened subnet architecture,* representada pela Figura 3-1. Outra forma

de minimizar o problema seria criar uma política de segurança para os *host*s da rede interna, criando uma política de senhas seguras, instalação de antivírus, atualização diária dos sistemas operacionais. Há ainda uma terceira técnica que é conscientizar os funcionários sobre boas práticas de segurança.

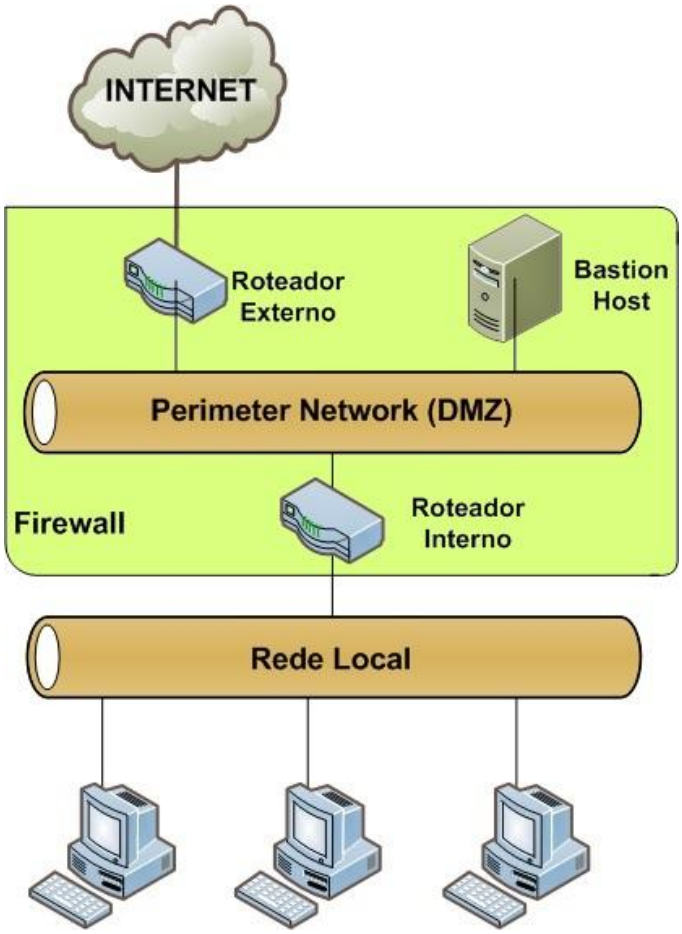

Figura 3-1. *Screened Subnet Architecture*

Fonte: Autor

3.1.2 O Sistema Operacional a ser usado

Chapman (1995:94) afirma que o melhor Sistema Operacional para se usar é aquele que se domina. Porém ele relata os benefícios de sistemas baseados em Unix, entre eles, a grande quantidade de ferramentas para a construção dos *bastion hosts*. A mesma dica é dada para se escolher a versão do Sistema Operacional, escolher aquela com que se tem maior familiaridade.

Porém, para uma empresa que pensa em economizar, sem perder a confiabilidade, uma boa opção seria, realmente, *bastion hosts* baseados em Unix. Por esse motivo o FreeBSD será usado nesse trabalho como modelo. Na sessão 3.1.5 será mostrado algumas ferramentas de segurança nativas desse Sistema Operacional.

3.1.3 *Hardware*

O *hardware* de um *bastion host* não necessita ser poderosíssimo. Basta que ele consiga fazer o trabalho para o qual foi concebido. De acordo Chapman (1995:97), um computador muito rápido pode ser um problema, pois, se ele for comprometido, o atacante teria uma máquina poderosa para atacar a rede interna.

3.1.4 A configuração de um Bastion *Host*

Chapman (1995:103-126) dá uma série de recomendações sobre como deve ser configurado um *bastion host*. Aqui serão mostrados os principais tópicos abordados por esse autor.

Em primeiro lugar é aconselhável realizar uma instalação limpa do sistema operacional, ou seja, formatar o HD e depois realizar a instalação. Também é recomendável fazer a instalação mínima, pois é importante conhecer todos os serviços que estarão em funcionamento no sistema.

Em segundo lugar, deve-se verificar se há algum problema de segurança conhecido com a versão do sistema operacional que está sendo instalada. Geralmente essa informação pode ser obtida no site do fabricante: deve-se realizar um *checklist* com os principais erros de segurança conhecidos e verificar se a instalação está isenta de problemas de segurança.

Os *logs* são outro ponto importante na configuração do *bastion host*. Duas medidas devem ser tomadas para garantir a disponibilidade e integridade dos logs. A primeira é manter um log no próprio *bastion host*, devido à facilidade caso o log precise ser consultado. A segunda é manter uma cópia em um servidor remoto, pois,

se um invasor ganhar acesso à máquina provavelmente ele alterará ou apagará os logs para encobrir seus rastros. Há algumas ferramentas no FreeBSD que fazem esse trabalho e estão descritas na sessão 3.1.6.

É recomendável instalar somente os serviços que realmente serão necessários para o *bastion host* desempenhar sua função. Isso porque quanto mais serviços esse computador possuir, maiores serão as chances dos pacotes apresentarem problemas de segurança. Também na sessão 3.1.6 é mostrado um *software* para FreeBSD que alerta sobre as vulnerabilidades de segurança de pacotes instalados no sistema operacional.

Em sistemas operacionais baseados em Unix/Linux é interessante se compilar o Kernel retirando todas as funcionalidades que não serão utilizadas. Também é importante verificar as partições do servidor e configurar, sempre que possível, o maior número de partições *read-only*. Isso é feito para que um possível atacante não consiga alterar os arquivos de configuração.

Após esses passos é importante executar algum *software* de auditoria do sistema para verificar falhas de segurança.

Somente após esses passos conecte o *bastion host* à rede. Caso haja a necessidade de acesso à Internet para realizar atualizações do sistema operacional, deve-se certificar-se que o servidor se encontra em uma rede própria para esse fim. Essa rede precisa possuir características específicas para instalação de servidores, devendo ter, por exemplo, filtros de pacotes que permitam acesso apenas a sites sabidamente seguros e importantes para a atualização do sistema operacional. A idéia é não expor o servidor à rede externa enquanto ele não estiver pronto para suportar possíveis ataques.

3.1.5 Componentes nativos de segurança do FreeBSD

Segundo Giraldi (2010), após a instalação do *bastion host* com FreeBSD, é interessante tomar algumas medidas de segurança para evitar problemas com esse sistema operacional. Algumas dessas medidas serão descritas abaixo.

No FreeBSD há um usuário chamado *toor* que é pré-configurado e possui os mesmos privilégios do usuário root. Esse usuário pode ser usado, por exemplo, no caso do esquecimento da senha do usuário root. Giraldi (2010) faz duas recomendações com relação a esse usuário, caso o administrador deseje usá-lo, configurar uma senha forte; caso contrário, apagar essa conta.

É recomendável que os *bastion hosts* fiquem fisicamente em locais onde apenas pessoas autorizadas têm acesso. Isso porque se um usuário não autorizado tiver acesso físico ao sistema ele pode reiniciar a máquina, entrar em modo monousuário e alterar a senha do root. A partir dessa alteração ele poderia ter acesso total ao servidor. Para evitar esse problema, é necessário realizar algumas alterações no arquivo /etc/ttys. Nesse arquivo encontram-se os terminais virtuais usados para o administrador interagir com o Sistema Operacional. Pode-se alterar esse arquivo de forma a mudar as palavras da última coluna de *secure* para *insecure*. Isso significa que o FreeBSD interpretará que o servidor está localizado fisicamente em um local inseguro e ao ser reiniciado em modo monousuário será solicitado a senha do root. Porém, o administrador não conseguirá recuperar a senha do root caso a esqueça.

A configuração acima também impede que um usuário conecte-se diretamente ao servidor com o usuário root. Primeiramente ele teria que se autenticar com um usuário comum para só depois ter acesso como root. Isso incrementa um nível de segurança no momento da autenticação no sistema.

Uma das maiores preocupações dos administradores é que um atacante consiga comprometer a máquina, alterar arquivos de configuração e instalar *softwares* maliciosos nela. Para reduzir esse risco o FreeBSD conta com alguns dispositivos interessantes.

O primeiro dispositivo são as chflags, que estendem as permissões padrão do FreeBSD, que são leitura, escrita e execução. Entre as várias chflags que o FreeBSD contém, duas são especialmente importantes para a segurança, sappnd e schg. A primeira delas é interessante para arquivos de log, pois modifica a permissão do arquivo de modo que ele só permitirá a concatenação de dados, não permitindo a

edição. A segunda chflag modifica o arquivo tornando-o imutável, e é interessante para arquivos binários e de sistemas, que não precisam ser alterados.

