SEGURANÇA COMPUTACIONAL:

Segurança em Servidores Linux em Camadas

CARLOS EDUARDO SILVA DUMONT

LAVRAS

MINAS GERAIS - BRASIL

2006

CARLOS EDUARDO SILVA DUMONT

SEGURANÇA COMPUTACIONAL:

Segurança em Servidores Linux em Camadas

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras, como parte das exigências do Curso de Pós-Graduação *Lato Sensu* em Administração em Redes Linux, para a obtenção do título de Especialista em Administração em Redes Linux.

Orientador

Prof. Joaquim Quinteiro Uchôa.

LAVRAS
MINAS GERAIS - BRASIL
2006

CARLOS EDUARDO SILVA DUMONT

SEGURANÇA COMPUTACIONAL:

Segurança em Servidores Linux em Camadas

Monografia de Pós-Graduação apresentada ao Departamento de Ciências da Computação da Universidade Federal de Lavras, como parte das exigências da disciplina Trabalho de Conclusão de Curso para obtenção do título de especialista em Administração de Redes Linux aprovada pela seguinte banca examinadora:

Aprovada em _____ de _____ de _____

Prof. _____

Prof. _____

_____

Prof. Joaquim Quinteiro Uchôa

(Orientador)

LAVRAS
MINAS GERAIS - BRASIL

aos meus pais, Geraldo e Lucinea, pelo carinho e apoio em toda a minha a vida, à minha namorada, Gizelle, pelo amor e compreensão, a minha família, pela união, e aos meus colegas e amigos de classe, pelo companheirismo e amizade.

AGRADECIMENTOS

Agradeço a DEUS que me fez capaz de aprender e à Universidade Federal de Lavras pela oportunidade de aprendizado.

Agradeço imensamente a todos os professores e aos orientadores, os quais me transmitiram muito conhecimento ao longo do curso.

Agradeço aos meus pais e a minha namorada que me deram a estrutura e o amor necessário para crescer.

RESUMO

A proposta do presente trabalho é fazer uma reflexão sobre os perigos a que uma rede está sujeita e ações que podem ser executadas para prover segurança em quatro camadas em servidores Linux, através da utilização de várias técnicas e tecnologias, dando maior ênfase aos aspectos tecnológicos que envolvem a segurança de redes em ambientes cooperativos.

Após entender os riscos a que uma rede está sujeita e conhecer técnicas e tecnologias que podem ser empregadas para prover segurança, além de, entender como estas tecnologias podem atuar em diversas camadas, o administrador de rede terá uma visão mais ampla sobre as necessidades de segurança de uma empresa, facilitando assim o planejamento e implementação de uma política de segurança.

PALAVRAS CHAVES: Redes; Segurança; Linux;

SUMÁRIO

LISTA DE FIGURAS

CAPÍTULO 1 - INTRODUÇÃO

A segurança da informação tornou-se um fator prioritário na tomada de decisões e nos investimentos das empresas, tornando-se parte do negócio. Grande parte das empresas tem orçamento específico para TI[1] e para área de segurança.

O tema escolhido para esta monografia foi **Segurança Computacional: Segurança em Servidores Linux em Camadas,** devido a importância da segurança no cenário atual de TI.

O foco principal deste projeto é realizar um estudo e trazer informações sobre técnicas e tecnologias atuais empregadas, visando a segurança de redes em ambientes cooperativos com servidores Linux, apresentando estas tecnologias em quatro camadas de segurança.

Após entender os riscos a que uma rede está sujeita e conhecer técnicas e tecnologias que podem ser empregadas para prover segurança, além de, entender como estas tecnologias atuam em diversas camadas, o administrador terá uma visão mais clara sobre as necessidades de segurança de uma determinada empresa, facilitando assim o planejamento e implementação de uma política de segurança.

Diversos especialistas acreditam que, com a rápida evolução tecnológica, nenhum ambiente é totalmente seguro, e que, para minimizar os riscos de ataques, diversas tecnologias devem ser empregadas de acordo com a necessidade de cada empresa.

---

[1] TI: Tecnologia da Informação. Aplicação de diferentes ramos da tecnologia no processamento de informações.

Para maiores esclarecimentos sobre o tema escolhido, a seguir estão descritos alguns conceitos básicos que darão uma visão mais ampla, ajudando, assim, na compreensão desta monografia.

A ***Segurança em redes*** trata-se de um conjunto de técnicas e ferramentas utilizadas para proteger um sistema distribuído. A segurança da informação em uma rede de computadores envolve aspectos humanos, tecnológicos, processuais, jurídicos e de negócio.

Os ***Aspectos Tecnológicos*** são parte constituinte da segurança da informação que envolve recursos tecnológicos físicos e lógicos, isto é, *hardware*[2] e *software*[3].

O ***Ambiente Cooperativo*** é um ambiente empresarial heterogêneo que envolve matrizes, filiais, clientes, fornecedores, parceiros comerciais e usuários. Este ambiente é caracterizado pela integração dos mais diversos sistemas de diferentes organizações, nos quais as partes envolvidas cooperam entre si na busca de um objetivo comum.

Com o dinamismo da tecnologia, a cada dia surgem novas formas de conectividade e comunicação entre os diversos elementos de um ambiente cooperativo. Neles são utilizados diversos protocolos, sistemas operacionais, ferramentas, aplicativos, estruturas físicas, entre outros. Porém, o surgimento de uma nova tecnologia pode representar uma nova possibilidade de ataque contra a organização. A pergunta chave que orientou esta pesquisa foi: *Quais técnicas e tecnologias utilizar para garantir um nível eficaz de proteção em um ambiente cooperativo com servidores Linux?*

[2] *Hardware*: Definição para a parte física, ou seja, o próprio computador e seus periféricos.
[3] *Software*: Parte lógica em um sistema de computação.

Implantar segurança em um ambiente cooperativo com servidores Linux é uma tarefa complexa, pois no mesmo ocorrem muitas interações com o meio interno e externo. Para que a implantação da segurança seja eficaz são necessários profissionais qualificados, apoio dos executivos da organização e a utilização de uma série de técnicas e ferramentas tecnológicas.

Entre as principais técnicas e ferramentas estão a: adoção de uma boa política de segurança, utilização de um *firewall*, implantação de sistemas de detecção e prevenção de intrusos, criptografia[4], autenticação e configuração correta do sistema. Também são incluídos nesta lista, o monitoramento e verificação de integridade do sistema, definição de políticas de contingência, *backup*, além de diversos outros elementos, possibilitando assim a proteção da informação em diversos níveis. Empregando estes e outros recursos da maneira correta, a empresa atingirá um alto nível de segurança.

Este projeto se justifica pela necessidade e complexidade dos sistemas de segurança da informação nas empresas, que passaram a ser vistos como parte essencial do negócio. Além disso, a falta de informação e conhecimento sobre as ameaças existentes nas redes também constitui uma ameaça às organizações. Com a disseminação do conhecimento sobre ameaças, formas de defesa e a realização de pesquisas, as redes poderão se tornar mais seguras. Assim, a sociedade ganhará, pois terá à sua disposição serviços mais confiáveis e seguros.

O objetivo geral desta pesquisa é ampliar a base de conhecimento das empresas sobre segurança em servidores Linux, incentivando a realização de novas pesquisas, permitindo que, através da disseminação do conhecimento adquirido nestas, as empresas se tornem mais seguras, levando a sociedade a utilizar os recursos facilitadores do dia a dia de uma maneira mais confiável.

