Práticas de Segurança para Administradores de Redes Internet

NIC BR Security Office
nbso@nic.br

Versão 1.1.1
24 de setembro de 2002

Resumo

Este documento é destinado a administradores de redes ligadas à Internet, incluindo provedores de acesso e de conteúdo. O seu propósito é ser um guia com informações para configurar, administrar e operar estas redes de forma mais segura.

Sumário

1 Introdução

Este documento procura reunir um conjunto de boas práticas em configuração, administração e operação segura de redes conectadas à Internet. A implantação destas práticas minimiza as chances de ocorrerem problemas de segurança e facilita a administração das redes e recursos de forma segura. É importante frisar que este conjunto representa o mínimo indispensável dentro de um grande universo de boas práticas de segurança, o que equivale a dizer que a sua adoção é um bom começo mas não necessariamente é suficiente em todas as situações.

As recomendações apresentadas são eminentemente práticas e, tanto quanto possível, independentes de plataforma de *software* e *hardware*. A maioria dos princípios aqui expostos é genérica; a sua efetiva aplicação requer que um administrador determine como estes princípios podem ser implementados nas plataformas que ele utiliza.

Este documento é dirigido ao pessoal técnico de redes conectadas à Internet, especialmente aos administradores de redes, sistemas e/ou segurança, que são os responsáveis pelo planejamento, implementação ou operação de redes e sistemas. Também podem se beneficiar da sua leitura gerentes com conhecimento técnico de redes.

1.1 Organização do Documento

O restante deste documento está organizado da seguinte maneira. A seção 2 apresenta políticas importantes para respaldar e viabilizar os procedimentos técnicos descritos nas seções subseqüentes. A seção 3 mostra como configurar sistemas e redes de forma mais segura. Na seção 4 são discutidos métodos para se ter segurança na administração e operação de redes e sistemas. O apêndice A traz sugestões de material de consulta para quem queira obter conhecimentos mais aprofundados sobre algum dos temas abordados nas seções de 2 a 4.

1.2 Como Obter este Documento

Este documento pode ser obtido em `http://www.nbso.nic.br/docs/seg-adm-redes.html`. Como ele é periodicamente atualizado, certifique-se de ter sempre a versão mais recente.

No mesmo endereço também está disponível um *checklist* que resume as principais práticas apresentadas neste documento, e que pode ser usado para o acompanhamento da sua implantação.

Caso você tenha alguma sugestão para este documento ou encontre algum erro nele, entre em contato através do endereço `doc@nic.br`.

1.3 Nota de *Copyright* e Distribuição

2. Se este documento for distribuído apenas em partes, instruções de como obtê-lo por completo devem ser incluídas.

3. É permitido o uso dos exemplos de documentos e de configuração incluídos neste texto. Tal uso é completamente livre e não está sujeito a nenhuma restrição.

4. É vedada a distribuição de versões modificadas deste documento, bem como a comercialização de cópias, sem a permissão expressa do NBSO.

Embora todos os cuidados tenham sido tomados na preparação deste documento, o NBSO não garante a correção absoluta das informações nele contidas, nem se responsabiliza por eventuais conseqüências que possam advir do seu uso.

2 Políticas

2.1 Políticas de Segurança

Uma política de segurança é um instrumento importante para proteger a sua organização contra ameaças à segurança da informação que a ela pertence ou que está sob sua responsabilidade. Uma ameaça à segurança é compreendida neste contexto como a quebra de uma ou mais de suas três propriedades fundamentais (confidencialidade, integridade e disponibilidade).

A política de segurança não define procedimentos específicos de manipulação e proteção da informação, mas atribui direitos e responsabilidades às pessoas (usuários, administradores de redes e sistemas, funcionários, gerentes, etc.) que lidam com essa informação. Desta forma, elas sabem quais as expectativas que podem ter e quais são as suas atribuições em relação à segurança dos recursos computacionais com os quais trabalham. Além disso, a política de segurança também estipula as penalidades às quais estão sujeitos aqueles que a descumprem.

Antes que a política de segurança seja escrita, é necessário definir a informação a ser protegida. Usualmente, isso é feito através de uma análise de riscos, que identifica:

- recursos protegidos pela política;
- ameaças às quais estes recursos estão sujeitos;
- vulnerabilidades que podem viabilizar a concretização destas ameaças, analisando-as individualmente.

Uma política de segurança deve cobrir os seguintes aspectos:

- aspectos preliminares:
  - abrangência e escopo de atuação da política;
  - definições fundamentais;
  - normas e regulamentos aos quais a política está subordinada;
  - quem tem autoridade para sancionar, implementar e fiscalizar o cumprimento da política;
  - meios de distribuição da política;
  - como e com que freqüência a política é revisada.
- política de senhas:
  - requisitos para formação de senhas;
  - período de validade das senhas;
  - normas para proteção de senhas;
  - reuso de senhas;
  - senhas *default*.
- direitos e responsabilidades dos usuários, tais como:
  - utilização de contas de acesso;
  - utilização de *softwares* e informações, incluindo questões de instalação, licenciamento e *copyright*;

  - proteção e uso de informações (sensíveis ou não), como senhas, dados de configuração de sistemas e dados confidenciais da organização;
  - uso aceitável de recursos como *email*, *news* e páginas Web;
  - direito à privacidade, e condições nas quais esse direito pode ser violado pelo provedor dos recursos (a organização);
  - uso de antivírus.
- direitos e responsabilidades do provedor dos recursos, como:
  - *backups*;
  - diretrizes para configuração e instalação de sistemas e equipamentos de rede;
  - autoridade para conceder e revogar autorizações de acesso, conectar e desconectar sistemas e equipamentos de rede, alocar e registrar endereços e nomes de sistemas e equipamentos;
  - monitoramento de sistemas e equipamentos de rede;
  - normas de segurança física.
- ações previstas em caso de violação da política:
  - diretrizes para tratamento e resposta de incidentes de segurança;
  - penalidades cabíveis.

Cabe ressaltar que a lista de tópicos acima não é exaustiva nem tampouco se aplica a todos os casos. Cada organização possui um ambiente distinto e os seus próprios requisitos de segurança, e deve, portanto, desenvolver uma política de segurança que se molde a essas peculiaridades.

Alguns fatores importantes para o sucesso de uma política de segurança são:

- apoio por parte da administração superior;
- a política deve ser ampla, cobrindo todos os aspectos que envolvem a segurança dos recursos computacionais e da informação sob responsabilidade da organização;
- a política deve ser periodicamente atualizada de forma a refletir as mudanças na organização;
- deve haver um indivíduo ou grupo responsável por verificar se a política está sendo respeitada;
- todos os usuários da organização devem tomar conhecimento da política e manifestar a sua concordância em submeter-se a ela antes de obter acesso aos recursos computacionais;
- a política deve estar disponível em um local de fácil acesso aos usuários, tal como a *intranet* da organização.

Dentre os itens acima, **o apoio por parte da administração superior é essencial**. Se a política de segurança não for encampada pela administração, ela rapidamente será deixada de lado pelos demais setores da organização. Além disso, é importante que os seus membros dêem o exemplo no que diz respeito à observância da política de segurança.

Os seguintes fatores influem negativamente na aceitação de uma política de segurança e podem levá-la ao fracasso:

- a política não deve ser demasiadamente detalhada ou restritiva;

- o excesso de detalhes na política pode causar confusão ou dificuldades na sua implementação;
- não devem ser abertas exceções para indivíduos ou grupos;
- a política não deve estar atrelada a *softwares* e/ou *hardwares* específicos.

2.2 Políticas de Uso Aceitável

A política de uso aceitável (AUP—*Acceptable Use Policy*) é o documento que define como os recursos computacionais da organização podem ser utilizados. Ela deve ser pública e estar disponível a todos os que utilizam a infra-estrutura computacional da organização, sendo recomendável que a autorização para uso dos recursos seja condicionada a uma concordância expressa com os seus termos.

A AUP é geralmente parte integrante da política de segurança global. Para muitas organizações, ela será composta pelos itens da política que afetam diretamente os usuários de recursos computacionais, principalmente os que definem seus direitos e responsabilidades.

Por outro lado, organizações que oferecem acesso a usuários externos (tais como provedores de acesso Internet) devem definir uma política de uso aceitável para esses usuários que seja independente da AUP à qual estão sujeitos os seus usuários internos. É importante que os usuários externos tomem conhecimento dessa política e saibam que o uso dos recursos está condicionado ao seu cumprimento.

3 Instalação e Configuração Segura de Sistemas

Uma vez estabelecidas as políticas de segurança apropriadas para a sua rede (conforme exposto na seção 2), a etapa seguinte deve ser a configuração segura dos sistemas que estarão nessa rede.

Caso não exista uma documentação atualizada que detalhe a configuração de alguns ou todos os sistemas em uso na sua rede, é aconselhável que estes sistemas sejam reinstalados observando-se as recomendações aqui expostas, ou, pelo menos, que a sua configuração seja revisada e a documentação correspondente atualizada.

IMPORTANTE: um sistema só deverá ser conectado à Internet após os passos descritos nas seções 3.1 a 3.8 terem sido seguidos. **A pressa em disponibilizar um sistema na Internet pode levar ao seu comprometimento**.

3.1 Preparação da Instalação

A instalação de um sistema deve ser feita com ele isolado do mundo externo. Para tanto, os seguintes princípios devem ser seguidos:

- planeje a instalação, definindo itens como:
  - o propósito do sistema a ser instalado;
  - os serviços que este sistema disponibilizará;
  - a configuração de *hardware* da máquina;
  - como o disco será particionado, etc.
- providencie de antemão todos os manuais e mídias de instalação que serão utilizados;
- instale o sistema a partir de dispositivos de armazenamento locais (CD, fita ou disco), desconectado da rede;
- caso você precise ligar o sistema em rede (para fazer *download* de atualizações, por exemplo), coloque-o em uma rede isolada, acessível apenas pela sua rede interna.

Caso seja possível, evite concentrar todos os serviços de rede em uma única máquina, dividindo-os entre vários sistemas. Isto é desejável pois aumenta a disponibilidade dos serviços na sua rede e reduz a extensão de um eventual comprometimento a partir de um deles.

3.2 Estratégias de Particionamento

Conforme mencionado na seção 3.1, um dos aspectos que devem ser incluídos no planejamento da instalação é como será feito o particionamento do(s) disco(s) do sistema. Embora isso dependa basicamente da utilização pretendida para o sistema, existem alguns fatores que devem ser levados em consideração no momento de decidir como o disco deve ser particionado.