O segundo desses dispositivos é chamado de *kernel* securelevels, ou níveis de segurança do *kernel*. Esses níveis de segurança impõem restrições na forma como diversas operações são usadas pelo *kernel*. O *kernel securelevel* possui cinco níveis (-1,0,1,2,3), onde o -1, que é o padrão, é o menos seguro e o 3 é o mais seguro.

Utilizando o chflags e o *securelevel* é possível tornar o sistema imutável, mesmo para o usuário root, eliminando assim o risco de um atacante alterar arquivos do sistema. Dessa forma, mesmo que o invasor consiga acessar o *bastion host* com permissões do usuário root, ele estaria bastante limitado em suas ações.

3.1.6 *Softwares* de log e auditoria para o FreeBSD

O FreeBSD conta com diversas ferramentas que auxiliam o administrador na difícil tarefa de manter uma rede segura. Entre essas ferramentas estão o portaudit, o Syslog e o Syslog-NG.

O portaudit verifica possíveis vulnerabilidades contidas em *softwares* instalados. Dessa forma, se houver alguma vulnerabilidade conhecida com o *software* ele envia um e-mail para a conta que o administrador configurou no servidor. Essa verificação também pode ser feita a qualquer momento pelo administrador, com permissão de root, usando o comando: portaudit –a. A saída para esse comando pode ser vista na Figura 3-2.

```
Affected package: cups-base-1.1.22.0_1
Type of problem: cups-base -- HPGL buffer overflow vulnerability.
Reference: <http://www.FreeBSD.org/ports/portaudit/40a3bca2-6809-11d9-a9e7-0001020eed82.html>

1 problem(s) in your installed packages found.

You are advised to update or deinstall the affected package(s) immediately.
```

Figura 3-2. Saída do comando portaudit –a

Fonte: RHODES (Monitoring Third Party Security Issues)

É possível notar na Figura 3-2 que a saída do comando *portaudit -a* já exibe uma mensagem avisando que o update ou a desinstalação do pacote que apresenta vulnerabilidades deve ser feita imediatamente. Com isso o administrador consegue ser proativo e eliminar a vulnerabilidade do servidor antes que uma ameaça a encontre.

Outro *software* bastante interessante é o Syslog, ou System Log service. Ele é um processo que roda em *background,* e armazena, em algum local, geralmente um arquivo, os eventos que ocorrem com outros serviços do sistema. Uma evolução desse *software* é o Syslog-NG que permite a centralização dos logs em um servidor. Ele também é capaz de enviar esses logs pela rede de forma criptografada.

Logs são muito importantes em um ambiente corporativo, pois é através deles que se pode verificar ataques e obter informações sobre vulnerabilidades ou problemas no *bastion host*. É sempre importante manter cópias dos logs em servidores remotos, pois os atacantes geralmente tentam apagar os logs locais das máquinas comprometidas.

4 Configurações típicas de *firewall* em ambientes empresariais

Algumas das configurações de rede encontradas nos ambientes empresariais, de acordo com Nakamura (2000:204-233), serão descritas a seguir. O intuito deste capítulo é o de mostrar configurações comumente usadas nas empresas e no quinto capítulo apontar os principais erros de segurança encontrados nessas configurações.

Nakamura (2000) explica que os ambientes de rede empresariais evoluem na medida em que há um aumento das conexões na rede e da necessidade de compartilhamento de recursos. Em um primeiro momento a rede é criada apenas para conectar os recursos internos da organização, conforme mostra a Figura 4-1.

Figura 4-1. Rede Interna da Organização

Fonte: Autor

Já na Figura 4-2 pode-se ver o segundo passo na evolução da rede, decorrente da necessidade de comunicação entre a matriz e a filial, utilizando conexões dedicadas, que são caras. Até aqui os problemas de segurança são basicamente internos da rede, já que ainda não há acesso externo.

Os problemas de segurança da rede aumentam quando a rede passa a ter acesso à Internet. A conexão com a Internet permite que a rede interna, antes isolada, possa agora acessar esse grande conglomerado de redes que é a Internet.

Porém, a recíproca é verdadeira, ou seja, a rede interna da organização também pode ser acessada a partir da Internet. A Figura 4-3 mostra como seria essa rede.

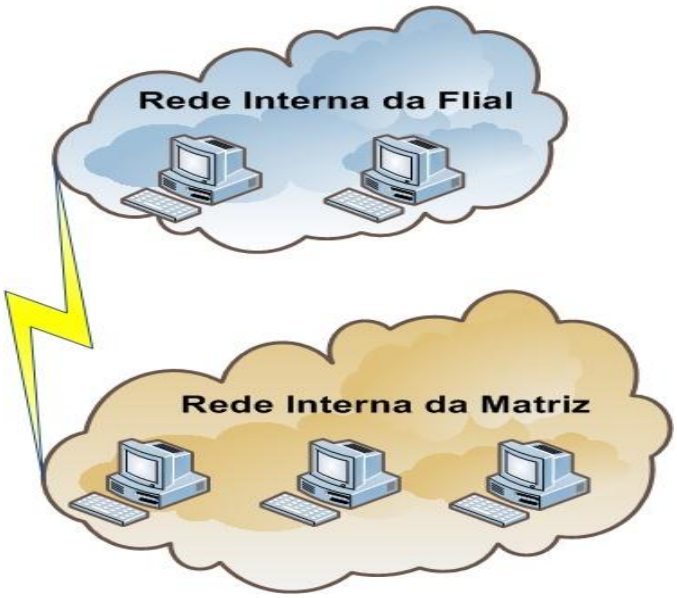

Figura 4-2.. Comunicação entre Matriz e Filial via conexão dedicada
Fonte: Autor

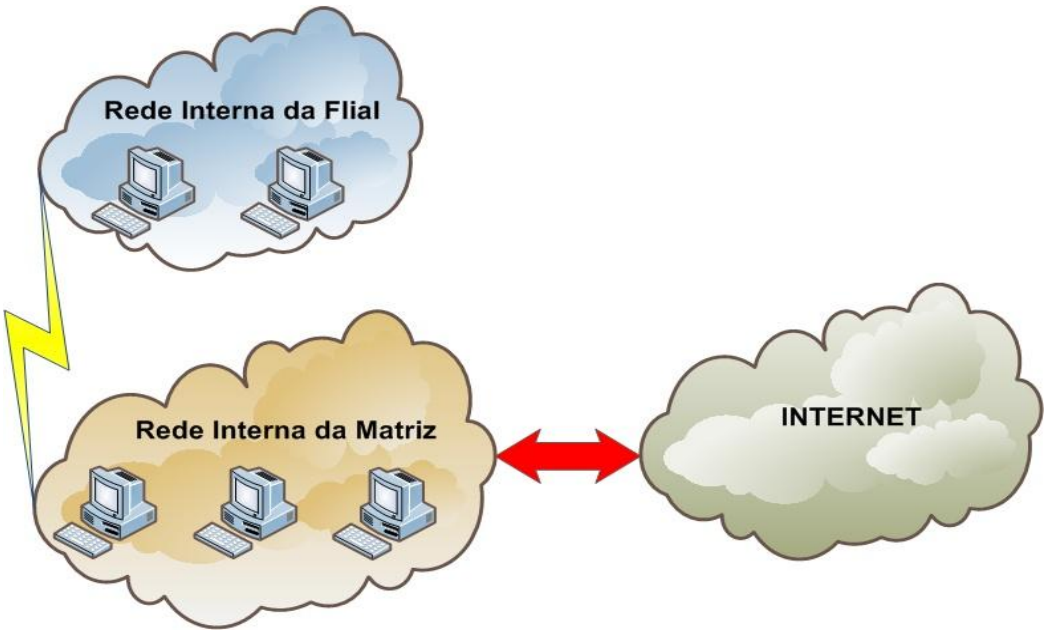

Figura 4-3. Rede da Matriz conectada diretamente com a Internet e a Filial conectada indiretamente.

Fonte: Autor

A grande maioria das empresas, ao providenciar uma conexão com a Internet, configura um *firewall* para isolar essa rede pública da rede privada da empresa, como mostra a Figura 4-4. A configuração mais comum utilizada nesse caso é bloquear as conexões que partem da rede pública e permitir àquelas que se originam na rede interna.

Contudo, quando a empresa possui servidores que passam a fornecer serviços para a Internet, a configuração do *firewall* tende a ficar cada vez mais complexa. Uma empresa, por exemplo, pode possuir servidores WEB, E-mail, FTP e Banco de Dados, posicionados na rede de acordo com a Figura 4-5.