[4] Criptografia: Sistema utilizado para cifrar, tornar ilegíveis dados ou programas.

O objetivo específico da pesquisa é apresentar uma visão ampla sobre a importância da segurança e como implantá-la, além de apresentar os diversos tipos de ameaças às quais uma organização está sujeita. A pesquisa também irá discriminar as diversas técnicas e tecnologias que podem ser utilizadas em ambientes cooperativos com servidores Linux para promover a segurança da informação em diversas camadas de proteção.

A metodologia utilizada nesta monografia foi baseada em pesquisa bibliográfica tomando como base a leitura de livros reconhecidos da área de segurança em redes e acesso a *sites* reconhecidos da área de segurança, que foram selecionados com base em pesquisas realizadas através da Internet e recomendações de profissionais da área de segurança. Inicialmente foram selecionados vários livros e *sites* e foi realizada uma análise do conteúdo pertinente ao assunto. A partir daí, foram selecionados os livros e *sites* com conteúdo mais adequado para, assim, se iniciar o desenvolvimento da monografia.

A proposta do presente trabalho é fazer uma reflexão sobre o tema Segurança Computacional. O Capítulo 2 apresenta alguns conceitos básicos sobre segurança, possibilitando que o leitor tenha uma visão mais abrangente sobre o assunto. O Capítulo 3 esclarece porque foi adotada a abordagem de divisão da segurança em quatro camadas.

O Capítulo 4 trata da primeira camada, que irá representar o controle de acesso ao sistema, incluindo a definição da localização do servidor, autenticação e filtro de pacotes. O Capítulo 5 trata da segunda camada, que irá representar a segurança interna do sistema através da configuração correta do sistema operacional e dos serviços, definição de permissões restritivas no sistema de arquivos e utilização de criptografia.

O Capítulo 6 trata da terceira camada, que irá representar o monitoramento do sistema através da utilização de ferramentas de monitoramento, integridade, detecção e prevenção de intrusão e pela inspeção de registros de eventos. Por fim, o Capítulo 7 trata da quarta camada, que irá representar os procedimentos para aumentar a disponibilidade do sistema e para recuperação do servidor, incluindo planos de contingência e *backup*.

CAPÍTULO 2 – CONCEITOS BÁSICOS DE SEGURANÇA

2.1 - A necessidade de segurança

Segundo Nakamura & Geus (2003), nas décadas de 70 e 80 o enfoque principal da segurança, nos negócios da organização, era o sigilo dos dados. Já nas décadas de 80 e 90, com o surgimento do ambiente de rede, a integridade era de suma importância, e a proteção era feita tendo em mente a informação e não os dados. A partir da década de 90 a informática tornou-se essencial para o negócio e com o crescimento das redes o enfoque passou a ser a disponibilidade, e a proteção passou a ser sobre o conhecimento.

A falta de planejamento em segurança pode parecer uma boa situação, pois tudo funciona adequadamente, até que surgem os problemas que podem resultar em custos elevadíssimos em sua resolução. O importante não é só funcionar, mas funcionar bem e com segurança. Muitas empresas ainda deixam a segurança em segundo plano, dando-lhe a devida importância somente quando ela se torna extremamente necessária.

A 9ª Pesquisa Nacional de Segurança da Informação (Módulo, 2003), realizada pela empresa Módulo *Security*[5], líder em consultoria na área de Segurança da Informação na América Latina, a visão das empresas sobre segurança esta mudando. Nos últimos anos, a segurança está deixando de ser vista como um gasto para as empresas, passando a ser vista como um investimento.

[5] http://www.modulo.com.br

Nesta pesquisa também foi identificado que, o principal obstáculo para implementação da segurança é a falta de consciência dos executivos, como mostrado na figura 2.1. Outra informação, é que o número de pessoas que já sofreram ataque passou de 43% em 2002 para 77% em 2003, como mostrado na figura 2.2.

Figura 2.1 - Principais obstáculos para implementação da segurança (Módulo, 2003)

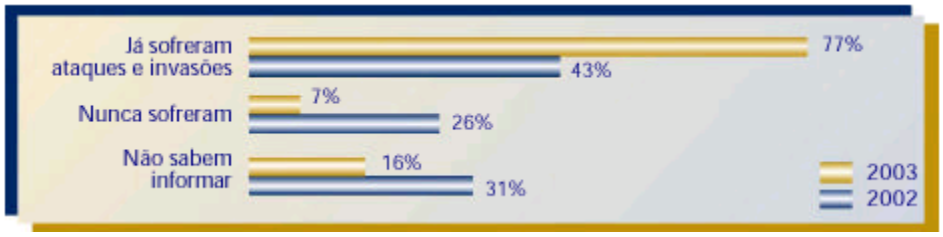

Figura 2.2 - Ocorrência de ataques e invasões (Módulo, 2003)

Conforme Nakamura & Geus (2003, p.46) "(...) a segurança é inversamente proporcional às funcionalidades". Assim, quanto maior o número de funcionalidades que um sistema disponibilizar, maior será a chance de haver alguma vulnerabilidade que pode ser explorada, e, em conseqüência, menor será a segurança do ambiente e maior será a responsabilidade dos administradores.

O administrador deve ter conhecimento suficiente sobre o sistema operacional para identificar quais são os componentes realmente necessários em um servidor para que um determinado serviço esteja disponível.

Ambientes cooperativos utilizam conexões complexas e aumentam a sua heterogeneidade com freqüência, disponibilizando cada vez mais serviços e permitindo comunicação entre diversos sistemas, sendo que a integração entre estas tecnologias e a segurança são essenciais.

2.2 - Riscos e considerações quanto à segurança

Como lembra Furmankiewicz & Figueiredo (2000), as organizações estão sujeitas a uma série de ataques às redes que envolvem uma mistura de técnicas, ações para cobrir vestígios, cooperação entre os atacantes e criatividade, dificultado a implantação de segurança.

A 9ª Pesquisa Nacional de Segurança da Informação (Módulo, 2003), indica que os principais fatores que comprometem a segurança são:

- a exploração de vulnerabilidades em sistemas operacionais, aplicativos, protocolos e serviços;
- a exploração de aspectos humanos das pessoas envolvidas;

- as falhas no desenvolvimento e implementação da política de segurança;
- e o desenvolvimento de ataques mais sofisticados.

Um atacante pode explorar aspectos humanos e tecnológicos, incluindo *hardware*, *software* e redes, de forma que, se um único ponto estiver vulnerável, a segurança pode ser comprometida. Desta forma, é importante ter uma ampla visão sobre segurança para que seja feito um planejamento adequado de uma política de segurança sem deixar nenhum desprotegido.

2.3 - Potenciais atacantes

Uma organização está sujeita tanto a ataques externos, quanto a ataques internos realizados pelos próprios funcionários ou por outros indivíduos.

No entendimento de Furmankiewicz & Figueiredo (2000), os *hackers* são aqueles que utilizam seu conhecimento para invadir sistemas, sem o intuito de causar danos às vitimas, mas sim, como um desafio às suas habilidades. Já os *crackers* são aqueles que tem a intenção de prejudicar a vítima, causando-lhe danos. Um sistema protegido pode dificultar as ações destes indivíduos.

De acordo com a 9ª Pesquisa Nacional de Segurança da Informação (Módulo, 2003), os principais responsáveis pelos ataques são os *hackers*. Porém, usuários, autorizados ou não, mesmo não intencionados, também podem causar prejuízos a empresa, como mostra a figura 2.3.