Um princípio básico é dividir o disco em várias partições em vez de usar uma única partição ocupando o disco inteiro. Isto é recomendável por diversas razões:

- Um usuário ou um programa mal-comportado pode lotar uma partição na qual tenha permissão de escrita (áreas temporárias e de armazenamento de *logs* são suscetíveis a este problema). Se os programas do sistema estiverem em outra partição eles provavelmente não serão afetados, evitando que o sistema trave.
- Caso uma partição seja corrompida por alguma razão, as outras partições provavelmente não serão afetadas.
- Em alguns sistemas (notadamente sistemas UNIX), é possível definir algumas características individuais para cada partição. Por exemplo, algumas partições podem ser usadas em modo *read-only*, o que é útil para partições que contenham binários que são modificados com pouca freqüência.
- Em alguns casos a existência de várias partições permite múltiplas operações de disco em paralelo e/ou o uso de otimizações individuais para cada partição, o que pode aumentar significativamente o desempenho do sistema.
- O uso de várias partições geralmente facilita o procedimento de *backup* do sistema, pois simplifica funções como:
  - copiar partições inteiras de uma só vez;
  - excluir partições individuais do procedimento;
  - fazer *backups* a intervalos diferentes para cada partição.

As partições específicas que devem ser criadas variam de sistema para sistema, não existindo uma regra que possa ser sempre seguida. Entretanto, recomenda-se avaliar a conveniência da criação de partições separadas para as áreas onde são armazenados itens como:

- programas do sistema operacional;
- dados dos usuários;
- *logs*;
- arquivos temporários;
- filas de envio e recepção de *emails* (servidores SMTP);
- filas de impressão (servidores de impressão);
- repositórios de arquivos (servidores FTP);
- páginas Web (servidores HTTP).

Note que a lista acima não é exaustiva, podendo existir outras áreas do sistema que mereçam uma partição separada. Da mesma forma, existem itens dentre os acima que não se aplicam a determinados casos. Consulte a documentação do seu sistema operacional para ver se ela contém recomendações a respeito do particionamento adequado dos discos.

As partições devem ser dimensionadas de acordo com os requisitos de cada sistema. Em muitos casos, o tamanho ocupado pelo sistema operacional é fornecido na sua documentação, o que pode auxiliar na determinação do tamanho de algumas partições.

Qualquer que seja a estrutura de particionamento escolhida, é recomendável que você tenha pelo menos um esboço dela por escrito antes de começar a instalação. Isto agiliza o processo de instalação e reduz a probabilidade de que se faça uma determinada escolha sem que as suas conseqüências sejam adequadamente previstas.

3.3 Documentação da Instalação e Configuração

Uma medida importante para permitir uma rápida avaliação da situação de um sistema é a documentação de sua instalação e configuração. A idéia é ter uma espécie de *logbook* (ou "diário de bordo"), que detalhe os componentes instalados no sistema e todas as modificações na sua configuração global.

Esse *logbook* pode ser particularmente útil para determinar qual versão de determinado pacote está instalada no sistema ou para reconstituir uma dada instalação. Muitas vezes um administrador precisa consultar diversas fontes e realizar várias tentativas antes de instalar e/ou configurar corretamente um determinado pacote de *software*. A existência de um documento que relate quais os passos exatos que tiveram que ser seguidos para que a instalação/configuração fosse bem sucedida permite que esse mesmo pacote possa ser instalado com correção e rapidez em outro sistema ou ocasião. Conforme será visto na seção 4.2, a importância deste documento cresce na medida em que a responsabilidade pela administração dos sistemas seja compartilhada por diversas pessoas.

O formato e o grau de sofisticação do *logbook* dependem de diversos fatores, e cada administrador deve determinar qual a melhor maneira de manter essas informações. Um simples arquivo texto pode revelar-se extremamente eficaz, como mostram os exemplos da figura 1. O que realmente importa é que esse documento esteja disponível em caso de falha (acidental ou provocada) do sistema, e que ele contenha informações suficientes para que, a partir dele, seja possível reconstituir a exata configuração que o sistema possuía antes da falha, sem que seja necessário recorrer a *backups*.[1]

É essencial que alterações na configuração do sistema e de seus componentes estejam documentadas neste *logbook*. A entrada referente a estas alterações deve conter, no mínimo, os seguintes itens:

- data da modificação;
- responsável pela modificação;
- justificativa para a modificação;
- descrição da modificação.

Deve ser possível, ainda, reconstituir a situação antes da mudança na configuração a partir dessa entrada.

A figura 1 mostra um exemplo com algumas entradas do *logbook* de um servidor FTP. A primeira entrada registra a instalação inicial do sistema, realizada no dia 26/02 por um administrador chamado "Joe", e descreve ainda:

- o sistema operacional utilizado;
- como ele foi instalado;
- como o disco foi particionado;
- onde pode ser encontrada a lista de pacotes instalados;
- quais as portas que ficaram ativas após a instalação;
- quais os usuários criados (com seus respectivos UIDs e GIDs).

[1] A existência do *logbook* não diminui a importância dos *backups*, que serão tratados na seção 4.8.

```
Logbook para vangogh.example.org (IP 192.0.2.24)
================================================

26/Fev/2002      Responsável: Joe

   Instalação de vangogh.example.org, servidor FTP de example.org.  Instalado o
   sistema operacional GoodBSD versão 6.5. A instalação foi feita usando a opção
   'custom' do menu de instalação. O disco foi particionado da seguinte maneira:

   Filesystem   Size   Mount point  |   Filesystem   Size   Mount point
   /dev/wd0a    100M   /            |   /dev/wd0f    2.0G   /usr
   /dev/wd0d    3.0G   /var         |   /dev/wd0g    400M   /home
   /dev/wd0e    500M   /tmp         |   /dev/wd0h    4.0G   /home/ftp

   Uma lista dos pacotes instalados está em /usr/local/sysadm/pkg.inst.  As portas
   abertas após a instalação são (netstat -a):

  Active Internet connections (including servers)
  Proto Recv-Q Send-Q  Local Address           Foreign Address     (state)
  tcp        0      0  *.ftp                   *.*                 LISTEN
  tcp        0      0  *.ssh                   *.*                 LISTEN
  udp        0      0  vangogh.example.org.ntp *.*
  udp        0      0  localhost.ntp           *.*
  udp        0      0  *.ntp                   *.*
  udp        0      0  *.syslog                *.*

   Criados os usuários 'joe' (UID=501), 'alice' (UID=502), 'bob' (UID=503) e 'caio'
   (UID=504). Cada usuário pertence ao seu próprio grupo (GID=UID) e 'joe', 'alice' e
   'bob' pertencem também ao grupo 'wheel'.

--------------------------------------------------------------------------------
01/Mar/2002      Responsável: Alice

   Instalado o 'foo-1.2.3', uma ferramenta para análise de logs de FTP. Os fontes
   estão em /usr/local/src/foo-1.2.3. Os comandos necessários para a instalação foram:

   $ ./configure
   $ make
   # make install

   Para usar o programa, foi necessário criar o usuário 'foo' (UID=300,GID=100/users)
   e o diretório /usr/local/share/foo (owner=foo, group=users), com permissões 0755.

--------------------------------------------------------------------------------
03/Mar/2002      Responsável: Bob

   Criado o grupo 'foobar' (GID=300), para os usuários do pacote 'foo'.  O diretório
   /usr/local/share/foo teve seu grupo alterado de 'users' para 'foobar' e as
   permissões de 0755 para 0750. Modificação necessária para que apenas usuários
   pertencentes ao grupo 'foobar' tenham acesso aos programas do pacote 'foo'. Os
   usuários 'alice', 'bob' e 'caio' foram adicionados ao grupo 'foobar'.

--------------------------------------------------------------------------------
03/Mar/2002      Responsável: Alice

   Modificado o /etc/rc.local para carregar o daemon 'bard' (usado pelo pacote
   'foo') no boot. Um diff da modificação está em /usr/local/sysadm/rc.local-bard.diff.
```

Figura 1: Exemplos de entradas no *logbook*

Após a instalação inicial do sistema operacional, no dia 01/03 foi instalado um pacote chamado foo, versão 1.2.3. A entrada correspondente no *logbook* descreve os passos que foram seguidos para compilar e instalar o pacote e para preparar o sistema para o seu uso (criação de um usuário e um diretório, com suas respectivas informações).

A terceira entrada registra algumas alterações que tiveram que ser feitas na configuração do sistema para que o pacote foo pudesse ser usado corretamente. Por sua vez, a última entrada do exemplo apresenta uma modificação na inicialização do sistema para carregar um *daemon* (*software* servidor) usado pelo pacote. Observe que ambas as entradas permitem que a situação anterior do sistema (ou seja, a situação antes das modificações descritas) seja restaurada, caso isso se revele necessário ou desejável.

IMPORTANTE: o *logbook* de um sistema é um documento sensível, pois contém informações que podem ser usadas para comprometer mais facilmente a segurança deste sistema. Sendo assim, ele deve ser armazenado e manipulado com cuidado, de acordo com a política para documentos sensíveis da sua organização.

3.4 Senhas de Administrador

Durante a instalação de um sistema, em determinado momento será solicitado que você informe uma senha de administrador (*root* ou *Administrator*). Na maioria dos casos, é o próprio programa de instalação que solicita a escolha da senha. Em outros casos, a senha de administrador deve ser definida após o primeiro *boot* do sistema.

Procure estabelecer esta senha tão cedo quanto possível durante a instalação do sistema. De preferência, tenha uma senha já em mente quando começar a instalação.

Uma senha adequada é aquela fácil de ser lembrada e difícil de ser adivinhada. Ela deve respeitar, no mínimo, os seguintes critérios:

- ter um comprimento mínimo de 8 caracteres;
- ser formada por letras, números e caracteres especiais;
- não ser derivada de seus dados pessoais, tais como nomes de membros da família (incluindo animais de estimação), números de telefone, placas de carros, números de documentos e datas;
- não dever ser adivinhada por quem conheça suas preferências pessoais (time para o qual torce, escritor, ator ou cantor favorito, nomes de livros, filmes ou músicas, etc.);
- não estar presente em dicionários (de português ou de outros idiomas).

Uma sugestão para formar senhas que se encaixem nos requisitos acima é usar as primeiras ou últimas letras das palavras de uma frase, adicionando números e símbolos e trocando minúsculas e maiúsculas para dificultar ataques baseados em força bruta. Por exemplo, a partir das iniciais de “the book is on the table” obtém-se, inicialmente, “tbiott”. A partir daí, é possível trocar a letra “o” por um “0” (zero) e o penúltimo “t” por um símbolo “+”, colocar algumas letras em maiúsculo e acrescentar outras letras, chegando a “tBi0+TbL”, uma senha bastante difícil de ser adivinhada ou obtida por força bruta.[2]

[2] Evidentemente esta deixou de ser uma senha segura por constar neste documento.