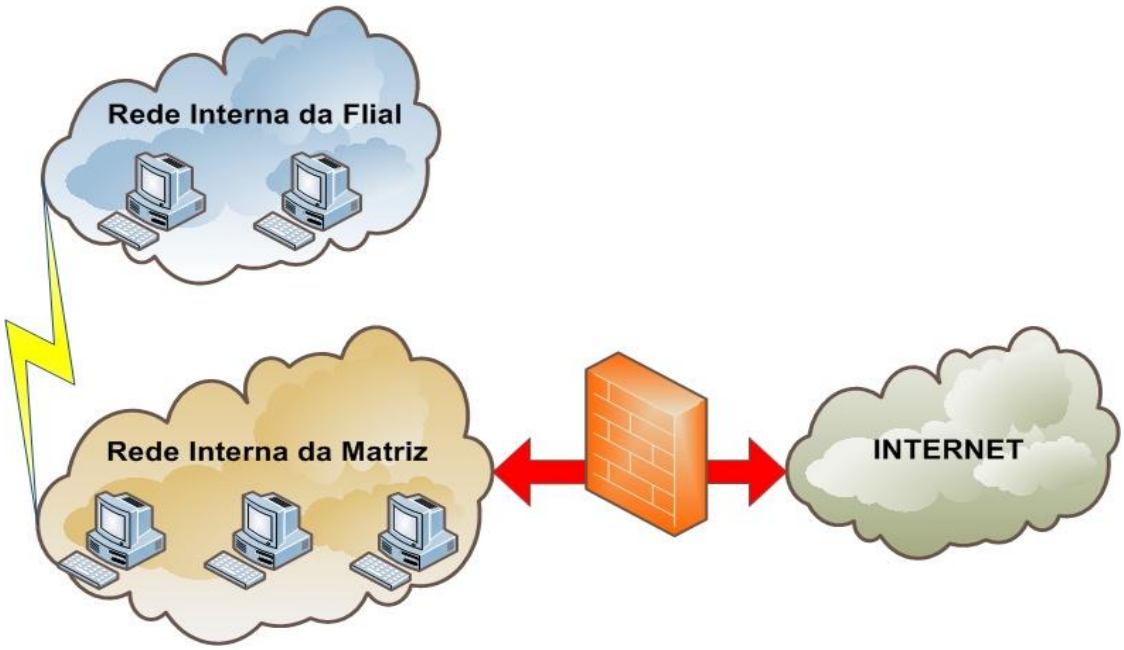

Figura 4-4. *Firewall* separando a Internet da rede privada da organização

Fonte: Autor

É possível notar na Figura 4-5 que não há nada separando a rede interna da organização dos servidores. Isso pode ocasionar problemas de segurança que serão discutidos no quinto capítulo

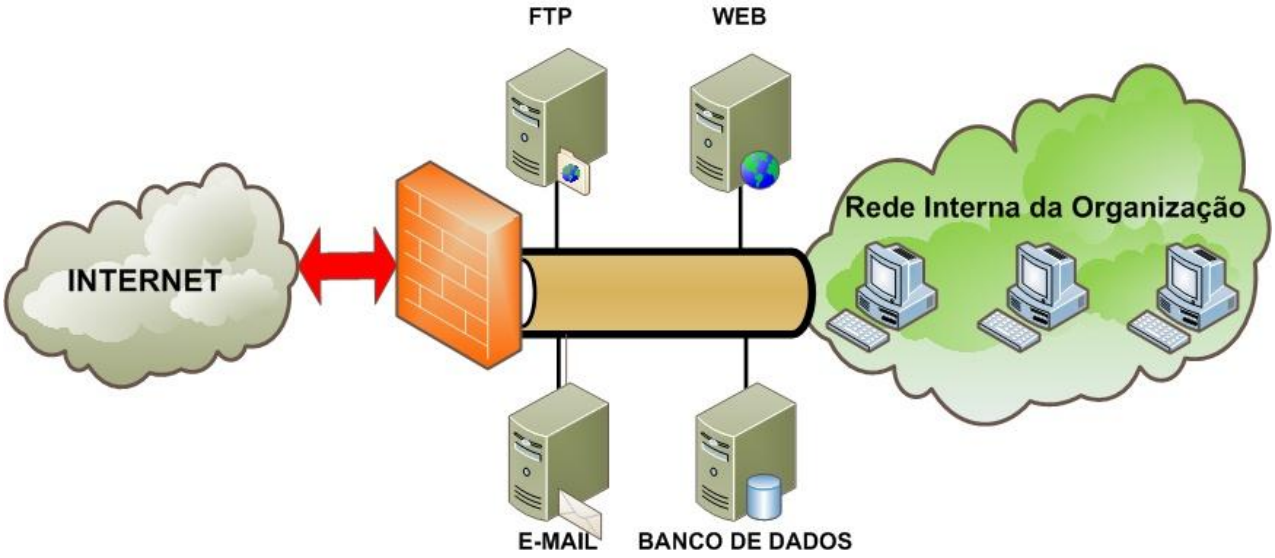

Figura 4-5. Servidores conectados diretamente a Rede Interna da Organização e separados da Internet por *Firewall*

Fonte: Autor

5 Modelos eficientes de arquitetura de *firewall*

O objetivo desse capítulo é analisar as proposições apresentadas no capítulo 4 e verificar os principais erros de configuração das redes apresentadas, expô-los, e propor alternativas seguras de configuração baseando-se em técnicas de arquitetura de *firewall* conhecidas e discutidas na sessão 2.3, além de outras variações dessas arquiteturas.

5.1 Soluções para pequenas e médias empresas apenas com acesso a Internet

Os problemas de segurança aumentam significativamente quando a empresa passa a ter acesso à Internet, como se pode observar na Figura 4-3. E como foi mostrado na Figura 4-4, a primeira providência, geralmente, que uma empresa adota é colocar um *Firewall* para separar a Internet de sua rede interna.

Quando uma organização possui apenas a rede interna e uma conexão com a Internet, é possível adotar algumas das arquiteturas de *firewall* vistas na sessão 2.3.

A primeira delas pode ser a *dual-homed host architecture*, representada pela Figura 5-1. Embora seja a menos segura, pois há apenas um ponto de falha, atenderia aquelas empresas que tem dificuldades em conseguir verba para investir em segurança. Como a *dual-homed host architecture* precisa apenas de um computador com duas, ou mais, placas de rede, conforme mostra a sessão 2.3.1, praticamente qualquer empresa poderia adotá-la.

É possível configurar uma *dual-homed host* com *softwares* livres sem prejudicar a segurança. Para isso é importante tomar alguns cuidados descritos no capítulo 3. Após a instalação do sistema, será necessário configurar um *software* de *proxy*, como por exemplo o SQUID que foi apresentado brevemente na sessão 2.2.1.9.

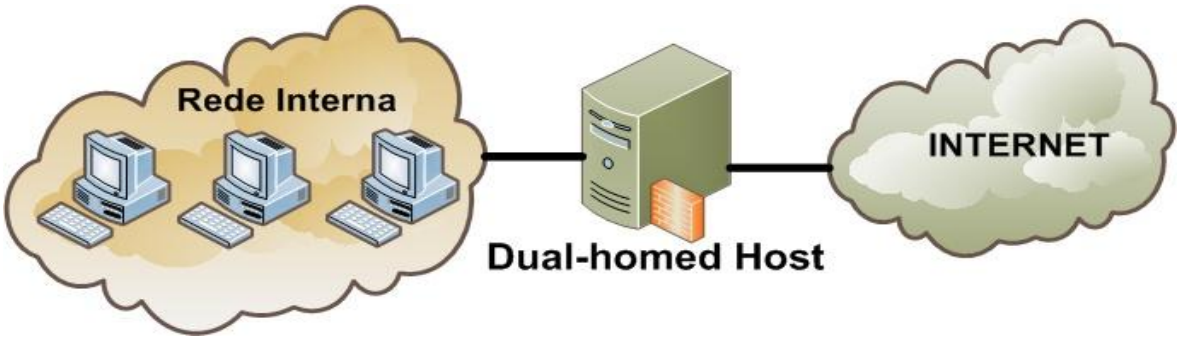

Figura 5-1. Dual-homed Host Architecture

Fonte: Autor

Outra configuração possível seria utilizando o *screened host architecture*, descrita na sessão 2.3.2. Na Figura 5-2 é possível notar que há um roteador (*screening router*) e um *bastion host*. Isso encarece a solução de *Firewall*, pois seria necessário comprar dois equipamentos: um roteador com filtro de pacotes, e um computador. Por outro lado, o filtro de pacotes acrescenta mais um nível de proteção para a solução.