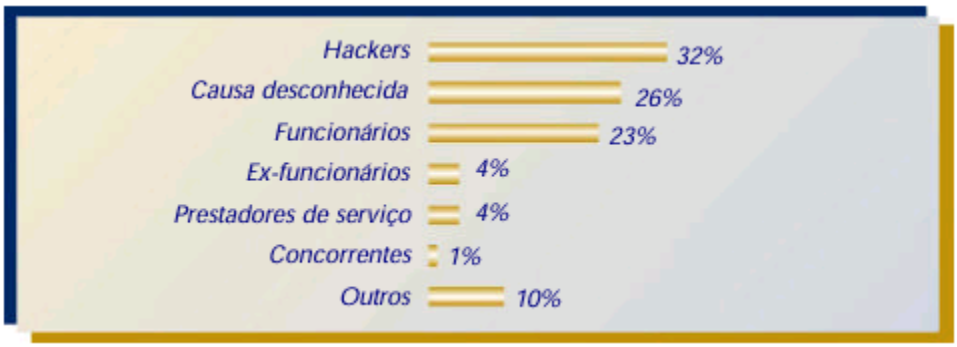

Figura 2.3 - Principais responsáveis pelos ataques (Módulo, 2003)

É importante destacar que na figura 2.3 não se distingue o termo *hacker* e *cracker*, o que é uma falha grave levando-se em consideração que esta pesquisa foi realizada por uma empresa bem conceituada na área de segurança.

2.4 - Pontos explorados

As invasões exploram deficiências na concepção, implementação, configuração ou no gerenciamento do sistema e dos serviços.

Conforme Gonçalves (2000), os ataques podem explorar vulnerabilidades existentes em qualquer um dos níveis relacionados à proteção

da informação. São incluídos nesta lista, sistema operacional, serviços e protocolos, rede e telecomunicações, aplicação, usuários e nível físico.

Para o *hacker*, pode ser suficiente explorar apenas uma vulnerabilidade em um desses níveis para conseguir acesso ao sistema, enquanto o profissional de segurança precisa encontrar e fechar todas as brechas.

Outro fator que colabora com a falta de segurança é o comportamento de alguns fabricantes que, na ânsia de lançar produtos no mercado antes dos concorrentes, preferem consertar falhas de segurança a construir sistemas conceitualmente seguros.

Os principais pontos de invasão utilizados pelo atacantes são os sistemas internos e principalmente a Internet, conforme mostrado na figura 2.4 retirada da 9ª Pesquisa Nacional de Segurança da Informação (Módulo, 2003).

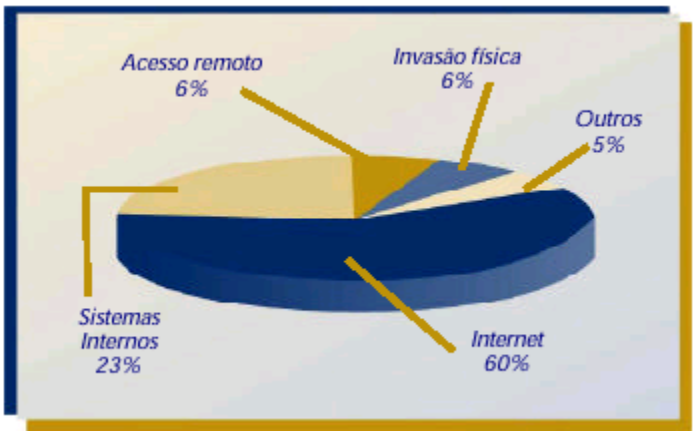

Figura 2.4 - Principais pontos de invasão (Módulo, 2003)

Devido à facilidade em se encontrar ferramentas na Internet com interface gráfica de fácil utilização, os ataques simples e comuns podem ser executados sem grandes dificuldades. A cada dia surgem novos tipos de ataques, e muitas ferramentas de defesa existentes protegem os sistemas somente contra ataques já conhecidos, representando um risco à segurança.

2.5 - Planejamento de um ataque

As motivações para um ataque são diversas, variando de acordo com o atacante. A motivação pode ser pelo aprendizado, dinheiro, fama, necessidades psicológicas, vingança, espionagem industrial ou curiosidade (Nakamura & Geus, 2003).

O primeiro passo para um ataque é a obtenção de informações. Segundo Gonçalves (2000), um atacante pode conseguir informações através do monitoramento da rede; penetração nos sistemas; inserindo códigos prejudiciais ou informações falsas no sistema; ou enviando uma enxurrada de pacotes ao sistema.

Para fazer um mapeamento de vulnerabilidades, um atacante pode procurar por fontes públicas de informações, usar código de exploração público postado em listas de distribuição e usar ferramentas de varredura de vulnerabilidades automatizadas.

A primeira técnica utilizada para o mapeamento de vulnerabilidades é o *footprinting*[6], visando descobrir cada elemento de informação possível. O

[6]*Footprinting*: Busca detalhada da maior quantidade de informações possíveis do alvo da invasão.

mapeamento de vulnerabilidades envolve: o reconhecimento de rede; o mapeamento de atributos como, o sistema operacional, arquitetura e versões de serviços; e a enumeração e priorização de pontos de entrada potenciais (Scambray et al., 2001) .

As conseqüências dos ataques podem ser variadas, tais como, vazamento de informações, modificações, corrupção de serviços, fraude, imagem prejudicada, perda de negócios, trabalho extra para recuperação, entre outras.

Segundo Wadlow (2000, p.193), "(...) em sua maioria, os atacantes são bastante inteligentes para não serem rastreados facilmente". Após ganhar acesso ao sistema, o atacante tentará encobrir todas as ações que ele realizar. Para isso, podem ser utilizadas técnicas de substituição ou remoção de arquivos de *log*, substituição de arquivos importantes para o mascaramento de suas atividades ou até a formatação do sistema.

2.6 - Tipos de ataques

Os principais tipos de ataques, de acordo com pesquisas realizadas por Nakamura & Geus (2003), são os ataques direcionados e os oportunistas. Os direcionados são menos comuns, porém, são os mais perigosos, pois envolvem pessoas com objetivos formulados que podem ter estudado a empresa antes de iniciar o ataque. Os ataques oportunistas são mais comuns e são realizados de maneira aleatória.

Diversas técnicas e ferramentas podem ser utilizadas na obtenção das informações que podem levar a um ataque de sucesso. Estas mesmas técnicas e

ferramentas podem ser utilizadas também pelo próprio administrador do sistema para identificar e corrigir vulnerabilidades.

O atacante pode utilizar engenharia social, ataques físicos, informações livres, *packet sniffing*[7], *port scanning*[8], *scanning de vulnerabilidades*[9] e *firewalking*[10] (Nakamura & Geus, 2003). O *IP spoofing*, técnica na qual o atacante pode disfarçar seu endereço IP dificultando a identificação da origem do ataque, é utilizado como técnica auxiliar para os outros métodos de obtenção de informações.

Conforme Uchôa (2005) os principais tipos de ataques são:

- *footprinting*, que consiste em coletar informações sobre um sistema alvo;
- *spoofing*, que consiste em fazer uma máquina se passar por outra;
- código malicioso, que consiste em *softwares* com códigos não autorizados que efetuam ações desconhecidas e não desejadas pelo usuário;
- *exploits*, que consistem em programas criados para explorar falhas;
- e ataques de senhas, que consistem em tentar descobrir a senha de um ou mais usuários.