3.5 Instalação Mínima

Um sistema mais seguro começa pela instalação do mínimo possível de pacotes e componentes, especialmente os que implementam serviços de rede. Este mínimo depende fundamentalmente do propósito do sistema em questão e do ambiente de rede no qual ele está inserido. Por exemplo, em princípio um sistema dedicado a servir páginas Web não precisa de um *software* servidor SMTP, assim como uma estação de trabalho não precisa de um servidor HTTP.

A justificativa para esta recomendação é bastante simples. É comum que serviços não utilizados não sejam monitorados por falhas de segurança, o que aumenta a possibilidade de não ser aplicada uma correção necessária. A redução no número de pacotes instalados diminui a chance de que o sistema possua uma vulnerabilidade que possa vir a ser explorada por um atacante.

Muitas vezes, administradores preferem instalar componentes cujo propósito ou funcionalidade desconheçam por receio de que alguma coisa deixe de funcionar no sistema. Entretanto, a maioria dos sistemas atuais possui algum mecanismo de controle de dependências que avisa quando determinado componente precisa de outro para funcionar. Em outras palavras, freqüentemente é possível deixar de instalar vários componentes *sem* comprometer a funcionalidade do sistema. Consulte a documentação do seu sistema ou o suporte técnico do seu fornecedor para saber se isto se aplica ao seu caso.

Alguns programas de instalação permitem que o administrador escolha entre uma instalação típica e uma personalizada ("para *experts*"). Quando possível, opte pela personalizada, evitando instalar componentes cuja funcionalidade seja desconhecida ou que você não esteja certo quanto à sua necessidade.

Em outros sistemas a instalação se dá em duas etapas, a instalação do sistema base (sobre a qual o administrador tem mínimo ou nenhum controle) e a instalação de pacotes ou componentes adicionais. Neste caso, instale o sistema base e selecione cuidadosamente quais os componentes extras que serão adicionados ao sistema. Neste tipo de sistema, a desativação de serviços não utilizados (seção 3.6) é muito importante e deve ser realizada com especial atenção.

3.6 Desativação de Serviços Não Utilizados

O passo seguinte a uma instalação mínima é a desativação de serviços (locais e, principalmente, de rede) que não serão imediatamente utilizados no sistema. A lógica por trás desta recomendação é a mesma por trás da instalação mínima de pacotes: reduzir a exposição do sistema a vulnerabilidades.

Embora possa parecer que exista redundância entre este passo e o anterior, na prática surgem situações nas quais o administrador é forçado a instalar um pacote ou componente completo para poder utilizar um subconjunto das funcionalidades oferecidas por esse pacote. Além disso, muitos programas de instalação de sistemas operacionais optam por maximizar a funcionalidade disponibilizada aos usuários, e a configuração padrão do sistema traz ativados todos os serviços que foram instalados. Caso uma dessas situações ocorra, as funcionalidades que não serão utilizadas deverão ser desativadas ou mesmo removidas do sistema.

Por exemplo, suponha que um pacote de serviços de impressão contenha tanto um cliente quanto um servidor de impressão remota. Se o sistema necessitar apenas do *software* cliente, o administrador deve desabilitar a parte referente ao *software* servidor neste sistema.

Caso não seja possível desativar serviços individualmente, uma alternativa é usar um filtro de pacotes para bloquear as portas TCP/UDP usadas por esses serviços, impedindo que eles sejam acessados através da rede. Isto será discutido em maiores detalhes na seção 4.11.

IMPORTANTE: a desativação de serviços e/ou a remoção de arquivos efetuadas nesta fase deverão ser documentadas no *logbook* do sistema.

3.7 Instalação de Correções

Depois de um sistema ter sido corretamente instalado e configurado, é necessário verificar se não existem correções (*patches*, *fixes*, *service packs*) para vulnerabilidades conhecidas nos componentes instalados. A maioria dos fornecedores de *software* libera correções para problemas de segurança que sejam descobertos em um sistema, sem que se tenha de esperar pela sua próxima versão. Na maioria das vezes, estas correções estão disponíveis através da Internet. Consulte seu fornecedor para saber como manter-se informado a respeito de correções para o seu sistema e de que forma elas podem ser obtidas.

Nem sempre todas as correções disponíveis precisam ser instaladas. Restrinja-se àquelas que corrigem problemas em componentes que estejam efetivamente instalados no seu sistema. Em caso de dúvida, recorra ao suporte técnico do seu fornecedor. A instalação indiscriminada de atualizações pode enfraquecer a segurança do sistema ao invés de fortalecê-la.

Registre no *logbook* a instalação de correções. Mantenha em sua rede um repositório das atualizações que já foram utilizadas, para facilitar a instalação das mesmas em outros sistemas.

IMPORTANTE: muitas vezes algumas configurações do sistema são alteradas durante o processo de instalação de correções. Sendo assim, é recomendável que você reveja a configuração do seu sistema após instalar uma correção para certificar-se de que a instalação não tenha revertido eventuais modificações que você tenha feito (especialmente aquelas destinadas a desativar serviços).

IMPORTANTE: a instalação de correções deve ser realizada não só como parte da instalação inicial do sistema, mas também durante o seu tempo de vida, a intervalos periódicos ou sempre que surgirem vulnerabilidades que o afetem. A seção 4.9 traz algumas recomendações sobre como manter-se informado a respeito de novas vulnerabilidades que afetem os seus sistemas.

3.8 Prevenção de Abuso de Recursos

Existem alguns serviços que, se mal configurados, podem permitir que usuários externos abusem dos recursos da sua rede, ainda que isso não implique na ocorrência de uma invasão. Dois destes serviços são o *email* e os *proxies* de Web.

A configuração incorreta destes serviços pode causar vários efeitos indesejáveis. Um deles é que recursos computacionais da organização—a começar pelo *link* Internet, mas incluindo CPU, discos e memória dos servidores—são consumidos por terceiros sem que eles paguem por esse uso. Em muitos casos, esses recursos são exauridos de forma que usuários legítimos não possam utilizar o serviço.

Além disso, servidores mal configurados são muitas vezes usados para disseminar conteúdo ilegal, tal como pornografia envolvendo crianças. Se um conteúdo deste tipo for encontrado em sistemas sob sua responsabilidade, existe a possibilidade de que você e/ou sua organização venham a ser legalmente implicados no caso.

3.8.1 Controle de *Relay* em Servidores SMTP

Na sua configuração padrão, muitos servidores SMTP vêm com o *relay* aberto, permitindo que eles sejam usados para enviar mensagens de e para qualquer rede ou domínio, independente dos endereços envolvidos serem da sua rede ou não. Estes servidores são amplamente explorados para envio de SPAM.

Além das conseqüências já mencionadas, diversas redes bloqueiam a recepção de mensagens a partir de servidores que tenham sido ou estejam sendo usados para envio de SPAM, fazendo com que usuários do servidor com *relay* aberto não possam enviar mensagens a usuários dessas redes. Há que se considerar também que o uso de servidores SMTP de terceiros torna mais difícil a localização e identificação dos *spammers*, diminuindo as chances de que eles sejam punidos por estes abusos.

Para resolver este problema de *relay* aberto você precisa configurar os seus servidores SMTP corretamente. A configuração adequada deve permitir apenas:

- envio de mensagens com endereço de origem local e endereço de destino local ou externo;
- recepção de mensagens com endereço de origem local ou externo e endereço de destino local.

Informações sobre como corrigir este problema para diferentes servidores SMTP estão disponíveis em http://www.mail-abuse.org/tsi/.

Na maioria dos casos, é possível fechar o *relay* mesmo quando a rede possui *roaming users*, usando mecanismos como POP-before-SMTP e SMTP AUTH. Caso a sua rede necessite suportar usuários deste tipo, consulte a documentação do seu servidor SMTP ou o seu fornecedor para saber como fechar o *relay* sem prejudicar a utilização do serviço por parte deles.

3.8.2 Controle de Acesso a *Proxies* Web

Assim como no caso dos servidores SMTP, *softwares* que fazem *proxy* de Web (tais como Squid, Wingate e Microsoft Proxy Server) também podem ser abusados se não forem tomadas as devidas precauções.

Um *proxy* mal configurado pode ser usado por usuários externos como um "trampolim" para acessar recursos de forma anônima. Esta anonimidade pode ser usada para cometer crimes, tais como envio de mensagens caluniosas, difamatórias ou ameaçadoras e divulgação de pornografia envolvendo crianças.

A configuração correta para um *proxy* Web é aquela que libera o acesso somente aos endereços IP de usuários autorizados (pertencentes à sua rede). Consulte a documentação do seu *software* ou o suporte técnico do seu fornecedor para obter maiores informações sobre como configurar o controle de acesso no seu *proxy*.

4 Administração e Operação Segura de Redes e Sistemas

4.1 Ajuste do Relógio

4.1.1 Sincronização de Relógios

Os relógios de todos os sistemas da sua rede (incluindo as estações de trabalho) deverão estar sincronizados, ou seja, deverão ter exatamente o mesmo horário. Para que isso aconteça, você deve usar um protocolo de sincronização de relógios, tal como o NTP (*Network Time Protocol*). Este protocolo é o mais recomendado, pois existem implementações dele para os mais variados sistemas operacionais, como pode ser visto em `http://www.ntp.org/`.

Para obter maior precisão no ajuste e para minimizar o tráfego desnecessário na rede, sugere-se que a sincronização via NTP seja implementada observando-se as seguintes recomendações:

1. Procure manter em sua rede um servidor NTP local. Esse servidor é quem irá realizar a sincronização com um servidor externo. As demais máquinas da sua rede, por sua vez, terão seus relógios sincronizados com o relógio do servidor local.
2. Muitos *backbones* disponibilizam um servidor NTP a seus clientes. Consulte o suporte técnico do seu *backbone* para verificar se ele oferece este serviço e como você pode fazer para utilizá-lo.

4.1.2 *Timezone*

Caso você trabalhe com servidores UNIX, ajuste o relógio de *hardware* destes sistemas para a hora padrão de Greenwich (GMT) e configure adequadamente o seu fuso horário (*timezone*) para que a hora local seja exibida corretamente.

O uso do *timezone* certo também possibilita o ajuste automatizado do relógio por conta do horário de verão. Para que isso seja possível, você deverá criar ou obter um arquivo de informações de *timezone* com as datas corretas de início e fim do horário de verão. Para maiores informações, consulte a documentação do comando `zic`.