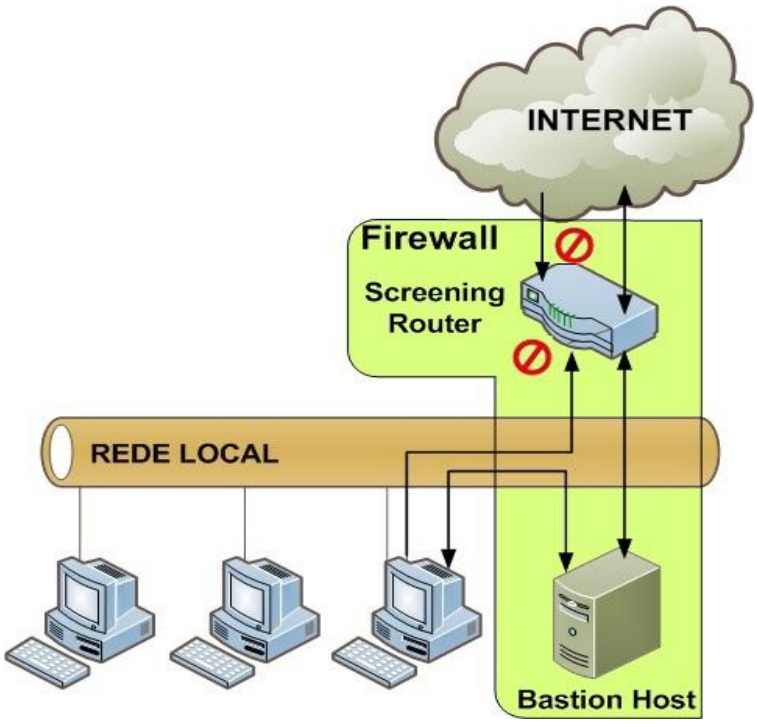

Figura 5-2. *Screened Host Architecture*

Fonte: Autor

Uma terceira solução é adotar a *screened subnet architecture*. Como foi visto na sessão 2.3.3. Essa arquitetura acrescenta um componente bastante importante em termos de segurança: a DMZ ou *perimeter network*. O modo padrão de configurar essa solução pode ser vista na Figura 5-3. A *screened subnet architecture* é composta basicamente de um roteador externo, um *bastion host* e um roteador interno. Dessa forma, temos um filtro de pacotes externo, que protege o *bastion host* e a rede interna; um filtro de pacotes interno que protege a rede interna; e o *bastion host*, que intermedeia os acessos dos *host*s entre a rede interna e a rede pública, através de um *Proxy*.

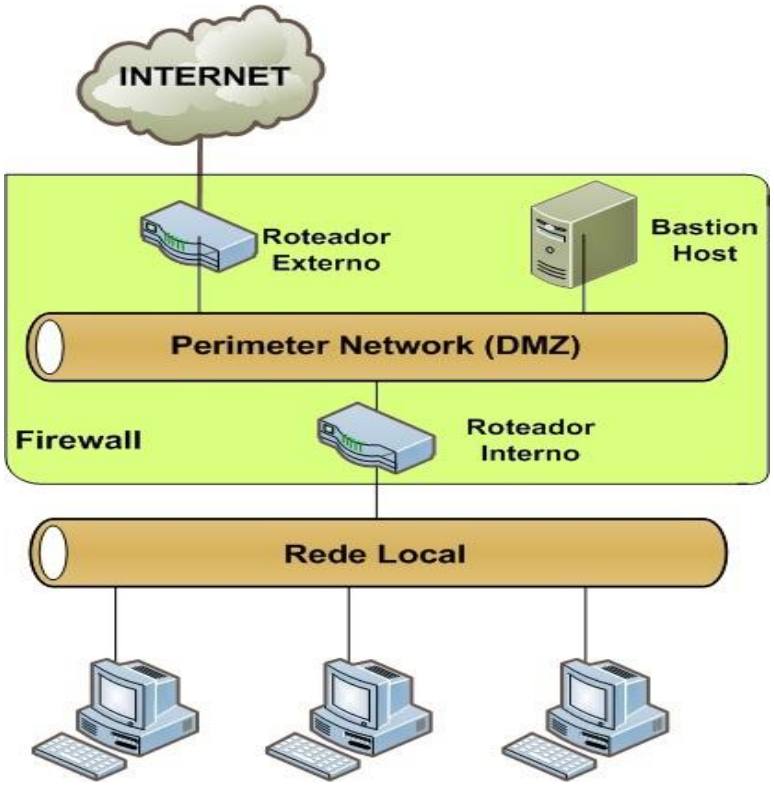

Figura 5-3. *Screened Subnet Architecture*

Fonte: Autor

Entre as três soluções apresentadas nesta sessão, sem dúvida a *screened subnet architecture* é a mais segura. Porém, nem sempre as empresas brasileiras dispõem de recursos para implementá-la, já que seriam necessários dois roteadores e um computador. Mas, segundo Chapman (1995:74-75) e Nakamura (2000:208-210) não há problemas em agregar os serviços do *bastion host* com o roteador

externo, pois isso não criaria grandes problemas de segurança. Essa configuração não apresenta grandes riscos de segurança porque o *bastion host*, embora complemente a proteção dada pelo roteador externo, não é a segunda linha de proteção para este. Assim, se o roteador externo for comprometido o invasor precisa acessar também o roteador interno para invadir a rede interna, sem necessariamente ter que acessar o *bastion host* para isso. A Figura 5-4 mostra a fusão entre o *bastion host* e o roteador externo.

Essa fusão utiliza um roteador a menos, portanto, só haveria a necessidade de um computador e um roteador. O computador deverá possuir um *software* de filtro de pacotes e um *software* de *proxy*, enquanto o roteador interno deverá possuir um filtro de pacotes. Mas Chapman (2005:74-76) alerta que é perigoso mesclar o *bastion host* com o roteador interno, pois essa configuração alteraria a arquitetura do *firewall*.

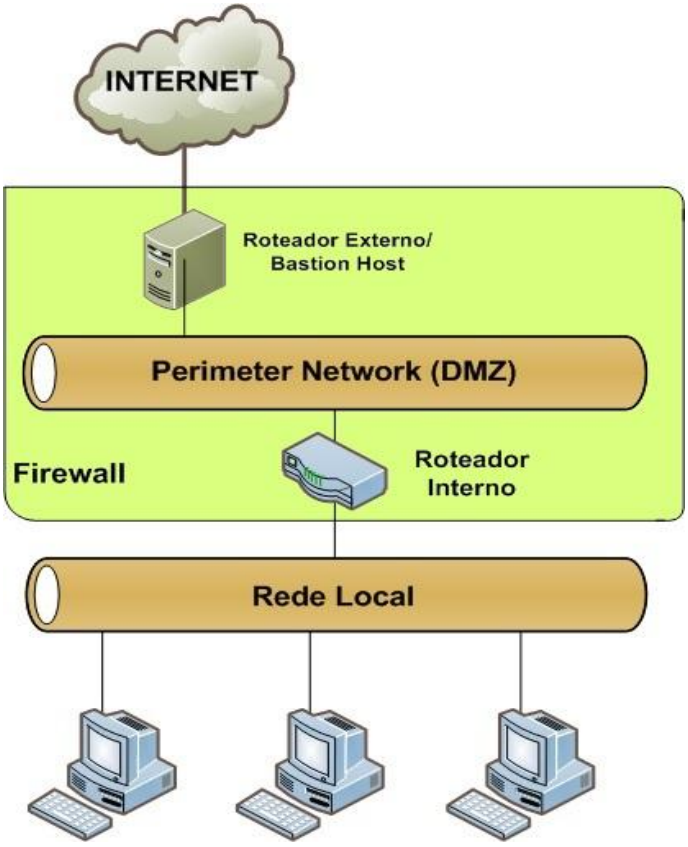

Figura 5-4. Arquitetura usando a fusão do *bastion host* com o roteador externo

Fonte: Autor

Isso ocorre porque nessa arquitetura o roteador interno funciona como uma segunda linha de proteção no caso de falha do *bastion host*. Porém, se o *bastion host* for responsável também pelas funções do roteador interno, não haverá nada que proteja a rede interna no caso de invasão do *bastion host*, portanto, fundir o roteador interno com o *bastion host* é perigoso, pois, o *bastion host* não ficaria mais isolado na DMZ. Essa fusão faria com que o *firewall* se comportasse como a *screened host firewall architecture* e não mais como a *screened subnet firewall architecture*. A Figura 5-5 mostra como ficaria a fusão, não recomendada, entre o *bastion host* e o roteador interno.