[7] *Packet sniffing*: ataque no qual um intruso pode ler diretamente as informações transmitidas e o conteúdo da base de dados.
[8] *Port scanning*: aplicativo que faz uma varredura nas portas do equipamento informando seu estado.
[9] *Scanning de vulnerabilidades*: aplicativo que faz uma varredura no equipamento procurando por vulnerabilidades que possibilitem um ataque.
[10] *Firewalking*: método de envio de pacotes ao *firewall* com o objetivo de descobrir vulnerabilidades.

O ataque de senhas é bastante conhecido e utilizado, e pode ser realizado com certa facilidade. Se houver uma diretiva de bloqueio de conta de usuários após um determinado número de tentativas de *logon* sem sucesso, este ataque poderá ser evitado. Porém, esta diretiva pode levar ao bloqueio da conta de vários usuários da rede durante uma tentativa de invasão.

A engenharia social também é um tipo de ataque muito utilizado e difícil de ser combatido. Através da engenharia social, o atacante pode obter informações privilegiadas enganando os usuários, utilizando identificações falsas ou conquistando a confiança da vítima. Para isso diversos meios podem ser utilizados, entre eles, o telefone, *e-mail* ou contato direto.

No entendimento de Scambray et al. (2001), outros ataques comuns são:

- ataques de força bruta;
- ataques dirigidos por dados;
- ataques de estouro de *buffer*;
- ataques de validação de entradas;
- ataques de telnet reverso e canais de retorno;
- ataques a serviços como TFTP, NFS, sendmail, etc;
- ataques de estouro de pilha;
- ataques de descritor de arquivo;
- ataques de condição de corrida;
- ataques a bibliotecas compartilhadas;
- ataques a arquivos do cerne;
- ataques de falhas de *kernel* ;
- ataques a sistema configurados incorretamente;
- ataques de negação de serviços;

- e ataques de negação de serviços distribuídos.

Os ataques de negação de serviços ("*Denial-of-Service attack*" – DOS) permitem explorar os recursos de um servidor de maneira agressiva, de modo que usuários legítimos fiquem impossibilitados de utilizá-los.

Já os ataques coordenados são os mais evoluídos, também conhecidos como ataques de negação de serviços distribuídos ("*Distributed Denial of Service* – DDoS"). Este ataque faz com que diversos *hosts* distribuídos sejam atacados e coordenados para realização de ataques simultâneos aos alvos. Isso resulta em um ataque extremamente eficiente, no qual a vítima pode ficar praticamente indefesa, sem conseguir descobrir a origem dos ataques, já que estes procedem de *hosts* intermediários controlados pelo atacante.

Os ataques podem ser classificados como ataques de acesso remoto e local. Acesso remoto é definido como ganhar acesso via rede ou outro canal de comunicação. Acesso local, conhecido também como ataque de escalação de privilégio, é definido como ter um *login* ou *shell* de comando real no sistema (Scambray et al. 2001).

Uma vez que o atacante consegue acesso local ao sistema, este poderá coletar informações e utilizar a estação invadida como ponto de partida para ataques adicionais. Um sistema invadido pode não ser mais confiável se não houver ferramentas que permitam identificar quais as ações executadas pelo atacante no sistema durante a invasão.

CAPÍTULO 3 – SEGURANÇA EM CAMADAS

3.1 - Proposta de segurança em camadas

Em um ambiente cooperativo, a cada dia surgem novas formas de conectividade e comunicação entre os seus diversos elementos, no qual são utilizados diversos protocolos, sistemas operacionais, ferramentas, aplicativos e estruturas físicas.

Estes tipo de ambientes heterogêneo, está sujeito a diversos tipos de ataques que podem explorar os seus vários elementos. Muitas são as ferramentas e técnicas que podem ser utilizadas em um ataque, e se houver um único ponto vulnerável na rede, toda a segurança pode ser comprometida.

Para prover segurança nestes ambientes, devem ser utilizadas diversas técnicas e ferramentas tecnológicas, cada uma com um objetivo específico. Estas técnicas e ferramentas se complementam para prover um alto nível de segurança.

Devido a complexidade de um ambiente cooperativo e de seus elementos, esta monografia adotou a abordagem de apresentar a segurança em quatro camadas de proteção, com o objetivo de:

- facilitar a compreensão do leitor, descrevendo as técnicas e ferramentas tecnológicas utilizadas para prover segurança e tornando mais claro o entendimento sobre qual é o objetivo de cada uma, onde elas podem ser aplicadas e como interagem e se complementam;

- ampliar a visão sobre segurança, facilitando a identificação de quais elementos estão sendo protegidos e quais não estão, minimizando a possibilidade de que algum ponto seja esquecido. Desta forma, fica mais fácil identificar quais são as necessidades de segurança da empresa;
- e facilitar o planejamento e implantação de uma política de segurança. O administrador, ao entender como as tecnologias de segurança atuam em diversas camadas, tem uma visão mais clara sobre as necessidades de segurança de uma determinada empresa, tornando mais simples a tarefa de planejar e implantar uma política de segurança.

3.2 - As quatro camadas de segurança

Com base em estudos realizados durante o desenvolvimento deste trabalho, o autor desta monografia definiu quatro camadas de segurança, que são: *controle de acesso ao sistema*; *segurança interna do sistema*; *monitoramento do sistema*; e *recuperação e disponibilidade do sistema*.

A escolha destas quatro camadas foi baseada nas técnicas e tecnologias hoje empregadas para prover segurança, e nas ações mais comuns que um atacante executa ao tentar invadir um sistema.

Um atacante, inicialmente, utilizando como meio de comunicação uma rede, pode tentar realizar o mapeamento de vulnerabilidades do sistema, além de utilizar uma série de ferramentas para ganhar acesso ao sistema. Estas tentativas de acesso não autorizadas podem ser bloqueadas pela camada de *controle de acesso ao sistema*.

Em uma segunda ação, o atacante, ao ganhar acesso ao sistema, poderá aproveitar falhas na configuração interna do sistema, serviços e sistema de arquivos, para tentar escalar privilégios, ganhando acesso local como superusuário. A camada de *segurança interna do sistema* pode impedir esta ação.

Em uma terceira ação, uma atacante, ao conseguir acesso ao sistema, pode executar diversas ações que vão deste a alteração de arquivos, até a formatação do sistema. A implementação da camada de *recuperação e disponibilidade do sistema* pode tornar possível a recuperação do sistema a um estado anterior, de uma maneira segura, e em um tempo aceitável.

Por fim, a camada de *monitoramento do sistema,* irá atuar juntamente com as outras camadas, tornando possível detectar tentativas de invasão, verificar o funcionamento correto do sistema e dos serviços, e identificar as ações executadas por um atacante após ganhar acesso ao sistema. A partir das informações coletadas, é possível identificar o que precisa ser recuperado após um ataque. Para isso, podem ser utilizados os registros de eventos, e as informações geradas por ferramentas de monitoramento e integridade.

É importante entender que as camadas de segurança interagem e se complementam. Por exemplo, ao configurar um serviço com segurança, além de proteger a camada de *segurança interna do sistema,* também pode se estar protegendo a camada de *controle de acesso ao sistema.* Um serviço configurado incorretamente ou desatualizado, pode resultar em uma vulnerabilidade que pode permitir o acesso ao sistema.