4.2 Equipes de Administradores

Em muitas redes, a administração de sistemas é uma responsabilidade dividida entre várias pessoas. Nesses casos, é necessário estabelecer algumas regras para garantir a eficiência do trabalho em equipe.

4.2.1 Comunicação Eficiente

Em primeiro lugar, é essencial que os diferentes administradores comuniquem-se de maneira eficiente. Um bom modo de fazer isto é estabelecer listas de discussão por *email* que sejam internas à sua organização. Estas listas podem ser usadas para, entre outros propósitos, comunicar alterações na configuração dos sistemas, notificar os demais administradores a respeito de ocorrências relevantes e servir como mecanismo de acompanhamento da divisão de tarefas.

A grande vantagem de usar listas de discussão é que elas possibilitam a comunicação entre os administradores mesmo que alguns trabalhem em diferentes turnos ou locais. O histórico das listas pode servir para documentar decisões tomadas e para atualizar um administrador que tenha passado algum tempo afastado de suas atividades.

4.2.2 Controle de Alterações na Configuração

A partir do momento em que várias pessoas ficam encarregadas da administração de um sistema, torna-se necessário dispor de meios que possibilitem a identificação de quem foi o responsável por cada alteração na sua configuração. Isso permite resolver problemas de forma mais eficiente, pois a pessoa que realizou determinada modificação é a mais indicada para ajudar na resolução de eventuais complicações dela decorrentes.

Conforme mostrado na seção 3.3, o *logbook* pode auxiliar nessa tarefa. Para isso, é necessário que em cada entrada no *logbook* conste o nome da pessoa responsável pelas modificações ali descritas.

Uma forma mais automatizada de fazer isso é através do uso de ferramentas de controle de versão como o RCS (http://www.cs.purdue.edu/homes/trinkle/RCS/) e o CVS (http://www.cvshome.org). Essas ferramentas também permitem manter um histórico de arquivos de configuração, de forma a possibilitar a recuperação de antigas versões desses arquivos. Recomenda-se que, sempre que possível, este tipo de ferramenta seja usado em sistemas que possuam múltiplos administradores.

4.2.3 Uso de Contas Privilegiadas

Um problema que surge em sistemas com múltiplos administradores é a dificuldade de se manter um registro do uso de contas privilegiadas, tais como *root* e *Administrator*.

Sempre que possível, estas contas não devem ser usadas diretamente. Um administrador deve entrar no sistema usando sua conta pessoal e a partir dela realizar suas tarefas, usando os privilégios mais elevados apenas quando estritamente necessário. Em sistemas UNIX, isso é realizado através do comando `su`. O `su` traz como benefício adicional o fato de que o seu uso normalmente fica registrado nos *logs* do sistema, permitindo que se identifique quem acessou a conta de *root* em um determinado período.

O sudo (http://www.courtesan.com/sudo/) é uma ferramenta que permite que o administrador do sistema conceda a determinados usuários a possibilidade de executar comandos predefinidos como se eles fossem *root* (ou outro usuário), registrando nos *logs* do sistema a utilização desses comandos. O uso do sudo reduz a necessidade de compartilhamento da senha de *root*, uma vez que os usuários entram com sua própria senha para utilizar os comandos disponíveis através dele. Isso pode ser usado, por exemplo, quando existem contas de operador que são usadas para a realização de *backups* ou para invocar o procedimento de desligamento do sistema.

O sudo é extremamente configurável, possibilitando, entre outros recursos, a definição de grupos de usuários, de comandos e de *hosts* e o uso de restrições por *host* ou grupo de *hosts* (permitindo que o mesmo arquivo de configuração seja usado em sistemas diferentes).

IMPORTANTE: o uso de uma **conta administrativa única** com senha compartilhada, que não permita determinar qual dos administradores acessou o sistema, deve ser **evitado ao máximo**.

4.3 *Logs*

Logs são muito importantes para a administração segura de sistemas, pois registram informações sobre o seu funcionamento e sobre eventos por eles detectados. Muitas vezes, os *logs* são o único recurso que um administrador possui para descobrir as causas de um problema ou comportamento anômalo.

4.3.1 Geração de *Logs*

Para que os *logs* de um sistema sejam úteis para um administrador, eles devem estar com o horário sincronizado via NTP, ser tão detalhados quanto possível, sem no entanto gerar dados em excesso. Informações especialmente úteis são aquelas relacionadas a eventos de rede, tais como conexões externas e registros de utilização de serviços (arquivos transferidos via FTP, acessos a páginas Web, tentativas de *login* sem sucesso, avisos de disco cheio, etc.).

Para registrar estas informações, é necessário configurar o sistema da maneira apropriada. A forma de fazer isto geralmente varia para cada componente específico; consulte a documentação para descobrir como habilitar o *logging* de informações no seu sistema e em *softwares* específicos como *firewalls* e servidores HTTP.

4.3.2 Armazenamento de *Logs*

Armazenamento *on-line*

Os *logs* são tradicionalmente armazenados em disco, no próprio sistema onde são gerados. Essa é a opção mais óbvia, mas ela possui alguns riscos inerentes que devem ser compreendidos.

O primeiro deles diz respeito à possibilidade dos *logs* serem destruídos durante uma invasão do sistema (uma ocorrência bastante comum). Em alguns sistemas, isso pode ser contornado através da instalação de um *loghost* centralizado.

Um *loghost* centralizado é um sistema dedicado à coleta e ao armazenamento de *logs* de outros sistemas em uma rede, servindo como um repositório redundante de *logs*. Via de regra, o *loghost* não disponibiliza nenhum outro serviço, nem mesmo acesso remoto para os administradores, para minimizar a possibilidade de que ele seja comprometido. Outra vantagem de *loghosts* centralizados é que eles facilitam a análise dos *logs* e correlação de eventos ocorridos em sistemas distintos. **Sempre que possível, o uso de *loghosts* centralizados é fortemente recomendado**.

Um segundo risco é a possibilidade de um atacante usar o *logging* para executar um ataque de negação de serviço contra um determinado sistema, gerando eventos em excesso até que o disco onde são armazenados os *logs* fique cheio e o sistema trave em conseqüência disto. Conforme discutido na seção 3.2, o uso de uma partição separada para armazenar os *logs* pode minimizar o impacto deste problema.

Outro ponto que merece atenção é a rotação automática de *logs*. Quando este recurso é utilizado, deve-se garantir que os *logs* sejam movidos para o armazenamento *off-line* antes que eles sejam removidos do sistema pela rotação, evitando assim a perda de registros. Alguns sistemas trazem a rotação automática habilitada na sua configuração padrão; ao instalar um destes sistemas, verifique se esta configuração é compatível com os seus procedimentos de *backup* e armazenamento *off-line* de *logs*.

Armazenamento *off-line*

Evidentemente, os *logs* não podem ser mantidos *on-line* por tempo indeterminado, pois acabam por consumir muito espaço em disco. A melhor estratégia para resolver esta questão é transferir periodicamente os *logs* do disco para dispositivos de armazenamento *off-line*, tais como fita, CD-R ou CD-RW.

É recomendável gerar um *checksum* criptográfico (tal como MD5 ou SHA-1) dos *logs* que são armazenados *off-line*. Esse *checksum* deve ser mantido separado dos *logs*, para que possa ser usado para verificar a integridade destes caso eles venham a ser necessários.

Os *logs* armazenados *off-line* devem ser mantidos por um certo período de tempo, pois podem vir a ser necessários para ajudar na investigação de incidentes de segurança descobertos posteriormente. O Comitê Gestor da Internet no Brasil recomenda que *logs* de conexões de usuários de provedores de acesso estejam disponíveis por pelo menos 3 anos (vide http://www.cg.org.br/acoes/desenvolvimento.htm). É aconselhável que os demais *logs* sejam mantidos no mínimo por 6 meses.

É importante que os *logs* armazenados *on-line* sejam incluídos no procedimento de *backup* dos seus sistemas (*backups* são tratados na seção 4.8).

4.3.3 Monitoramento de *Logs*

Os *logs* possibilitam o acompanhamento do que acontece com a sua rede e com os seus sistemas. Para tanto, é importante que eles sejam monitorados com freqüência para permitir que eventuais problemas sejam rapidamente identificados.

Existem algumas práticas recomendáveis no que diz respeito ao monitoramento de *logs*:

- incorpore o hábito de inspecionar os *logs* à sua rotina de trabalho;
- faça isso pelo menos uma vez por dia, mas tenha em mente que sistemas muito importantes ou que geram muita informação podem precisar ter seus *logs* analisados com maior freqüência;
- procure investigar as causas de qualquer registro que lhe pareça incorreto ou anômalo, por mais insignificante que ele aparente ser;
- procure identificar o padrão de comportamento normal dos seus sistemas, para poder encontrar eventuais anomalias com maior rapidez.

Quando estiver analisando *logs*, você deve certificar-se do *timezone* usado para registrar o horário dos eventos. Por exemplo, alguns *softwares* (como o Microsoft IIS, dependendo da configuração adotada) registram eventos com a hora de Greenwich (GMT), e não com a hora local. O desconhecimento do *timezone* em que estão os *logs* pode facilmente invalidar uma análise e levá-lo a conclusões equivocadas.

À medida em que você venha a adquirir prática com a análise dos seus *logs*, você poderá escrever *scripts* ou pequenos programas para auxiliá-lo nesta tarefa, automatizando assim parte do processo. Estes *scripts* são úteis, por exemplo, para pré-processar os *logs* em busca de determinados conteúdos e para elaborar um resumo que pode ser enviado por *email* para o administrador do sistema.

Uma outra opção são ferramentas que permitem monitorar *logs* em tempo real, tais como o swatch (http://www.oit.ucsb.edu/~eta/swatch). O swatch requer que você especifique um conjunto de padrões a serem monitorados e as ações a serem tomadas quando um destes padrões é registrado nos *logs*. As ações podem ser de diversos tipos, como exibir a informação registrada, notificar um determinado usuário por *email* e invocar um programa do sistema. A capacidade de execução de comandos arbitrários do swatch torna-o muito atraente, pois permite, por exemplo, que sejam tomadas medidas como filtragem de um endereço IP que gere determinado *log* e envio de uma mensagem de alerta para um telefone celular.

4.4 DNS

O DNS (*Domain Name System*) é hoje um serviço essencial para o funcionamento da Internet. Essa importância, associada à natureza das informações que ele armazena, o tornam um dos alvos mais atraentes para atacantes. Desse modo, uma configuração adequada dos servidores DNS é crucial para aumentar a segurança e colaborar para o bom funcionamento da rede.