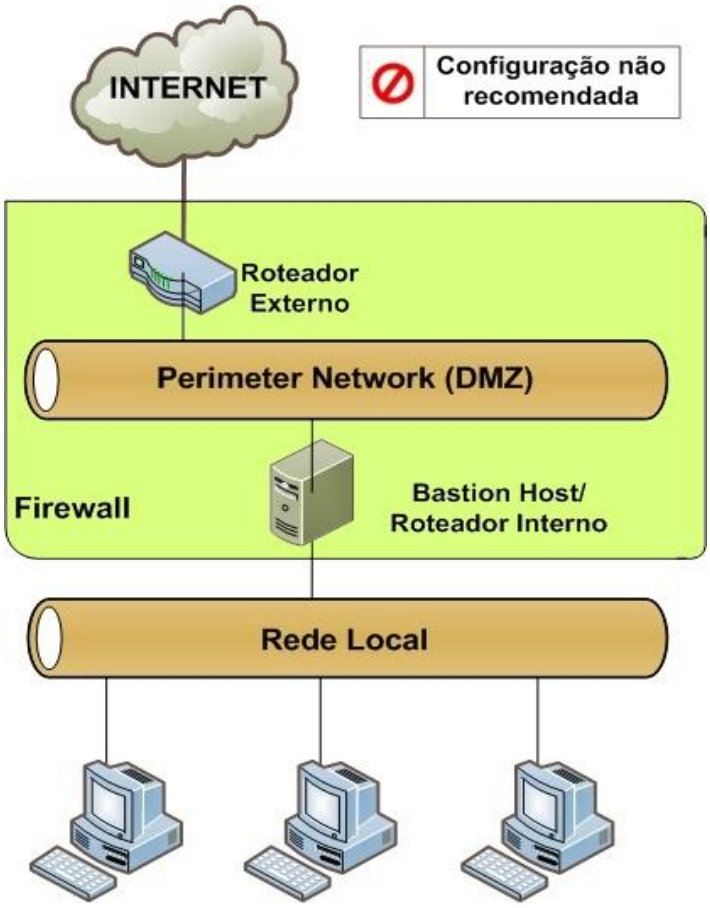

Figura 5-5. Arquitetura usando a fusão do *bastion host* com o roteador interno

Fonte: Autor

5.2 Configuração para grandes empresas com acesso a internet

No caso de grandes empresas, podem ocorrer problemas de desempenho devido à presença de um único roteador interno. Além disso, a empresa pode desejar ter redundância para evitar a parada da rede no caso de falha desse roteador. Porém Chapman (1995:75-78) não recomenda colocar vários roteadores internos ligados a DMZ, conforme mostra a Figura 5-6. Primeiro, porque a configuração desses vários roteadores é bastante complexa. Em segundo lugar, dados sensíveis da rede interna poderiam, por erro de configuração, trafegar pela DMZ. Isso poderia provocar um grande problema de segurança se algum invasor conseguir acesso ao *bastion host*, pois ele poderia capturar, através de um *sniffer*, esse tráfego passando pela DMZ.

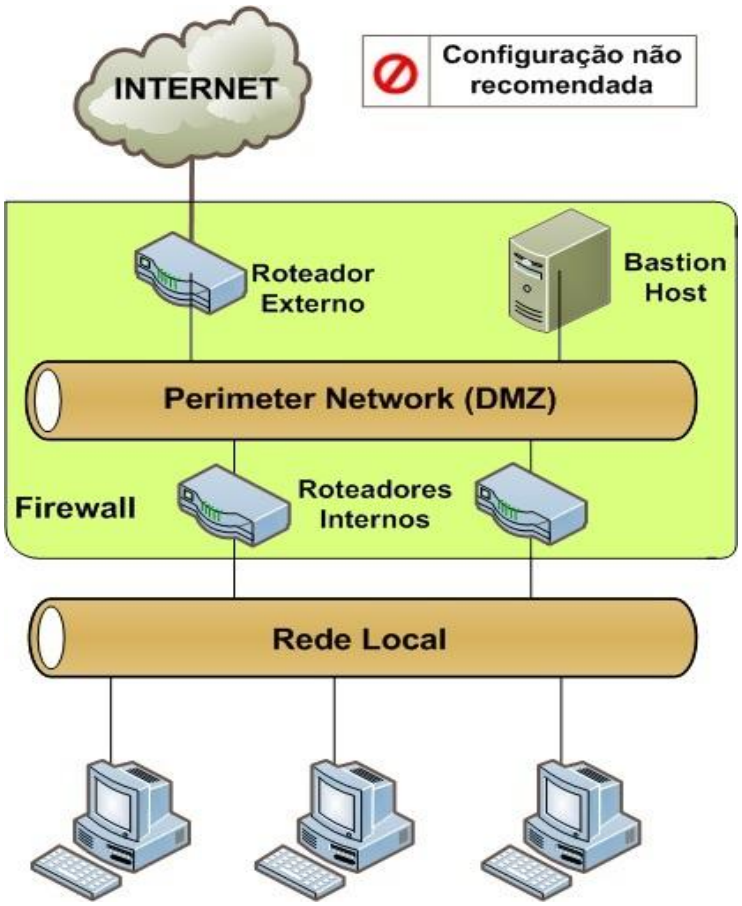

Figura 5-6. Arquitetura usando múltiplos roteadores internos

Fonte: Autor

Outra razão para se ter vários roteadores internos, segundo Chapman (1995:77-78), seria a presença de várias redes internas. Uma solução seria conectar cada uma dessas redes nas interfaces de um único roteador, conforme a Figura 5-7. Embora esse método aumente bastante a complexidade da configuração do roteador, não haverá os riscos apresentados pela presença de vários roteadores internos na rede.

Contudo, se houver realmente muitas redes para um único roteador ou se a solução com um único roteador não for tecnicamente possível ou ainda se não satisfizer as necessidades da organização, é possível criar um *backbone* interno e conectá-lo à DMZ com um único roteador. A Figura 5-8 mostra como a configuração com o *backbone* funciona.

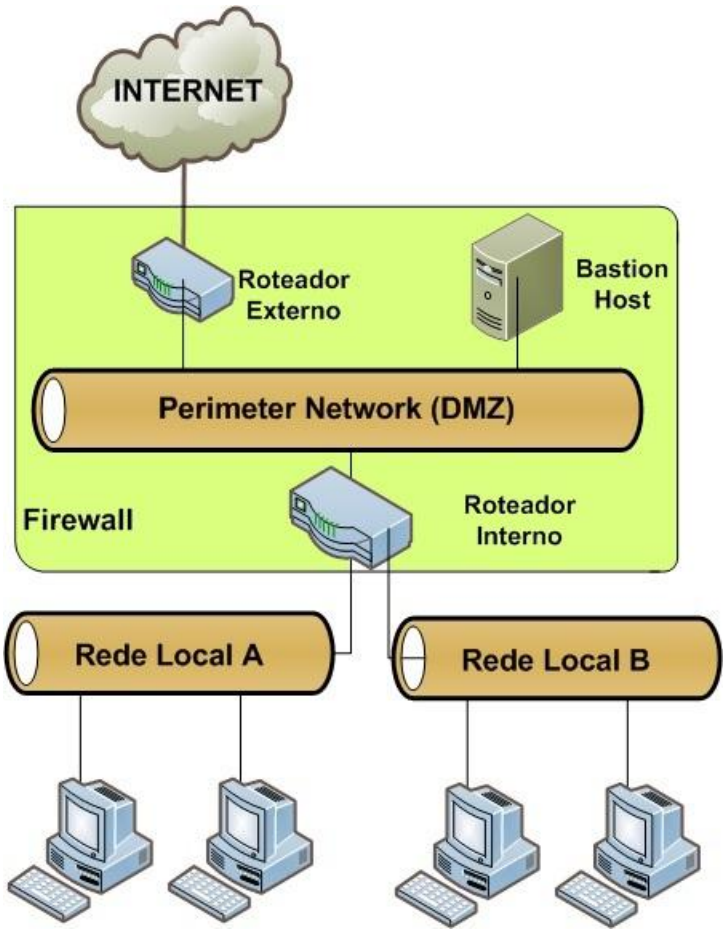

Figura 5-7. Múltiplas redes internas separadas por interfaces de um único roteador

Fonte: Autor

Dessa forma, mesmo que os roteadores posicionados entre as redes locais e o *backbone* trocassem informações entre si, esse tráfego ficaria restrito ao *backbone* e não mais à DMZ. Portanto, mesmo que um invasor consiga acesso ao *bastion host*, ele não conseguirá visualizar o tráfego da rede interna. Essa arquitetura é chamada por Chapman (1995) como *backbone architecture.*

5.3 Configuração de rede para empresas que possuem servidores de Internet

Um próximo passo na evolução das redes nas empresas é começar a prover serviços de Internet, ou seja, instalar servidores na rede para disponibilizar serviços para a rede pública, esses servidores podem ser de Web, e-mail, FTP entre outros. Porém, para essas máquinas funcionarem na Internet elas precisarão ter um IP público, conforme foi visto na sessão 2.2.1.2. Isso significa que esses servidores estarão “visíveis” na Internet para potenciais atacantes.