A configuração correta do serviço também pode envolver a configuração do nível de informações que será armazenada em um *log* e o local de armazenamento deste *log*. O *log* gerado, poderá ser utilizado para verificar se o

serviço está funcionando corretamente, atuando na camada de *monitoramento do sistema*.

A figura 3.1, ilustra as quatro camadas se segurança. A camada de monitoramento é a que tem maior interação com as outras três camadas. O canal de acesso, representa o meio utilizado pelo atacante para ter acesso ao sistema. Por exemplo, o canal de acesso pode ser uma conexão de rede.

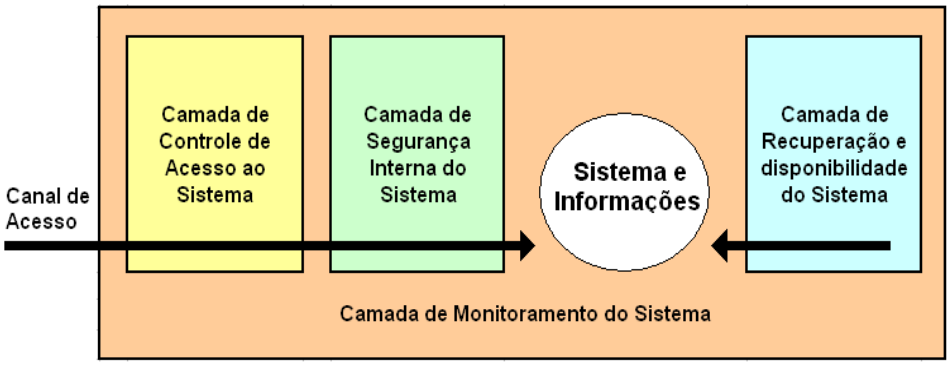

Figura 3.1 – Camadas de segurança

Cada camada tem uma função e objetivo. A camada de *controle de acesso ao sistema* trata das interações dos usuários, dispositivos ou aplicativos durante o acesso ao sistema. Para aumentar a segurança de um sistema, apenas os usuários autenticados, devem ter acesso somente às portas e protocolos autorizados em um determinado servidor.

A segunda camada trata da segurança interna do sistema. Um sistema torna-se mais seguro quando os seus componentes são atualizados e configurados corretamente conforme documentação oficial do desenvolvedor.

Apenas os componentes realmente necessários devem ser adicionados ao sistema. Outro ponto importante é definir permissões restritivas no sistema de

arquivos e usar criptografia para proteger arquivos ou diretórios, ou para proteger as comunicações de rede.

A camada *monitoramento do sistema,* trata das técnicas e tecnologias utilizadas para monitorar os componentes do sistema. O monitoramento do sistema pode ser realizado através do monitoramento de registro de eventos, ferramentas específicas de monitoramento e verificação de integridade, ou *scripts*.

O monitoramento do sistema permite que sejam monitorados os componentes de *hardware* e *software* do computador. Conforme Uchôa (2005), em um sistema medianamente seguro, uma invasão irá exigir esforço e tempo, de forma que, com um monitoramento eficiente, a invasão pode ser bloqueada em seu início.

A camada de *recuperação e disponibilidade do sistema,* trata das técnicas e tecnologias utilizadas para garantir a recuperação do sistemas e dos dados após um desastre, e para aumentar a disponibilidade do sistema. Um procedimento de contingência bastante utilizado para recuperação é a realização de *backups*.

Entre as tecnologias empregadas para aumentar a disponibilidade de um sistema, estão, a utilização de *clusters*, *storages* e RAID de discos.

A figura 3.2, representa uma tentativa de ataque ao sistema. Um atacante irá utilizar um canal de acesso para tentar transpor as camadas de *monitoramento*, *controle de acesso* e *segurança interna*.

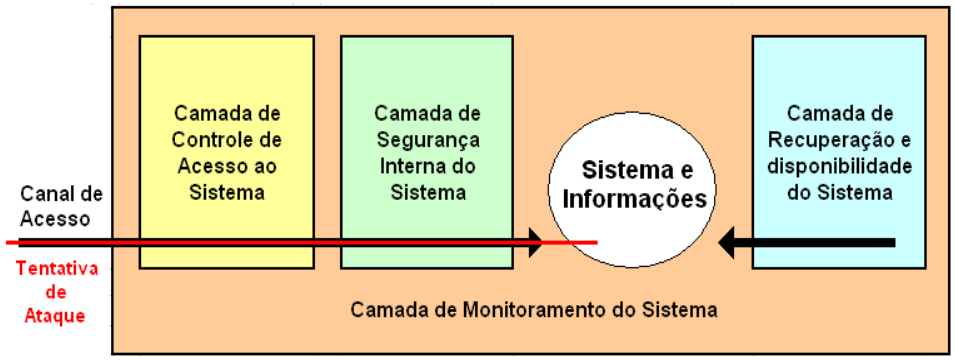

Figura 3.2 – Tentativa de Ataque ao sistema

Uma vez que o ataque foi realizado com sucesso, o administrador do servidor, a partir das informações coletadas pela camada de monitoramento, poderá utilizar a camada de recuperação para restaurar o que foi perdido ou alterado e restabelecer a integridade do sistema e das camadas de segurança. Por exemplo, as camadas podem ser reestabelecidas através da restauração de um *backup* e identificação e correção da vulnerabilidade que permitiu a invasão.

3.3 – Política de segurança

O administrador, ao conhecer as ameaças a que uma rede está sujeita e ao entender como as tecnologias de segurança atuam em diversas camadas, tem

uma visão mais clara sobre as necessidades de segurança de uma empresa, facilitando assim o planejamento e a implantação de uma política de segurança.

No entendimento de Uchôa (2005), a definição de uma política de segurança é o elemento mais importante da segurança em redes, e deve envolver a segurança física, segurança lógica, privacidade e a legalidade de s*oftware*. Sem uma política de segurança bem elaborada não se sabe o que se vai proteger, nem porque ou qual a melhor forma.

O objetivo da política de segurança não é definir procedimentos específicos de manipulação e proteção da informação, mas atribuir direitos e responsabilidades às pessoas que lidam com essa informação (NBSO, 2003).

A política de segurança define o que deve ser feito e não o meio para se fazer. Por exemplo, é definido na política de segurança que as senhas utilizadas nos sistemas da empresa devem atender a quesitos de complexidade e devem ter no mínimo oito caracteres. Também é definido que devem ser utilizados mecanismos para forçar o usuário a cumprir estas definições. Observe que, a política definiu o que deve ser feito, mas não as ferramentas e as configurações necessárias para se chegar a este objetivo.

Se a política de segurança for definida independentemente de *hardware* ou *softwares* específicos, esta política terá sua vida útil prolongada e irá requerer um número menor de atualizações.

De acordo com Uchôa (2005), quando se trata de segurança computacional, uma boa abordagem é adotar a política do menor privilégio, bloqueando todos os recursos e liberando apenas os recursos essenciais ao funcionamento da rede. Porém, a adoção de algumas medidas definidas pode implicar em perda de performance ou conveniência para o usuário, de forma

que, as medidas a serem adotadas devem ser devidamente estudadas antes da sua implantação.

A política de segurança norteia e sustenta as ações do administrador para aumentar o nível de segurança da rede. Por exemplo, o administrador, seguindo a política de segurança, configura o sistema para que as estações sejam automaticamente bloqueadas após dez minutos de inatividade. Se os usuários ficarem insatisfeitos com a mudança, o administrador estará apoiado pela existência das políticas de segurança, que definem que esta regra deve existir. Isto é, não foi o administrador que decidiu implantar segurança e sim a empresa.