Servidores DNS expostos à Internet estão sujeitos a uma série de riscos, dentre os quais destacam-se:

- Vazamento de informações sensíveis sobre a rede da organização através de transferências de zonas DNS. Essas informações podem ajudar um atacante a identificar os pontos fracos da rede e a escolher futuros alvos.
- Ataques de envenenamento de *cache* (*cache poisoning*), que levam um servidor a armazenar informações forjadas. Tais informações podem ser usadas para comprometer a segurança de clientes que façam consultas a esse servidor.
- Comprometimento do servidor através de vulnerabilidades no *software* de DNS, o que pode facilitar outras quebras de segurança no restante da rede da organização.

Esta seção apresenta os principais mecanismos usados para eliminar ou minimizar estas ameaças, trazendo também recomendações sobre a configuração de DNS reverso. Informações mais detalhadas podem ser obtidas no documento *Securing an Internet Name Server*, do CERT/CC (disponível em `http://www.cert.org/archive/pdf/dns.pdf`) e nas referências do apêndice A.

4.4.1 Limitação de Transferências de Zona

Transferências de zona são usadas para que os servidores DNS escravos (secundários) atualizem suas informações sobre uma determinada zona DNS em relação ao servidor mestre (primário) para essa zona. Restringir os endereços que podem fazer transferências de zona é uma importante medida para evitar que atacantes obtenham informações detalhadas sobre a rede da organização, tais como endereços de roteadores, servidores de correio eletrônico e outros servidores DNS.

As limitações de transferências de zona devem ser aplicadas a todos os servidores com autoridade para um domínio, independente de eles serem mestres ou escravos. Um equívoco comum é limitar as transferências de zona no servidor mestre e não fazê-lo nos servidores escravos.

Preferencialmente, as transferências de zona devem ser limitadas através da configuração de controles de acesso no *software* servidor DNS. No BIND, por exemplo, isso é feito no `named.boot` (BIND 4) ou `named.conf` (BIND 8 e 9). Consulte a documentação do seu *software* ou o suporte técnico do seu fornecedor para obter informações sobre como limitar transferências de zona da maneira mais apropriada.

IMPORTANTE: uma concepção errônea, infelizmente bastante difundida, é a de que a limitação de transferências de zona deve ser feita filtrando o tráfego para a porta 53/TCP do servidor DNS. Como a porta 53/TCP também é usada na resolução de nomes, essa filtragem pode comprometer seriamente a funcionalidade do seu serviço de nomes.

4.4.2 Separação de Servidores

Servidores DNS possuem duas funções principais. A primeira delas é a de disponibilizar informações a respeito de zonas sobre as quais possuem autoridade (caso dos servidores mestres e escravos para uma determinada zona). A segunda função é a de resolver nomes para clientes na sua rede (neste caso, o servidor é dito recursivo). Muitas vezes, o mesmo servidor desempenha ambas funções.

Uma prática recomendável é separar a função de servidor com autoridade (mestre ou escravo) da função de servidor recursivo. Isso minimiza a eficácia de ataques de envenenamento de *cache* DNS. Na prática, essa separação significa que os servidores que possuem autoridade para uma ou mais zonas respondem somente a consultas relativas a essas zonas; por sua vez, os servidores recursivos não possuem autoridade sobre nenhuma zona DNS.

A forma mais simples de se fazer essa separação é configurar os servidores DNS com autoridade em máquinas distintas dos servidores DNS recursivos. Alguns *softwares* servidores DNS podem ser configurados para permitir que essa separação seja feita na mesma máquina; um exemplo é a versão 9 do BIND, que implementa isso através de visões (*views*).

4.4.3 Uso de Privilégios Mínimos

Os *softwares* servidores de DNS estão entre os alvos mais visados pelos atacantes, e já foram responsáveis por comprometimentos de segurança no passado. Dessa forma, uma medida recomendável é minimizar os privilégios com os quais o *software* servidor DNS é executado.

Em ambientes UNIX, muitas vezes é possível executar o servidor DNS em uma jaula `chroot()`. Versões mais recentes do BIND permitem também que ele seja executado com permissões de um usuário diferente de *root*. Consulte a documentação do seu *software* ou o suporte técnico do seu fornecedor para ver se uma dessas opções pode ser utilizada.

4.4.4 Cuidado com Informações Sensíveis no DNS

O DNS oferece alguns tipos de registros de recursos que armazenam informações adicionais sobre os nomes de domínio, tais como HINFO, TXT e WKS. Estes registros não são necessários para o funcionamento correto da resolução de nomes, sendo geralmente usados para facilitar a administração e a manutenção da rede.

Conforme é discutido em maiores detalhes na seção 4.10, informações sobre configuração de sistemas na sua rede devem ser consideradas sensíveis, pois podem ser usadas por um atacante para facilitar o comprometimento desses sistemas (ajudando-o a identificar máquinas com sistemas que possuam vulnerabilidades conhecidas, por exemplo). Em vista disso, o mais prudente é evitar registrar esse tipo de informação no DNS.

Caso você deseje usar estes tipos de registros para facilitar a administração da rede, recomenda-se fortemente que essas informações não estejam disponíveis para usuários externos à sua rede. Isso pode ser conseguido usando-se servidores DNS inacessíveis externamente ou, para o BIND 9, através do uso adequado de visões.

4.4.5 DNS Reverso

O uso mais freqüente do DNS é a tradução de nomes em endereços IP. Entretanto, ele também permite descobrir o nome associado a um determinado endereço IP. Isso é chamado DNS reverso, e possibilita a identificação do domínio de origem de um endereço IP.

Um DNS reverso mal configurado ou inexistente pode causar alguns transtornos. O primeiro deles é que muitos *sites* negam o acesso a usuários com endereços sem DNS reverso ou com o reverso incorreto.[3] Em segundo lugar, erros na configuração do DNS depõem contra a competência técnica da equipe de administração de redes responsável pelo domínio, e isso pode vir a causar dificuldades quando for necessário interagir com equipes de outras redes.

É recomendável que você mantenha atualizado o DNS reverso dos endereços sob sua responsabilidade. Em alguns casos a administração do DNS reverso dos seus blocos pode ser delegada à sua rede, enquanto em outros o seu provedor de *backbone* é quem é responsável pelo DNS reverso dos seus endereços. Entre em contato com o seu provedor de *backbone* para obter informações sobre como atualizar o seu DNS reverso.

4.5 Informações de Contato

Existem na Internet alguns endereços eletrônicos (*emails*) que são usados para entrar em contato com administradores quando se deseja comunicar determinadas ocorrências relacionadas à segurança de suas redes e sistemas. É extremamente importante que estas informações **sejam válidas** e estejam **sempre atualizadas**, pois assim garante-se que estas comunicações serão recebidas pela pessoa certa no menor espaço de tempo possível, o que pode muitas vezes impedir um incidente de segurança ou limitar as conseqüências. Esta seção mostra quais são essas informações e como você deve proceder para atualizá-las.

4.5.1 Endereços Eletrônicos Padrão

A RFC 2142[4] define uma série de *emails* padrão que devem existir em todas as redes e que podem ser usados para contatar os seus responsáveis. Dentre os endereços padrão, existem dois que estão relacionados com segurança: `abuse` e `security`.

O endereço `abuse` é usado para reportar abusos de recursos na rede. O endereço `security`, por sua vez, é utilizado para comunicar incidentes e alertar sobre problemas de segurança.

Mensagens enviadas para estes endereços deverão chegar até os responsáveis por lidar com essas ocorrências. Não é necessário criar usuários com estes nomes, basta que sejam configurados *aliases* redirecionando as mensagens enviadas a estes endereços para os usuários apropriados.

Cabe observar que muitas vezes estes endereços não são usados da maneira mais apropriada, com `abuse` recebendo reclamações de incidentes de segurança e `security` relatos de abusos, ou ambos os endereços sendo usados na mesma notificação. Sendo assim, é importante que sua rede possua ambos os endereços e que eles sejam constantemente monitorados pela sua equipe de segurança.

4.5.2 Contato no DNS

Cada domínio registrado em um servidor DNS possui uma série de parâmetros de configuração no registro de SOA (*Start of Authority*). Um destes parâmetros é o *email* do responsável pelo domínio, que muitas vezes também é usado para comunicar problemas de segurança envolvendo esse domínio.

[3] O caso mais comum de incorreção é quando existe um nome para resolver um dado IP mas este mesmo nome não está registrado em nenhum DNS direto, ou então resolve para outro endereço IP. Um exemplo disso seria o endereço IP 192.0.2.34 resolver para `foo.example.org` mas este nome resolver para o IP 192.0.2.76.

[4] D. Crocker, "Mailbox Names for Common Services, Roles and Functions", RFC 2142, Internet Mail Consortium, May 1997. Disponível em `ftp://ftp.isi.edu/in-notes/rfc2142.txt`.

Um exemplo de registro SOA para o domínio `example.org` pode ser visto na figura 2. Nesta figura, `ns.example.org` é o nome do servidor DNS primário e `joe.example.org` representa o endereço `joe@example.org`, que seria o endereço de contato para o domínio `example.org`.

```
example.org. IN SOA ns.example.org. joe.example.org. (
                    2002030101   ; serial (yyyymmddnn)
                    10800        ; refresh (3h)
                    3600         ; retry (1h)
                    6084800      ; expire (1 semana)
                    86400 )      ; TTL (1 dia)
```

Figura 2: Exemplo de registro SOA

Mantenha esse endereço do campo de SOA atualizado em todos os domínios sob sua responsabilidade, incluindo os de DNS reverso. Se preferir, use um *alias* em vez de um *email* real. Não se esqueça que o formato é ***usuário.domínio***, e não *usuário@domínio*.

4.5.3 Contatos no WHOIS

Cada domínio ou bloco de endereços IP registrado possui uma lista de informações de contato que remetem às pessoas responsáveis por estes domínios ou blocos. Geralmente existem três tipos de contatos:

- contato técnico: responsável técnico pela administração e operação do domínio ou bloco;
- contato administrativo: quem tem autoridade sobre o domínio ou bloco;
- contato de cobrança: quem recebe correspondências de cobrança das despesas de registro e manutenção do domínio ou bloco.

Os endereços de *email* destes contatos devem estar sempre atualizados e ser válidos. No caso do contato técnico, isso significa dizer que mensagens enviadas para este endereço devem ser recebidas por um administrador de redes responsável pelo bloco ou domínio, e não por pessoal administrativo ou jurídico da organização. Este contato é usado com muita freqüência para notificação de incidentes de segurança e outros problemas com a infra-estrutura de rede envolvendo o domínio ou bloco.