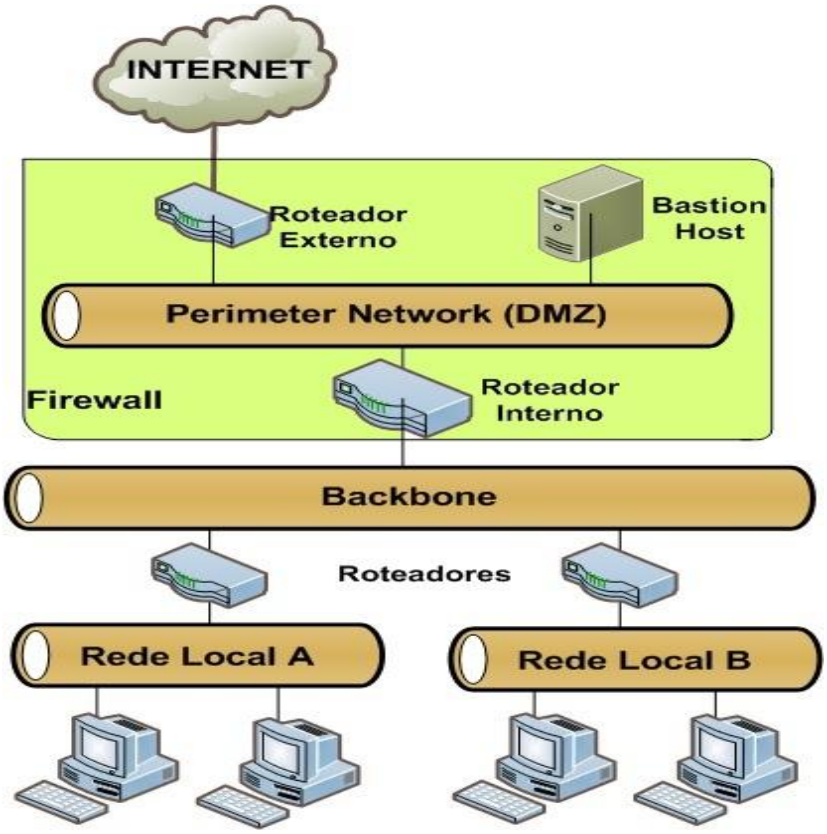

Figura 5-8. Múltiplas redes internas (*backbone architecture*)

Fonte: Autor

Por isso é muito importante que os servidores sejam configurados com muito cuidado, como foi comentado no capítulo 3. Caso uma máquina seja comprometida as perdas podem ser bem maiores do que apenas a indisponibilidade da rede. Por exemplo, se o servidor Web de uma grande corporação for invadido e a página inicial for alterada, isso pode provocar a perda de credibilidade da empresa perante seus clientes.

Além desses ataques aos *bastion hosts*, o aumento do número de servidores em uma rede pode significar novos perigos à rede interna. Isso ocorre porque com mais *bastion hosts* conectados à rede, maiores serão as chances de um atacante invadir um desses sistemas e tentar comprometer a rede interna. Por esse motivo a rede representada pela Figura 5-9 apresenta graves problemas de segurança, já que está conectada diretamente a rede interna da empresa.

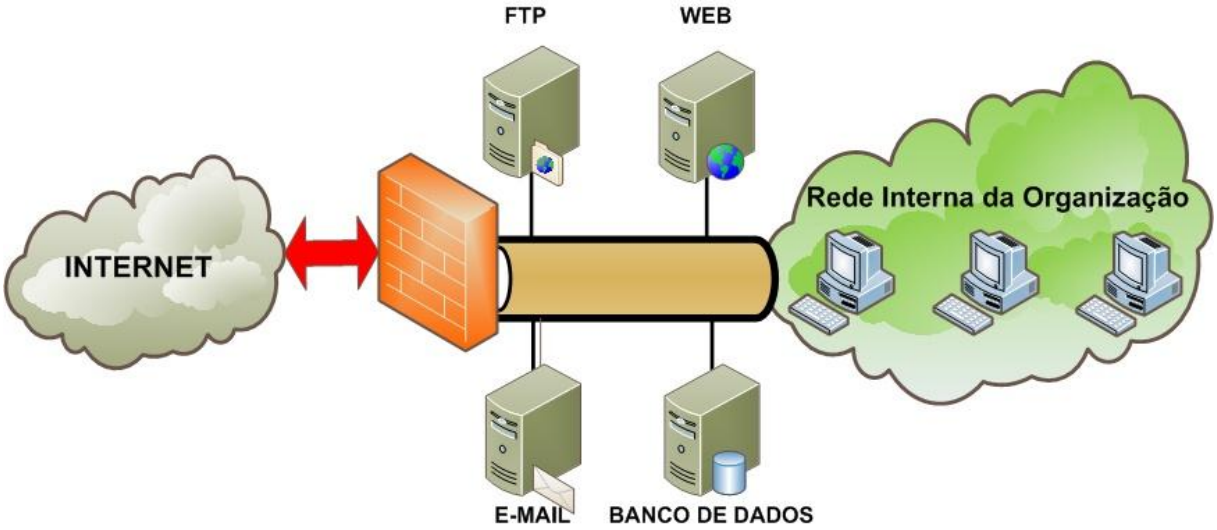

Figura 5-9. Servidores conectados diretamente a Rede Interna da Organização e separados da Internet por Firewall

Fonte: Autor

A primeira medida a ser tomada seria aplicar a *screened subnet architecture*, ou seja, deixar os *bastion hosts* isolados em uma DMZ, conforme mostra a Figura 5-10.

Na Figura 5-10 os muros de tijolos representam os dois roteadores com filtros de pacotes. É possível notar que os *bastion hosts* precisam passar por um desses filtros de pacotes para alcançar a rede interna da organização. Isso garante uma camada extra de proteção para a rede interna.

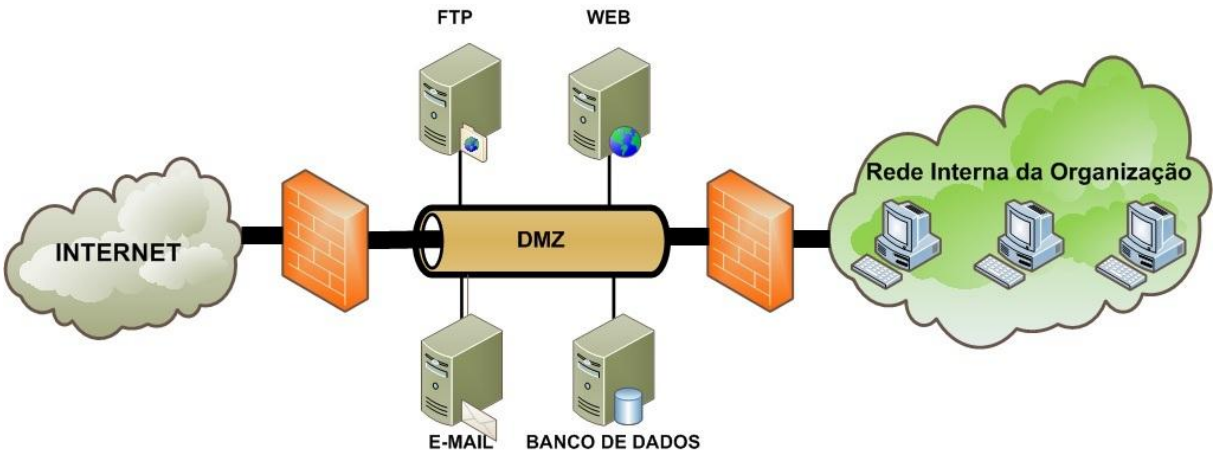

Figura 5-10. Servidores localizados na DMZ

Fonte: Autor

Porém, segundo Nakamura (2000:210-213) ainda há alguns problemas com a configuração apresentada na Figura 5-10. O primeiro deles é referente ao servidor de banco de dados. Por ser um serviço crítico, pois pode conter toda a base de dados de clientes da empresa, ele precisa de cuidados especiais. O roubo de informações como as de cartões de crédito de clientes, por exemplo, poderia resultar em grandes prejuízos para a empresa e seus clientes.

Uma boa prática para melhorar a segurança dos servidores de banco de dados é colocá-los em uma DMZ separada, como mostra a Figura 5-11.