Conforme NBSO (2003), antes de definir a política, é necessário definir a informação a ser protegida. Uma política deverá cobrir os seguintes aspectos:

- aspectos preliminares, como, por exemplo, abrangência, escopo, normas e regulamentos e meios de distribuição.
- política de senhas;
- direitos e responsabilidades dos usuários;
- direitos e responsabilidades do provedor dos recursos, como, por exemplo, *backup*, monitoramento e normas de segurança física;
- e ações previstas em caso de violação da política.

A definição da política de segurança será influenciada por vários elementos, incluindo, a infra-estrutura de rede, requisitos de segurança de uma empresa, e a visão do administrador de quais são as necessidades de segurança da empresa. Esta visão pode ser ampliada a partir do momento que se conhece as diversas camadas de segurança, entendendo onde cada elemento pode atuar, facilitando assim o planejamento e definição das políticas. É importante que a

política cubra todos os pontos que envolvam segurança e que seja mantida atualizada.

As políticas devem ser devidamente divulgadas, conhecidas e apoiadas por todos os níveis da organização, incluindo o alto escalão da empresa. Também deve ser verificado se as políticas estão sendo realmente aplicadas e cumpridas na empresa.

Nos próximos capítulos serão descritos os componentes que integram cada uma das quatro camadas de segurança.

CAPÍTULO 4 – CAMADA 1: ACESSO AO SISTEMA

4.1 - Localização do servidor

O primeiro passo para restringir o acesso a um servidor é definir a localização física e lógica deste servidor.

Um servidor deve estar localizado fisicamente em um local seguro onde apenas pessoas autorizadas possam ter acesso físico a este servidor. Também é importante que o servidor esteja conectado a uma rede elétrica estabilizada e protegida por um *no-break*.

Em relação a localização lógica, se o servidor for acessado através da Internet, o mesmo deverá estar protegido por um *firewall* em uma zona desmilitarizada (DMZ), permitindo o controle de acesso a este servidor.

4.2 - Autenticação

Os mecanismos de autenticação tem como objetivo verificar a identidade de um usuário ou a elegibilidade do usuário para acessar um objeto do sistema. A identificação e autenticação são responsáveis pela verificação do usuário, permitindo acesso ou não aos sistemas e aos recursos da organização. A identificação é a função em que o usuário declara uma determinada identidade para um sistema, enquanto a autenticação é a função responsável pela validação

dessa declaração de identidade do usuário. Através da auditoria é possível verificar tentativas de acesso válidas e inválidas.

Segundo Northcutt et al. (2001), a autenticação pode ser feita com base no que o usuário sabe (senha, chave criptográfica, número de identificação), com base no que o usuário possui ("*token*", cartão ou "*smart card*") ou com base nas características do usuário (biometria, ou seja, reconhecimento de voz, retina, íris, impressão digital, etc). Todos esses métodos têm pontos positivos e negativos, sendo recomendada a combinação de dois ou mais métodos. Após ser autenticado o usuário poderá ter diversos níveis de acesso. Deve ser configurado o nível de acesso mais restritivo permitindo acesso apenas aos recursos realmente necessários.

Um atacante poderá utilizar várias técnicas para tentar se autenticar em um sistema. Conforme Scambray et al. (2001), um ataque muito utilizado para ganhar acesso ao sistema é o ataques de força bruta[11], sendo que, os tipos de serviço mais atacados incluem: telnet, FTP, serviços remotos como rlogon e rsh, POP e HTTP. Estes serviços devem ser substituídos por serviços mais seguros que utilizem túneis criptográficos na comunicação.

Outro ataque conhecido é o de dicionário automatizado, que consiste em tentar adivinhar a senha de uma dada conta criptografando uma palavra ou texto e comparando o resultado com o *hash*[12] de uma senha criptografada obtida. Entre programas mais conhecidos para Linux estão o *Crack 5.0a*, e o *John the Ripper*[13].

---

[11] Ataque de força bruta: é um método de obtenção de senha ou outros textos criptografados tentando-se todos os valores possíveis.

[12] *Hash*: é um método para transformar dados de tal forma que o resultado seja exclusivo e não possa ser retornado ao formato original.

[13] http://www.openwall.com/john/

Para dificultar a realização deste tipo de ataque, as senhas do sistema devem ser definidas com base nos critérios de complexidade definidos na política de segurança. A senha padrão de alguns serviços e usuários também deve ser alterada.

Algumas ferramentas podem ser utilizadas para impedir que usuários escolham senhas inseguras. Outras podem ser utilizadas para descobrir senhas fáceis de serem quebradas.

Outra ação para dificultar os ataques é utilizar o PAM, que permite implementar a autenticação de usuários de uma forma altamente configurável. Conforme Uchôa (2005), o PAM possui diversos módulos que podem ser utilizados para incrementar a segurança controlando itens como: limites de recursos, uso de senha escondida, limite de acesso, *shell* restrito, entre outros. O PAM é usado para implementar a autenticação de usuários de uma maneira confiável, permitindo que a autenticação atenda à diversas necessidades

O PAM disponibiliza quatro tipos diferentes de módulos sendo cada um relacionado com a autenticação de usuários. Também são definidos quatro sinalizadores de controle possíveis, que determinam a ação do aplicativo, quando o comando é bem sucedido ou falha. Para mais detalhes sobre o PAM, consulte (Morgan, 2006).

4.3 - Filtro de pacotes

Os filtros de pacote consistem em mecanismos que fazem o encaminhamento de pacotes, conforme o conteúdo de certos campos dos

cabeçalhos dos pacotes (Foster, 1998). O *kernel* do Linux conta com um filtro de pacotes bastante funcional que é controlado por regras de *firewall.*

Antes de configurar um *firewall* em um servidor, é importante configurar o sistema corretamente, de forma que apenas os componentes essenciais estejam instalados. A partir daí, deve-ser conhecer quais são as portas e protocolos necessários aos serviços que serão disponibilizados. Se o administrador tem um sistema com componentes e serviços ativados sem necessidade ou se não sabe quais as portas e protocolos devem ser autorizados, mesmo com o *firewall* ativado podem ser deixadas brechas de segurança.

Segundo Jang (2003), um *firewall* pode verificar cada pacote de dados que entra na sua rede e tomar decisões com base no tipo de dados ou serviços, sendo possível ajustar diferentes níveis de proteção para diferentes computadores.

Um *firewall* composto de várias regras, pode ser colocado no ponto de entrada da rede, sendo direcionado para uma ou mais interfaces de rede, fazendo o controle do acesso a serviços para toda a rede (Teixeira e Mercer, 2004). Desta maneira, a configuração pode ser feita em um único local, para proteção de todas as máquinas da rede.

Um *firewall* também pode ser instalado em servidores específicos visando a proteção destes servidores. Segundo Jang (2003) o principal *firewall* do Linux é o *Iptables*[14]. Para detalhes de instalação, configuração e uso do *Iptables*, consulte (Andreasson, 2005).

Ao se instalar um *firewall* em um servidor, é recomendado configurar regras para bloquear todo o tráfego de rede e permitir apenas as portas e protocolos utilizados na rede e apenas para as redes ou estações que devem ter

[14] http://www.netfilter.org

acesso ao serviço. O *firewall* também deve ser configurado com regras para bloquear ataques conhecidos como, por exemplo, o *ping da morte,* e para registrar eventos significativos em *logs*.