Estas informações de contato são mantidas em uma base de dados chamada WHOIS. Esta base de dados é normalmente gerenciada por entidades que registram domínios (informações sobre domínios) e por provedores de *backbone* (informações sobre endereços IP). No Brasil, estas informações são administradas e disponibilizadas pelo Registro .br (http://registro.br).

O procedimento de atualização dos contatos no WHOIS varia de acordo com a entidade de registro ou provedor de *backbone*. Entre em contato com a sua entidade de registro ou o seu provedor para obter informações detalhadas sobre como efetuar essa atualização. Para domínios registrados no Brasil, informações sobre como atualizar os contatos estão disponíveis em http://registro.br/faq/faq2.html.

4.6 Eliminação de Protocolos sem Criptografia

Uma medida de segurança muito importante na operação de redes é a substituição de protocolos onde não haja autenticação através de senhas, ou onde senhas trafeguem em claro, por outros que corrijam estas deficiências. A lista de protocolos cuja utilização deve ser evitada na medida do possível inclui:

- Telnet;
- FTP;
- POP3;
- IMAP;
- rlogin;
- rsh;
- rexec.

A maioria dos protocolos citados pode ser substituída pelo SSH.[5] Essa substituição, além de fazer com que o tráfego entre cliente e servidor passe a ser criptografado, traz ainda outras vantagens, como proteção da sessão contra ataques tipo *man-in-the-middle* e seqüestro de conexões TCP.

Em relação ao POP3, existem diversas possibilidades de substituição:

1. Usar uma das variantes do protocolo (APOP, KPOP, RPOP) que tornam a autenticação de usuários mais segura, pois eliminam o tráfego de senhas em claro. As desvantagens desta opção são que nem todos os clientes de *email* suportam estas variantes e o conteúdo dos *emails* (que pode conter informações sensíveis) não é criptografado.
2. Usar POP3 através de um túnel SSH ou SSL. A primeira opção é interessante quando o servidor POP3 e o servidor SSH residem na mesma máquina. Para a segunda, podem ser usadas ferramentas como o stunnel (`http://stunnel.mirt.net`). Alguns clientes de *email* já suportam SSL diretamente, não sendo necessário o uso de túneis.
3. Usar uma solução de Webmail sobre HTTPS (HTTP+SSL). Esta solução também é válida para o IMAP.

4.7 Cuidados com Redes Reservadas

Existem alguns blocos de endereços IP que são reservados pelo IANA (*Internet Assigned Numbers Authority*) para propósitos específicos. Não existe um documento único que registre todos estes blocos; alguns estão documentados em RFCs, enquanto que outros são considerados reservados por razões de compatibilidade histórica. A lista atual de redes reservadas conhecidas é mostrada na tabela 1, juntamente com um breve comentário sobre a finalidade cada rede.

Endereços pertencentes a estes blocos não devem ser propagados através da Internet, devendo ser filtrados no perímetro da sua rede, tanto para entrada quanto para saída.

[5] Implementações de SSH para diversos sistemas operacionais estão disponíveis em `http://www.openssh.com`.

Rede	Comentário
0.0.0.0/8	usada por sistemas antigos para *broadcast*
127.0.0.0/8	*loopback*
192.0.2.0/24	TEST-NET; usada para exemplos em documentação
10.0.0.0/8	usada em redes privadas (RFC 1918)
172.16.0.0/12	usada em redes privadas (RFC 1918)
192.168.0.0/16	usada em redes privadas (RFC 1918)
169.254.0.0/16	usada para autoconfiguração (está relacionada ao protocolo DHCP)
192.88.99.0/24	usada para 6to4 Relay Anycast (RFC 3068)
198.18.0.0/15	usada para testes de desempenho de equipamentos de rede (RFC 2544)
224.0.0.0/4	classe D
240.0.0.0/5	classe E

Tabela 1: Lista de redes reservadas pelo IANA

Caso você possua redes privadas com IPs reservados, certifique-se de que os endereços utilizados sejam os especificados na RFC 1918[6] (`10.0.0.0/8`, `172.16.0.0/12` e `192.168.0.0/16`).

Endereços reservados não devem estar associados a nomes em servidores DNS públicos. Se você utilizá-los em redes privadas e quiser usar nomes para as máquinas, configure um servidor DNS privado ou utilize tabelas de *hosts* (`/etc/hosts` ou `C:\WINDOWS\HOSTS`).

Caso você detecte um ataque proveniente de uma das redes da tabela 1 e estes endereços estiverem sendo filtrados no perímetro, os pacotes correspondentes só podem ter partido de dentro da sua própria rede. A causa mais freqüente para isso é a existência de erros de configuração que fazem com que os endereços reservados "vazem" de uma ou mais de suas redes privadas. Logo, deve-se procurar internamente a cusa do problema em vez de tentar contactar o IANA (que é a entidade listada nos contatos de WHOIS para estes blocos).

4.8 Políticas de *Backup* e Restauração de Sistemas

A importância dos *backups* na administração de sistemas nunca pode ser minimizada. Sem eles, muitos dados são simplesmente irrecuperáveis caso sejam perdidos devido a uma falha acidental ou a uma invasão.

Os *backups* devem fazer parte da rotina de operação dos seus sistemas e seguir uma política determinada. O melhor é fazê-los da forma mais automatizada possível, de modo a reduzir o seu impacto sobre o trabalho dos administradores e operadores de sistemas.

A lista de itens cujo *backup* deve ser feito com freqüência inclui:

- dados;
- arquivos de configuração;
- *logs*.

Um ponto que merece especial cuidado é o *backup* de binários (executáveis e bibliotecas), que geralmente deve ser evitado. Uma exceção a essa regra é uma cópia completa do sistema logo após a sua instalação,

[6] Y. Rekhter *et.al.*, "Address Allocation for Private Internets", RFC 1918, February 1996. Disponível em `ftp://ftp.isi.edu/in-notes/rfc1918.txt`.

antes que ele seja colocado em rede. *Backups* que incluem binários não são aconselháveis porque abrem a possibilidade de que eventuais Cavalos de Tróia ou executáveis corrompidos sejam reinstalados na restauração do sistema.

Alguns cuidados devem ser tomados em relação ao local onde são guardados os *backups*:

- o acesso ao local deve ser restrito, para evitar que pessoas não autorizadas roubem ou destruam *backups*;
- o local deve ser protegido contra agentes nocivos naturais (poeira, calor, umidade);
- se possível, é aconselhável que o local seja também à prova de fogo.

Os *backups* devem ser verificados logo após a sua geração e, posteriormente, a intervalos regulares. Isto possibilita a descoberta de defeitos em dispositivos e meios de armazenamento e pode evitar que dados sejam perdidos por problemas com *backups* que não podem ser restaurados.

Algumas organizações providenciam meios para armazenar *backups* fora das suas instalações, como em cofres de bancos, por exemplo. Essa é uma boa maneira de garantir a disponibilidade dos *backups* em caso de problemas nas instalações. Entretanto, isso pode comprometer a confidencialidade e integridade desses *backups*. Uma possível solução é criptografar o *backup* e gerar um *checksum* (MD5 ou SHA-1, por exemplo) dele antes que seja entregue a pessoas de fora da organização. Uma verificação do *checksum* antes da restauração pode servir como prova de que o *backup* não foi alterado desde que foi feito.

Quando for necessário restaurar um sistema, isto deve ser feito com a máquina isolada da rede. Caso o sistema em questão tenha sido comprometido, revise a sua configuração após a restauração para certificar-se de que não tenha ficado nenhuma porta de entrada previamente instalada pelo invasor.

4.9 Como Manter-se Informado

Administradores envolvidos com a segurança de redes e sistemas necessitam buscar informações de forma a se manterem atualizados em relação a novas vulnerabilidades e correções de segurança. Devido à sua natureza dinâmica, o principal meio de obtenção de tais informações é a própria Internet, através de listas de discussão por *email* e *sites* especializados.

Os tipos mais indicados de listas de discussão para um administrador incluem:

- lista de anúncios de segurança de fornecedores de *software* e *hardware* cujos produtos são usados na sua rede;
- listas de administradores e/ou usuários desses produtos;
- lista de alertas de segurança do CERT/CC.[7]

Destas, as listas de anúncios de segurança de fornecedores e a lista de alertas do CERT/CC são fortemente recomendadas a qualquer administrador. As listas destinadas a administradores e usuários de produtos, por sua vez, podem ajudá-lo a conhecer melhor as ferramentas disponíveis no seu ambiente computacional, muitas vezes levando-o a descobrir formas mais eficientes de trabalhar com elas.[8]

[7] Veja http://www.cert.org/contact_cert/certmaillist.html.

[8] A seção 4.10 mostra alguns cuidados que devem ser tomados por quem utiliza listas de discussão por *email*.

Existem outras listas que são indicadas para administradores que possuam alguma experiência e bons conhecimentos de programação. Essas listas costumam ter um alto tráfego e o conteúdo das suas discussões é bastante técnico, muitas vezes envolvendo o uso de conceitos avançados. A principal (e também a mais conhecida) destas listas é a Bugtraq.[9]

A Web também oferece boas fontes de informações atualizadas na área de segurança, tais como:

- http://www.cert.org/advisories/;
- http://www.cert.org/current/current_activity.html;
- http://online.securityfocus.com/;
- http://www.incidents.org/.

IMPORTANTE: é recomendável que você tome cuidado com a procedência de informações relacionadas com segurança, procurando se restringir a fontes confiáveis. Existem diversos relatos de informações propositalmente erradas terem sido divulgadas com o objetivo de abrir brechas na segurança da rede daqueles que as tenham seguido.

4.10 Precauções contra Engenharia Social

Engenharia social é a técnica (ou arte) de aproveitar-se da boa fé de pessoas para obter informações que possibilitem ou facilitem o acesso aos recursos computacionais de uma organização por parte de usuários não autorizados. Dentre as informações procuradas destacam-se as seguintes:

- senhas de acesso;
- topologia da rede;
- endereços IP em uso;
- nomes de *hosts* em uso;
- listas de usuários;
- tipos e versões de sistemas operacionais usados;
- tipos e versões de serviços de rede usados;
- dados sigilosos sobre produtos e processos da organização.