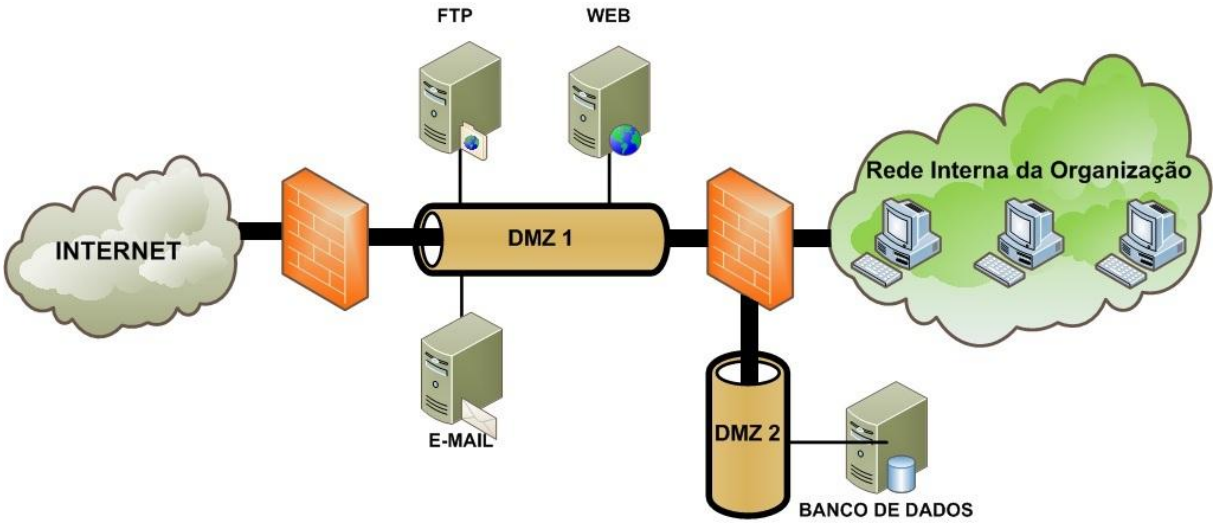

Figura 5-11. Servidor de banco de dados separado em uma segunda DMZ

Fonte: Autor

Mas colocar o banco de dados separado em outra DMZ não basta. É preciso que sejam definidas na política de segurança da empresa as regras de acesso para esse servidor. Por exemplo, em uma empresa de vendas on-line poderia ser definido que a única máquina que teria acesso direto ao servidor de banco de dados seria o servidor WEB. Isso evitaria que um atacante conseguisse invadir o servidor de banco de dados diretamente da Internet. Com a proposta apresentada acima, para o invasor conseguir comprometer o servidor de banco de dados ele precisaria invadir primeiro o servidor Web. Portanto, colocar o servidor de banco de dados em uma DMZ separada e criar uma política de acesso a ele, dificulta bastante o trabalho dos possíveis invasores.

5.4 Empresas que possuem o serviço VPN

Outro serviço de rede que está se tornando cada vez mais comum é a VPN, descrita na sessão 2.4. Ela pode ser usada para transportar com segurança informações entre a matriz e a filial, como e-mails e documentos. Porém, além do requisito básico, a informação não pode trafegar em claro pela rede, ainda há duas preocupações essenciais quando se instala uma VPN. A primeira é garantir que os dados não serão alterados no meio do caminho. A segunda é garantir que apenas os usuários legítimos possam usar esse serviço.

Uma solução interessante para garantir a confiabilidade e a integridade pode ser vista na sessão 2.6 que fala sobre infra-estrutura de chaves públicas (ICP). Outros métodos de autenticação podem ser visto na sessão 2.7. Porém a ICP, embora seja mais complexa de implementar, é um método de autenticação mais seguro que o par de usuário e senha. O ICP possui também outras vantagens como o não-repúdio, integridade, privacidade, responsabilização e confiança. Portanto, autenticar uma VPN em um ICP é bastante interessante do ponto de vista da segurança e praticidade.

Porém, para a solução de ICP funcionar corretamente é preciso que os certificados digitais sejam confiáveis (sessão 2.6). Para isso é necessário que o servidor de autoridade certificadora (AC) seja posicionado em um local seguro no *firewall*. Segundo Nakamura (2000:223-224), esse local seria a DMZ 2, que é o mesmo local onde o banco de dados se encontra. Analogamente ao servidor Web, que é o único que acessa diretamente o servidor de banco de dados, o servidor de VPN seria o único a acessar diretamente o servidor AC, conforme mostra a Figura 5-12. Nessa figura, a conexão entre o servidor VPN e o servidor AC é representada pela linha verde, enquanto a conexão entre o servidor WEB e o servidor de banco de dados é representada pela linha azul.

A Figura 5-12 também mostra a posição mais indicada na rede para o servidor VPN. Ele está posicionado na interface dedicada do *firewall* externo. Nakamura (2000:231) diz que essa posição é a mais recomendada, pois o servidor VPN é protegido pelo *firewall* contra ataques da rede pública e também os pacotes cifrados podem passar pelo *firewall*, ser decriptografados pela VPN, e ser filtrados pelo *firewall* de acordo com a política de segurança implementada.

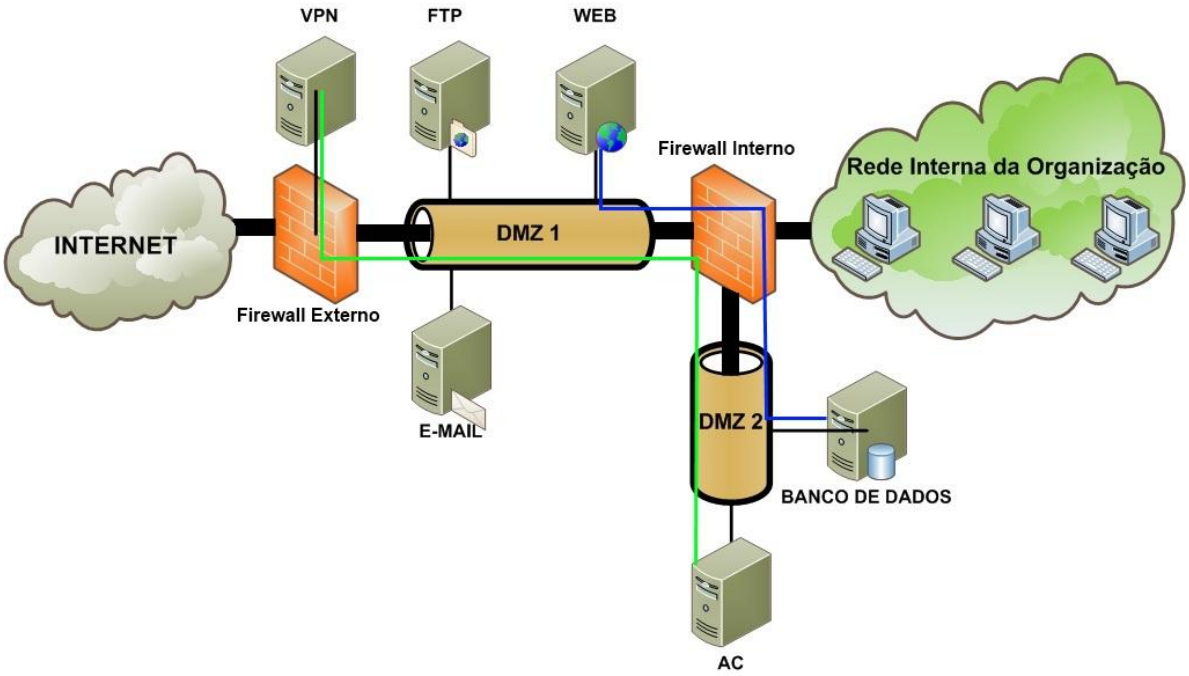

Figura 5-12. Servidor VPN localizado na interface dedicada do *firewall* externo e Servidor AC na DMZ 2

Fonte: Autor

5.5 Adicionando sistemas de detecção ou prevenção de intrusões

Conforme foi descrito na sessão 2.5, o sistema de detecção de intrusões (IDS) e o sistema de prevenção de Intrusões (IPS) precisam que o ambiente de rede esteja bem configurado para que esses serviços sejam bem aproveitados. Portanto esse é o momento correto de falar sobre IDS e IPS. Isso porque, como foi visto ao longo desse capítulo, as técnicas de arquitetura de *firewall* e o posicionamento correto dos componentes e serviços na rede geraram um ambiente sensivelmente mais seguro e organizado.

Nakamura (2000:225-227) informa três posições interessantes para se colocar o IDS na rede: em frente ao *firewall* externo, após o *firewall* externo e dentro da rede Interna. A Figura 5-13 mostra esses locais. Como foi explicado na sessão 2.5, o IPS é um IDS ativo, ou seja, ele bloqueia o tráfego malicioso. Ele poderia ser posicionado nos mesmos locais que o IDS.

Quando o IDS ou IPS é posicionado em frente ao *firewall* externo, IDS/IPS 1 da Figura 5-13, detecta-se ou previne-se ataques contra o *firewall*. O IDS/IPS 2 detecta ou previne ataques que passaram com sucesso pelo *firewall*. Já o IDS/IPS 3 detecta ou previne ataques internos na organização.

Como pôde ser visto ao longo desse capítulo, a disposição correta na rede dos componentes de *firewall* e dos *bastion hosts* tornam a rede mais segura. Embora não seja possível criar uma rede à prova de invasões, certamente é possível dificultar bastante as investidas de pessoas mal intencionadas. O objetivo de toda essa preocupação com a segurança é resguardar a rede e os negócios da empresa.