Um *firewall,* também pode ser usado para configurar o mascaramento de IP, que permite ocultar endereços IP's de computadores em uma rede local, substituindo estes por um endereço IP público, dificultando assim ataques diretos a estações em uma rede. O mascaramento de IP é uma forma de *Network Address Translation* (NAT). O NAT também pode ser implementado através de um servidor *proxy*[15].

De acordo com Uchôa (2005), um *proxy* é um *software* que atua como ponto entre duas redes, permitindo controle de tráfego de acordo com o seu conteúdo. É possível utilizá-lo para filtrar o tráfego de rede, de forma que, um *proxy* pode ser utilizado como *firewall*. Da mesma forma, um *firewall* pode ser configurado para funcionar como *proxy* quando se utiliza, por exemplo, o *Iptables* para mascaramento de pacotes ou NAT, o que equivale a um *proxy* transparente. O *proxy* mais utilizado atualmente no Linux é o *Squid*[16]. Para detalhes de instalação, configuração e uso do *Squid*, consulte (Pearson, 2006).

Um *proxy* pode mudar o endereço dos pacotes de rede de um cliente com objetivo de protegê-lo contra ataques vindos da Internet, sendo que, o único endereço que é utilizado para acesso a Internet, é o do próprio *proxy*. Desta forma, é reduzida a possibilidade de um atacante obter informações e executar um ataque aos computadores da rede interna.

Utilizando um *proxy,* também é possível bloquear aplicativos, usar listas de controle de acesso (ACL's) para criar filtros definindo diversos tipos de

[15] *Proxy*: Serviço que recebe pedidos de computadores ligados à rede e redireciona esses pedidos ao exterior da rede.
[16] http://www.squid-cache.org

permissões, e impor políticas de utilização. Também é possível coletar dados estatísticos e controlar o acesso à Internet mediante autenticação.

Existem outras ferramentas que podem fornecer uma camada extra de proteção para o sistema, controlando o acesso com base no endereço IP. Por exemplo, no Linux, o TCP-Wrappers, permite filtrar pacotes direcionados a serviços oferecidos por vários *daemons*[17].

[17] *Daemon:* Programa em execução em um computador que recebe solicitações de outros programas, executa determinadas ações e retorna resultados.

CAPÍTULO 5 – CAMADA 2: SEGURANÇA INTERNA DO SISTEMA

5.1 - Configurando o sistema com segurança

Diversos mecanismos podem ser utilizados para aumentar a segurança de um sistema, sendo que, o primeiro passo começa na instalação do sistema operacional. A princípio, o administrador deverá definir qual a distribuição será utilizada. É importante que o administrador escolha uma distribuição confiável e que ele domine, para facilitar a tarefa de configurar o sistema corretamente e com segurança.

Ao instalar o sistema, o administrador deve planejar o particionamento dos discos, definir quais os serviços serão necessários para o servidor e instalar apenas os componentes essenciais.

Quanto menor o número de componentes instalados em um sistema, menor é a possibilidade de surgir uma nova vulnerabilidade neste sistema. Se o administrador dominar as ferramentas de linha de comando e arquivos de configuração necessários para configuração do sistema e dos serviços, a instalação dos componentes gráficos pode até ser dispensada. Desta forma, o administrador reduz o número de componentes a serem gerenciados tornando o sistema mais “simples” e seguro.

Se a rede na qual o servidor estiver conectado for considerada insegura, o ideal é que o servidor seja instalado e atualizado fora da rede, utilizando um CD-ROM com uma versão atual do sistema, reduzindo assim a possibilidade de um ataque ao servidor antes que todas as configurações de segurança sejam realizadas.

Conforme Uchôa (2005), após instalado o sistema deve ser evitado efetuar *logon* com o superusuário, sendo que, quando for necessário executar uma tarefa que exija poderes de superusuário, deve ser utilizado o aplicativo *sudo*. Outra medida de segurança é definir quotas de disco para usuários, permitindo controlar o espaço em disco e limitar tentativas de invasão.

Os serviços devem ser configurados de forma a torná-los mais seguros para o cliente e para o servidor, sendo importante estudar a documentação oficial do desenvolvedor para que este objetivo seja alcançado. Quando possível, é recomendado oferecer apenas um tipo de serviço de rede por servidor.

Deve ser definido um usuário com os privilégios mínimos necessários para iniciar os serviços, sendo que, deve-se evitar ao máximo iniciar um serviço com o superusuário.

Outro ponto importante é definir o nível de detalhamento de registro de informações dos serviços em *log* e verificar nos *logs* se os serviços estão sendo executados corretamente.

Segundo Uchôa (2005), a configuração adequada do sistemas de arquivos também é importante para a segurança do sistema. Uma recomendação é a utilização de opções de montagem que restrinjam o acesso ao disco. Também é interessante a utilização de ferramentas para definir a permissão padrão com que os arquivos são criados pelo usuário e a utilização de atributos de um arquivo.

Outra medida de segurança, é evitar executar aplicativos e protocolos que trafegam senhas em texto claro, substituindo estes, por aplicativos e protocolos mais seguros. Como exemplo, temos o SSH, o SFTP e o IMAPS. Além de utilizar aplicativos seguros, é importante utilizar os recursos de segurança que estes aplicativos disponibilizam.

Por exemplo, o serviço SSH pode ser configurado para permitir acesso apenas com o uso de certificado digital. Ao configurar o serviço SSH com segurança na camada de *segurança interna do sistema*, também está sendo protegida a camada de *controle de acesso ao sistema*. Neste exemplo, fica claro como as camadas de segurança interagem e se complementam.

Serviços como *Webmail* e *sites* de compras *on-line,* devem utilizar o protocolo HTTPS. Com o HTTPS é possível transmitir os dados através de uma conexão criptografada e verificar a autenticidade do servidor e do cliente através de certificados digitais.

De acordo com Uchôa (2005), para evitar a ação dos *exploits* é importante que o administrador de rede se mantenha sempre informado sobre falhas em serviços e disponibilização de novas correções em *sites* especializados. O servidor deve ser atualizado periodicamente assim que as correções forem disponibilizadas.

Após configurar o sistema, o administrador pode utilizar ferramentas de escaneamento de portas e de vulnerabilidades, para verificar se existem vulnerabilidades no sistema.

5.2 – Criptografia

Conforme Nakamura & Geus (2003), a criptografia é uma ciência que tem importância fundamental para a segurança da informação, ao servir de base para diversas tecnologias e protocolos. A criptografia possibilita a integridade, a autenticidade, o não-repúdio e o sigilo da informação.

Diversos fatores devem ser analisados para a proteção adequada da informação. Entre os principais estão: geração das chaves, mecanismos de troca das chaves, taxa de troca das chaves, tamanho das chaves, qualidade do algoritmo e sua correta implementação (Bernstein, 1997).

Um algoritmo criptográfico é considerado eficiente se não existirem facilidades que permitam que se recuperem as informações sem a utilização de ataques de força bruta e se o número de chaves possíveis for suficientemente grande para fazer com que os ataques de força bruta se tornem impraticáveis.

Com o aumento exponencial da capacidade de processamento e o avanço da computação distribuída, é essencial considerar o tempo durante o qual a informação deverá ficar protegida, para que seja utilizado o tamanho ideal de chave.