Existem diversas formas de se efetuar um ataque de engenharia social, mas todas elas têm em comum a característica de usarem basicamente psicologia e perspicácia para atingir os seus propósitos. Atualmente, as mais populares são:

- usar telefone ou *email* para se fazer passar por uma pessoa (geralmente alguém da equipe de suporte técnico ou um superior da pessoa atacada) que precisa de determinadas informações para resolver um suposto problema;

[9] Veja http://online.securityfocus.com/.

- aproveitar informações divulgadas em um fórum público da Internet (lista de discussão por *email*, *newsgroup*, IRC) por um administrador ou usuário que busca ajuda para resolver algum problema na rede;
- enviar programas maliciosos ou instruções especialmente preparadas para um administrador ou usuário, com o objetivo de abrir brechas na segurança da rede ou coletar o máximo de informações possível sobre ela (esta técnica é particularmente eficaz quando a pessoa pede auxílio em algum fórum de discussão pela Internet);
- navegar por *websites* ou repositórios FTP em busca de informações úteis—muitas vezes é possível encontrar descrições detalhadas da infra-estrutura computacional e/ou documentos que, por descuido ou esquecimento, não foram removidos do servidor.

A principal maneira de se prevenir contra estes ataques é orientando os usuários e administradores de redes e sistemas sobre como agir nestas situações. A política de segurança da organização (seção 2.1) desempenha um papel importante neste sentido, pois é nela que são definidas as normas para proteção da informação na organização.

Recomenda-se fortemente que os administradores tenham cuidado ao buscar ajuda em listas de discussão e outros fóruns na Internet. Estes recursos podem ser valiosos na resolução de problemas, mas também podem ser usados por terceiros para coleta de informações.

Procure reduzir a exposição da sua rede em fóruns públicos—por exemplo, use endereços IP, nomes de *hosts* e usuários hipotéticos, e tente não revelar mais sobre a topologia da rede do que o estritamente necessário para resolver um dado problema. Tome cuidado com orientações passadas por pessoas desconhecidas, e evite executar programas de origem obscura ou não confiável—eles podem ser uma armadilha.

4.11 Uso Eficaz de *Firewalls*

Antes de apresentar técnicas para aumentar a eficácia de *firewalls*, é importante definir o que um *firewall* é e o que ele não é. Um *firewall* bem configurado é um instrumento importante para implantar a política de segurança da sua rede. Ele pode reduzir a informação disponível externamente sobre a sua rede, ou, em alguns casos, até mesmo barrar ataques a vulnerabilidades ainda não divulgadas publicamente (e para as quais correções não estão disponíveis).

Por outro lado, *firewalls* não são infalíveis. **A simples instalação de um *firewall* não garante que sua rede esteja segura contra invasores**. Um *firewall* não pode ser a sua única linha de defesa; ele é mais um dentre os diversos mecanismos e procedimentos que aumentam a segurança de uma rede.

Outra limitação dos *firewalls* é que eles protegem apenas contra ataques externos ao *firewall*, nada podendo fazer contra ataques que partem de dentro da rede por ele protegida.

Esta seção apresenta apenas alguns aspectos importantes da utilização de *firewalls*. Maiores informações podem ser obtidas em `http://www.interhack.net/pubs/fwfaq/` e nas referências do apêndice A.

4.11.1 A Escolha de um *Firewall*

Existem diversas soluções de *firewall* disponíveis no mercado. A escolha de uma delas está atrelada a fatores como custo, recursos desejados e flexibilidade, mas um ponto essencial é a familiaridade com a plataforma operacional do *firewall*. A maioria dos *firewalls* está disponível para um conjunto reduzido de plataformas

operacionais, e a sua escolha deve se restringir a um dos produtos que roda sobre uma plataforma com a qual os administradores da rede tenham experiência. Por exemplo, se você utiliza basicamente servidores UNIX, é aconselhável que você escolha um *firewall* que rode sobre a sua variante favorita de UNIX, e não um produto que requeira Windows NT.

Existem, basicamente, duas razões para esta recomendação. A primeira delas é que você deve estar familiarizado o suficiente com o sistema onde o *firewall* será executado para configurá-lo de forma segura. A existência de um *firewall* instalado em um sistema inseguro pode ser até mais perigosa do que a ausência do *firewall* na rede. A segunda razão é que os produtos tendem a seguir a filosofia da plataforma onde rodam; por exemplo, a maioria dos *firewalls* para Windows é configurada através de menus e janelas, ao passo que muitos *firewalls* para UNIX são configurados por meio de arquivos texto.

Administradores experientes em UNIX têm à disposição diversas ferramentas de *software* livre que podem ser usadas para implementar *firewalls*, conforme mostra a tabela 2. Estas ferramentas permitem construir *firewalls* eficientes a um custo relativamente baixo, uma vez que seus requisitos de *hardware* são modestos.

Ferramenta	**Plataforma**	**Característica**
ipchains	Linux	filtro de pacotes (*stateless*)
iptables	Linux	filtro de pacotes (*stateful*)
ipfw	FreeBSD	filtro de pacotes (*stateful*)
pf	OpenBSD	filtro de pacotes (*stateful*)
ipfilter	vários UNIX	filtro de pacotes (*stateful*)
TIS Firewall Toolkit	vários UNIX	*proxy* para vários protocolos
Squid	vários UNIX	*proxy* Web/FTP

Tabela 2: Ferramentas de *software* livre para a construção de *firewalls*

4.11.2 Localização dos *Firewalls*

A localização dos *firewalls* na rede depende normalmente da sua política de segurança. Entretanto, existem algumas regras que se aplicam à grande maioria dos casos:

- **Todo o tráfego deve passar pelo *firewall***. Um *firewall* só pode atuar sobre o tráfego que passa por ele. A eficácia de um *firewall* pode ser severamente comprometida se existirem rotas alternativas para dentro da rede (modens, por exemplo). Caso não seja possível eliminar todas esses caminhos, eles devem ser documentados e fortemente vigiados através de outros mecanismos de segurança.

- **Tenha um filtro de pacotes no perímetro da sua rede**. Esse filtro pode estar localizado entre o seu roteador de borda e o interior da rede ou no próprio roteador, se ele tiver esta capacidade e você se sentir confortável utilizando-o para esta tarefa. O filtro de pacotes de borda é importante para tarefas como bloqueio global de alguns tipos de tráfego (vide seção 4.11.3) e bloqueio rápido de serviços durante a implantação de correções após a descoberta de uma nova vulnerabilidade.

- **Coloque os servidores externos em uma DMZ**. É recomendável que você coloque os seus servidores acessíveis externamente (Web, FTP, correio eletrônico, etc.) em um segmento de rede separado e com acesso altamente restrito, conhecido como DMZ (*DeMilitarized Zone*, ou zona desmilitarizada). A principal importância disso é proteger a rede interna contra ataques provenientes dos servidores externos—uma precaução contra a eventualidade de que um destes servidores seja comprometido. Por exemplo,

suponha que um atacante invada o servidor Web e instale um *sniffer* na rede. Se este servidor Web estiver na rede interna, a probabilidade dele conseguir capturar dados importantes (tais como senhas ou informações confidenciais) é muito maior do que se ele estiver em uma rede isolada.

- **Considere o uso de *firewalls* internos**. Em alguns casos, é possível identificar na rede interna grupos de sistemas que desempenham determinadas tarefas comuns, tais como desenvolvimento de *software*, *webdesign* e administração financeira. Nestes casos, recomenda-se o uso de *firewalls* internos para isolar estas sub-redes umas das outras, com o propósito de aumentar a proteção dos sistemas internos e conter a propagação de ataques bem-sucedidos.

4.11.3 Critérios de Filtragem

Existem basicamente dois critérios de filtragem que podem ser empregados em *firewalls*. O primeiro é o de *default deny*, ou seja, todo o tráfego que não for explicitamente permitido é bloqueado. O segundo, *default allow*, é o contrário, ou seja, todo o tráfego que não for explicitamente proibido é liberado.

A configuração dos *firewalls* deve seguir a política de segurança da rede. Se a política permitir, é recomendável adotar uma postura de *default deny*. Esta abordagem é, geralmente, mais segura, pois requer uma intervenção explícita do administrador para liberar o tráfego desejado, o que minimiza o impacto de eventuais erros de configuração na segurança da rede. Além disso, ela tende a simplificar a configuração dos *firewalls*.

No perímetro da rede, pelo menos as seguintes categorias de tráfego devem ser filtradas:

- tráfego de entrada (*ingress filtering*): pacotes com endereço de origem pertencente a uma rede reservada (seção 4.7) ou a um dos blocos de endereços da sua rede interna;
- tráfego de saída (*egress filtering*): pacotes com endereço de origem pertencente a uma rede reservada ou que não faça parte de um dos blocos de endereços da rede interna.

Um aspecto que deve ser considerado com cuidado é a filtragem do protocolo ICMP. O bloqueio indiscriminado de ICMP pode prejudicar o funcionamento da rede. Por outro lado, o ICMP pode ser usado para revelar a um possível atacante informações sobre a rede e seus sistemas. Observe que muitos *firewalls* do tipo *stateful* permitem a passagem de mensagens ICMP de erro associadas a conexões estabelecidas, o que minimiza o impacto da filtragem.

O tráfego para a DMZ deve ser altamente controlado. As únicas conexões permitidas para os sistemas dentro da DMZ devem ser as relativas aos serviços públicos (acessíveis externamente). Conexões partindo da DMZ para a rede interna devem ser, na sua maioria, tratadas como conexões oriundas da rede externa, aplicando-se a política de filtragem correspondente.

IMPORTANTE: a DMZ e a rede interna não podem estar no mesmo segmento de rede (ligadas ao mesmo *hub* ou *switch*, por exemplo). **É imprescindível que estas redes estejam em segmentos de rede separados**.

4.11.4 Exemplos

Diversas arquiteturas podem ser empregadas para a implantação de *firewalls* em uma rede. A opção por uma delas obedece a uma série de fatores, incluindo a estrutura lógica da rede a ser protegida, custo, funcionalidades pretendidas e requisitos tecnológicos dos *firewalls*.

Esta seção apresenta duas destas arquiteturas. A intenção não é cobrir todas as possibilidades de uso de *firewalls*, mas fornecer exemplos de arquiteturas que funcionam e que podem eventualmente ser adotados (na sua forma original ou após passarem por adaptações) em situações reais.

A figura 3 mostra um exemplo simples de uso de *firewall*. Neste exemplo, o *firewall* possui três interfaces de rede: uma para a rede externa, uma para a rede interna e outra para a DMZ. Por *default*, este *firewall* bloqueia tudo o que não for explicitamente permitido (*default deny*). Além disso, o *firewall* usado é do tipo *stateful*, que gera dinamicamente regras que permitam a entrada de respostas das conexões iniciadas na rede interna; portanto, não é preciso incluir na configuração do *firewall* regras de entrada para estas respostas.