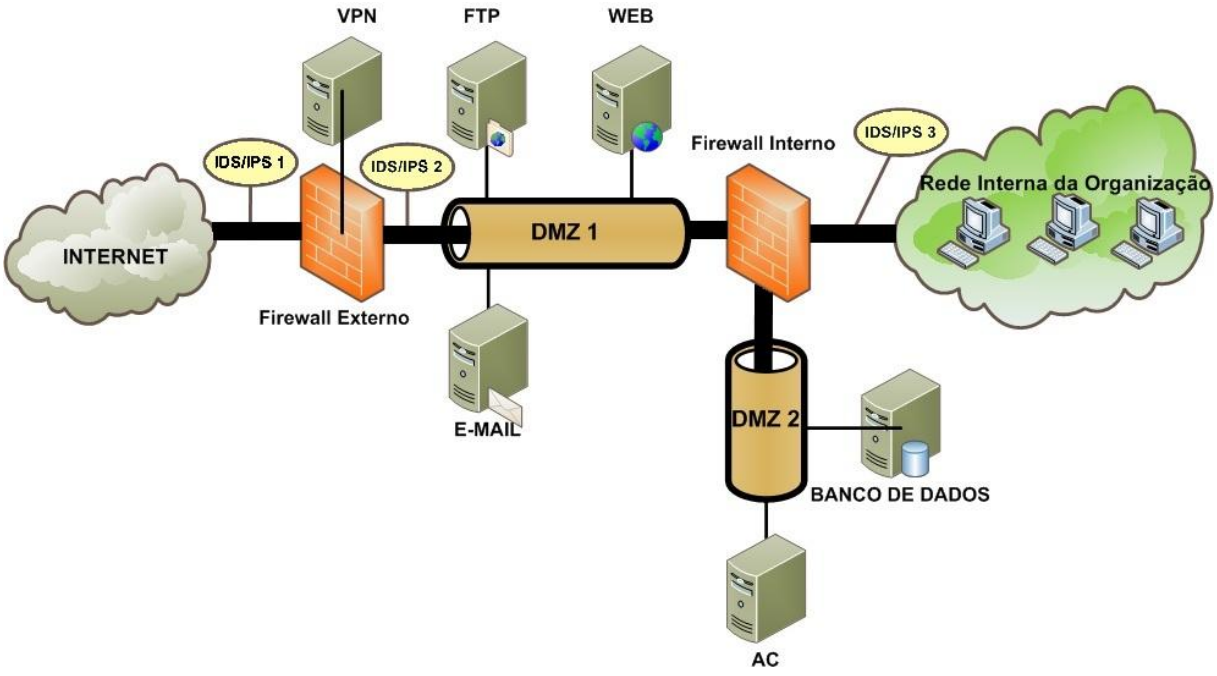

Figura 5-13. Posicionamento do IDS e IPS na rede

Fonte: Autor

6 Conclusão

A política de segurança, vista na sessão 2.1, é um documento essencial para o bom funcionamento da rede e da segurança do ambiente empresarial como um todo. Na sessão 2.2, fica claro que o *firewall* apenas ajuda a implementar tecnicamente o que foi definido pela política de segurança, logo, se a política de segurança for falha o *firewall* também será.

Na sessão 2.3 foram definidas algumas arquiteturas de *firewall* clássicas. Essas arquiteturas utilizam componentes descritos na sessão 2.2. E fica bastante claro que para criar uma configuração eficiente de *firewall* não basta agrupar vários componentes, pois a disposição errada desses componentes pode trazer riscos à segurança da rede.

A VPN e o IDS, descritos respectivamente nas sessões 2.4 e 2.5, são serviços que estão se tornando cada vez mais essenciais para as empresas modernas. A VPN permite que empresas criem redes privadas sobre redes públicas e o IDS monitora a rede em busca de potenciais atacantes.

Na sessão 2.6 o ICP foi descrito, assim como os benefícios em se adotar essa tecnologia. Entre eles estão: a autenticação digital de documentos, o não-repúdio, a confiabilidade no envio de documentos e outras funções importantíssimas para as empresas da era da informação. Entre essas funções está a autenticação, que pode ser feita de diversas formas, conforme foi discutido na sessão 2.7

No terceiro capítulo foi dada uma atenção especial a um dos componentes do *firewall*: o *bastion host*. Como esse componente é suscetível a ataques, a sua configuração precisa ser minuciosa. Questões como a escolha do *hardware*, sistema operacional e configuração de *softwares* foram tratadas nesse capítulo.

O quarto capítulo mostra algumas configurações típicas usadas nos ambientes empresarias. Finalmente, o quinto capítulo faz uma análise das configurações apresentadas no capítulo 4 e apresenta arquiteturas de *firewall* que

visam diminuir os riscos de segurança da rede e garantir o bom funcionamento dos negócios da empresa.

A preocupação com segurança nos ambientes empresariais tem crescido em ritmo acelerado. Nos últimos anos, devido à melhora e ao barateamento nos serviços de banda larga no Brasil, a quantidade de empresas com acesso à Internet tem crescido muito. E junto com a facilidade e os benefícios que a Internet proporciona também vêm os riscos.

Entre os benefícios estão as vendas on-line, que podem impulsionar enormemente os lucros da empresa. Porém, caso dados confidenciais de clientes, como os números dos cartões de crédito, forem roubados, a empresa pode perder credibilidade ou até sofrer processos milionários por parte dos clientes. Portanto, para usufruir das vantagens das transações on-line é necessário ter um ambiente de rede seguro.

A proposta do trabalho é justamente mostrar técnicas de arquiteturas de *firewall* eficientes para ajudar na criação de redes seguras, que possam sustentar os negócios da organização e possibilitar o crescimento da empresa no mundo digital.

Uma sugestão de pesquisa para o futuro é analisar e comparar sistemas operacionais e *softwares* livres que permitam construir uma solução corporativa de segurança com alto desempenho e confiabilidade a um baixo custo. Para atender os requisitos de segurança será necessário um estudo mais detalhado sobre infra-estrutura de chaves públicas e aplicativos, como filtros de pacotes e *softwares* de segurança. Este estudo ajudaria bastante as pequenas e médias empresas brasileiras, que nem sempre possuem recursos para manter soluções caras de segurança, a competirem em pé de igualdade no mercado digital com as grandes empresas.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Citação**: NBR-10520/ago - 2002. Rio de Janeiro: ABNT, 2002.

_____. **Código de prática para a gestão da segurança da informação**: NBR ISO/IEC 27002/ago. 2005. Rio de Janeiro: ABNT, 2005.

_____. **Referências**: NBR-6023/ago. 2002. Rio de Janeiro: ABNT, 2002.

AYUSO, P. N. What is iptables? Disponível em <http://www.netfilter.org/projects/iptables/index.html>. Acesso em: 28 abril 2011. 01h02.

CHAPMAN, D. Brent; ZWICKY, Elizabeth D. Building Internet *Firewalls*. O'Reilly & Associates, Inc. 1995.

CERT.br. Cartilha de Segurança para a Internet 3.1. Disponível em < http://cartilha.cert.br/glossario/ >. Acesso em: 14 maio 2011. 17h35.

GIRALDI, M. V. L; SOUSA R. L. de. Administração Básica de servidores: FreeBSD. Campinas, 2010. 148 f.

JUCÁ, H. L. Técnicas Avançadas de Conectividade e *Firewall* em GNU/LINUX. Rio de Janeiro: Brasport, 2005. 396p.

KORFF, Y; HOPE, P; POTTER, B. Mastering FreeBSD and OpenBSD Security. O'Reilly Media, Inc. 2005.

NAKAMURA, E. T. Um modelo de Segurança de Redes para Ambientes Cooperativos. 2000. 286f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Computação, Universidade Estadual de Campinas, Campinas. 2000.

REKHTER, Y.; MOSKOWITZ R. G.; KARRENBERG D.; GROOT G. J. de; LEAR E. Address Allocation for *Private* Internets. Disponível em: < http://tools.ietf.org/html/rfc1918 >. Acesso em: 26 abril 2011. 22h17.

RHODES, T. Monitoring Third Party Security Issues. Disponível em http://www.freebsd.org/doc/handbook/security-portaudit.html. Acesso em: 12 maio 2011. 22h02.

SCOTT, C; WOLFE P; ERWIN M. Virtual *Private* Networks, Second Edition. 2 ed. O'Reilly & Associates, Inc. 1999.

SQUID. Disponível em : < http://www.squid-cache.org/ >. Acesso em: 09 maio 2011. 21h30.

STALLINGS, W. Criptografia e segurança de redes: Princípios e práticas. Tradução de Daniel Vieira. 4 ed. São Paulo: Pearson Prentice Hall, 2008. 492p.

VACCA, J. R. Public Key Infrastructure: Building Trusted Applications and Web Services. Auerbach Publications, 2004.