O tráfego liberado no exemplo da figura 3 é o seguinte:

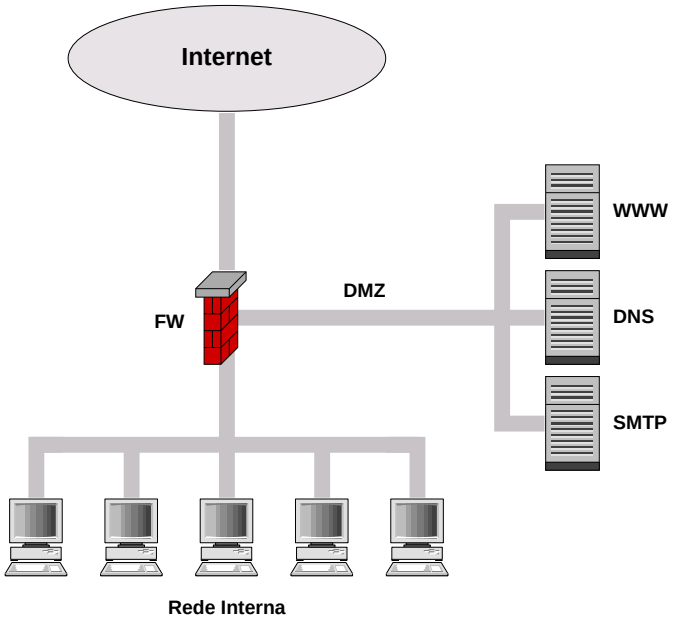

Figura 3: Um exemplo simples de *firewall*

- interface externa:
  - saída: tudo com exceção de
    - pacotes com endereços de origem pertencentes a redes reservadas;
    - pacotes com endereços de origem não pertencentes aos blocos da rede interna.
  - entrada: apenas os pacotes que obedecem às seguintes combinações de protocolo, endereço e porta de destino:
    - 25/TCP para o servidor SMTP;
    - 53/TCP e 53/UDP para o servidor DNS;
    - 80/TCP para o servidor WWW.
- interface interna:
  - saída: tudo;
  - entrada: nada;
- interface da DMZ:

– saída: portas 25/TCP (SMTP), 53/UDP e 53/TCP (DNS) e 113 (IDENT);
– entrada: além das mesmas regras de entrada da interface externa, também é permitido o tráfego para todos os servidores na com porta de destino 22/TCP (SSH) e endereço de origem na rede interna.

A figura 4 ilustra o uso de *firewalls* em uma situação mais complexa do que a anterior. Neste segundo exemplo, além dos servidores externos na DMZ, há também servidores na *intranet* e no setor financeiro da organização. Devido à importância das informações mantidas neste setor, a sua rede conta com a proteção adicional de um *firewall* interno, cujo objetivo principal é evitar que usuários com acesso à rede interna da organização (mas não à rede do setor financeiro) possam comprometer a integridade e/ou o sigilo dessas informações.

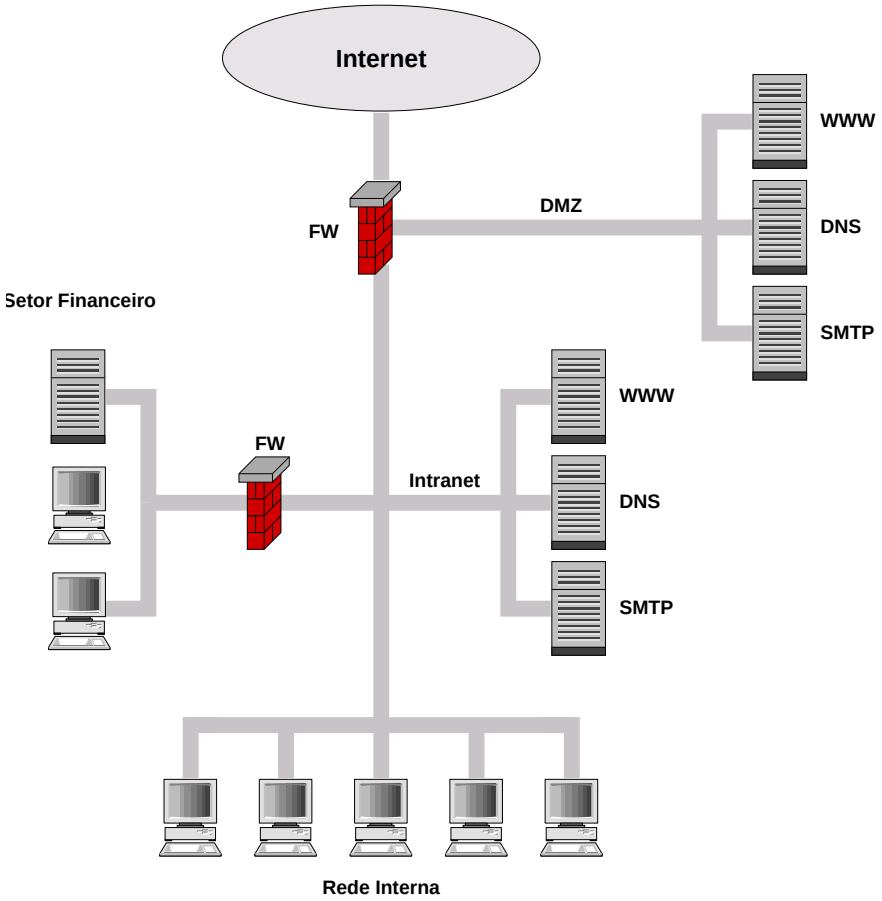

Figura 4: Um exemplo mais complexo de *firewall*

A configuração do *firewall* externo neste segundo exemplo é quase idêntica ao primeiro. Entretanto, no presente caso supõe-se que o servidor SMTP visível externamente (o da DMZ) repassa as mensagens recebidas para o servidor SMTP da *intranet*. Para que isso seja possível, é necessário mudar a regra de filtragem para a interface interna, liberando o tráfego do servidor SMTP da DMZ para a porta 25/TCP do servidor SMTP da *intranet*.

A configuração do *firewall* interno, por sua vez, é bastante simples. O servidor da rede do setor financeiro permite apenas acesso via HTTPS para que os funcionários da organização possam consultar seus contracheques; outros tipos de acesso não são permitidos. O tráfego liberado por este *firewall* é o seguinte:

- interface externa (rede interna):

  - saída: tudo;
  - entrada: apenas pacotes para o servidor do setor financeiro com porta de destino 443/TCP (HTTPS) e endereço de origem na rede interna;

- interface interna (rede do setor financeiro):
  - saída: tudo;
  - entrada: tudo (a filtragem é feita na interface externa).

A Referências Adicionais

A.1 URLs de Interesse

- "CERT Security Improvement Modules: Security Knowledge in Practice". http://www.cert.org/security-improvement/skip.html.

  Apresenta, de forma gráfica, os passos que estão envolvidos na obtenção de uma rede mais segura. Contém uma grande quantidade de material que aborda de forma mais aprofundada vários dos assuntos discutidos neste documento.

- "Security Links". http://www.nbso.nic.br/links/.

  Uma compilação de URLs sobre diversos aspectos de administração e segurança de redes e sistemas, incluindo diversos apresentados neste documento, e que é atualizada periodicamente.

- "Práticas de Segurança para Administradores de Redes Internet". http://www.nbso.nic.br/docs/seg-adm-redes.html.

  Página a partir da qual pode ser obtida a versão mais recente deste documento e do *checklist* que o acompanha. Contém também um histórico de revisões dos documentos.

A.2 Livros

- Nelson Murilo de O. Rufino. *Segurança Nacional*. Novatec, 2001.

  Uma boa referência sobre segurança computacional em português, com enfoque em aspectos práticos.

- W. Richard Stevens. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison-Wesley, 1994.

  A melhor obra disponível sobre TCP/IP. O texto é claro e didático, e numerosos exemplos (usando tcpdump) ajudam a compreender o funcionamento dos protocolos na prática.

- Simson Garfinkel e Gene Spafford. *Practical UNIX and Internet Security, 2nd Edition*. O'Reilly & Associates, 1996.

  Apesar de um pouco desatualizado em alguns aspectos, este livro é considerado referência obrigatória em segurança de sistemas UNIX.

- Paul Albitz e Cricket Liu. *DNS and BIND, 4th Edition*. O'Reilly & Associates, 2001.

  Este livro possui bastante informação sobre o protocolo DNS e a sua principal implementação, o BIND. A quarta edição contém um capítulo sobre segurança de servidores DNS.

- Elizabeth D. Zwicky, Simon Cooper e D. Brent Chapman. *Building Internet Firewalls, 2nd Edition*. O'Reilly & Associates, 2000.

  Um dos melhores livros disponíveis sobre *firewalls*, recheado com informações práticas sobre como construí-los.

- Evi Nemeth, Garth Snyder, Scott Seebass e Trent R. Hein. *UNIX System Administration Handbook, 3rd Edition*. Prentice Hall, 2001.

  O clássico sobre administração de sistemas UNIX, recentemente atualizado. Traz explicações claras e objetivas sobre como realizar, de forma eficiente, as diferentes tarefas que competem a um administrador de sistemas UNIX.

- Charles B. Rutstein. *Windows NT Security: A Practical Guide to Securing Windows NT Servers & Workstations*. McGraw-Hill, 1997.

  Um bom livro sobre segurança de Windows NT, incluindo instalação, configuração, uso do Registry, *logging*, entre outros assuntos.

- Roberta Bragg. *Windows 2000 Security*. New Riders Publishing, 2000.

  Este livro discute segurança no Windows 2000, dando maior ênfase aos aspectos práticos. Os temas abordados incluem IPsec, Kerberos, Active Directory, RAS e RRAS.

- Scott Barman. *Writing Information Security Policies*. New Riders Publishing, 2001.

  Este livro explica como escrever e implementar uma política de segurança. Contém vários exemplos extraídos de políticas reais, que podem ser usados como guia para a formulação de novas políticas.

- Série O'Reilly sobre administração de serviços de rede e sistemas operacionais específicos. http://www.oreilly.com.

  A editora O'Reilly é conhecida pela qualidade técnica dos seus livros, que geralmente abordam um assunto específico com bastante profundidade e com um enfoque bem prático. Existem guias para servidores como Apache (Web) e Sendmail (SMTP), além de diversos títulos sobre uso e administração de sistemas operacionais (incluindo UNIX, Linux e Windows).

Índice Remissivo