De acordo com Jang (2003), é possível ativar a criptografia em diferentes níveis de segurança, como, em senhas, serviços e sistemas. Entre os tipos de criptografia para sistemas estão: Senhas MD5, *Shadow Password Suite*, GNU *Privacy Guard*, RSA e DSA.

A criptografia pode ser utilizada para tornar a comunicação em uma rede mais segura, através da criação de túneis criptografados. Vários serviços podem ser tunelados utilizando protocolos seguros, como o SSL ou SSH. Os protocolos seguros possibilitam a encriptação, autenticação e integridade dos dados, possibilitando proteger a informação mesmo quando transferida através de redes públicas inseguras.

Além de tornar as comunicações em rede mais seguras, a criptografia pode ser utilizada para proteger arquivos pessoais dos usuários, de forma que, mesmo que um sistema seja invadido, os arquivos criptografados não possam ser acessados pelo invasor.

Outra aplicação da criptografia pode ser vista em redes privadas virtuais (VPN'S). Conforme Uchôa (2005), em uma VPN existe um túnel criptográfico entre duas ou mais redes que estão conectadas utilizando como meio uma rede pública.

Uma VPN usa protocolos de criptografia por tunelamento que fornecem a confidencialidade, autenticação e integridade necessárias para garantir a privacidade das comunicações, possibilitando comunicações seguras através de redes inseguras.

A criptografia pode também ser aplicada em um cliente de *e-mail*, que pode utilizar um certificado digital para criptografar uma mensagem, protegendo o conteúdo da mensagem contra acessos não autorizados.

CAPÍTULO 6 – CAMADA 3: MONITORAMENTO DO SISTEMA

6.1 - IDS e IPS

Um IDS (Sistema de Detecção de Intrusos) é uma ferramenta utilizada para monitorar o tráfego da rede e para detectar e alertar sobre ataques e tentativas de acessos indevidos. Na grande maioria das vezes não bloqueia uma ação, mas verifica se esta ação é ou não uma ameaça para um segmento de rede (Antunes, 2005).

Como complemento do IDS, o IPS (Sistema de Prevenção de Intrusos), tem a capacidade de identificar uma intrusão, analisar a relevância do evento e bloquear determinados eventos, fortalecendo assim a técnica de detecção de intrusos.

O IPS usa a capacidade de detecção do IDS juntamente com a capacidade de bloqueio de um *firewall*, notificando e bloqueando de forma eficaz ações suspeitas ou indevidas em diversos pontos de uma arquitetura de rede, sendo considerada uma das ferramentas de segurança de maior abrangência. O IPS atua na camada de monitoramento, mas pode ser configurado para gerar ações de bloqueio, de forma que, pode atuar também na camada de *controle de acesso ao sistema*.

O IDS e IPS devem ser configurados de forma a reduzir o número de falsos positivos facilitando a análise dos dados coletados. Configurado de forma incorreta, um IPS pode bloquear tráfego legítimo. Conforme Uchôa (2005), o IDS no seu sentido mais restrito se refere a aplicativos capazes de alertar sobre

tentativas de invasão, como o aplicativo *Snort*[18], e em um sentido mais amplo, também monitorar a integridade do sistema, como os aplicativos *AIDE*[19] ou o *Tripwire*[20].

Alguns aplicativos utilizados para detecção de intrusos, são capazes de desenvolver análise de tráfego e registro de pacotes em tempo real. Estes aplicativos analisam protocolos, buscam e associam padrões de conteúdo e podem ser usados para detectar uma variedade de ataques.

O *Snort* e o LIDS (Sistema de Detecção de intrusos do Linux) são algumas ferramentas que merecem destaque.

De acordo com Uchôa (2005), o *Snort* possui uma base de assinaturas bastante completa e exige pouco esforço computacional da máquina onde é instalado. Para mais detalhes sobre o Snort, consulte (Roesch et al., 2003). Já o LIDS, adiciona novas funcionalidades ao *kernel* do Linux para detecção de intrusos aumentando a segurança do sistema. Porém, para sua utilização, é necessária a recompilação do *kernel*.

6.2 - Registro de eventos

No entendimento de Uchôa (2005), o monitoramento de arquivos de registro é importante para identificar falhas em serviços, tentativas de invasão e ações anormais no sistema. Também é importante configurar o nível de detalhamento de registro das informações de acordo com a necessidade da

[18] http://www.snort.org
[19] http://www.cs.tut.fi/~rammer/aide.html
[20] http://www.tripwire.org

empresa e criticidade do serviço. Existem utilitários, como o *logwatch*[21], que permitem informar ao superusuário através de *e-mail* sobre registros ligados à segurança do sistema.

É importante inspecionar arquivos de registro para verificar atividades suspeitas. Por exemplo, nos arquivos de registro podem ser encontrados IP's que não deveriam estar acessando a rede, ou registro de atividades suspeitas em horários incomuns.

Um atacante pode camuflar uma invasão substituindo arquivos de registro de um computador. Para identificar ações realizadas pelo atacante, o administrador deve utilizar ferramentas de verificação de integridade.

6.3 – Monitoramento do sistema

Segundo Campos (2004), o MRTG[22] (*Multi Router Traffic Grapher*) é um *software* que facilita a tarefa de acompanhar o funcionamento do sistema. Embora o seu foco seja o acompanhamento de componentes de rede, é possível utilizá-lo para verificar o funcionamento de um servidor.

Outro recurso de segurança que pode ser utilizado para monitorar uma rede é o *honeypot* (pote de mel), cuja principal função é coletar informações, permitindo identificar um ataque e estudar as técnicas utilizadas pelo atacante, visando tornar um sistema real mais seguro.

[21] http://www.logwatch.org
[22] http://www.mrtg.org

Aplicativos farejadores, como o *Ethereal*[23] e o *Dsniff*[24], também são muito utilizados para capturar, interpretar e armazenar pacotes que viajam por uma rede. Estes aplicativos, podem ser utilizados pelo administrador da rede para solucionar problemas ou modelar o comportamento de uma rede, além de possibilitar identificar aplicativos que transmitem senhas em texto claro. Para detalhes de instalação, configuração e uso do *Ethereal*, consulte (Lamping et al., 2005).

Ferramentas como o *Nessus*[25], normalmente utilizadas por atacantes, também podem ser utilizadas como ferramentas de auditoria de segurança ajudando a identificar vulnerabilidades em uma rede. Para mais detalhes sobre o *Nessus*, consulte (TENABLE, 2003).

Conforme Uchôa (2005), é recomendável que o administrador verifique periodicamente as senhas dos usuário utilizando aplicativos como o *John The Ripper*[26], que realiza quebra de senha com base em dicionários. É importante também verificar o arquivo */etc/passwd* procurando por entradas incorretas ou estranhas.

6.4 - Integridade do sistema

Um invasor ao obter acesso a um sistema pode garantir a continuidade desse acesso através do uso de *rootkits*[27]. Para que seja possível monitorar

[23] http://www.ethereal.com
[24] http://www.monkey.org/~dugsong/dsniff
[25] http://www.nessus.org
[26] http://www.openwall.com/john
[27] *Rootkit*: Ferramenta capaz de capturar o tráfego de uma estação.

algumas ações praticadas pelo atacante é importante a utilização de ferramentas de verificação de integridade que fazem um *checksum* dos arquivos para posterior comparação. É recomendado se criar um *checksum* inicial logo após a instalação do sistema, enquanto este ainda é confiável (Uchôa, 2005).

Um atacante, ao invadir um sistema, pode tentar alterar os arquivos gerados pelas ferramentas de integridade. Para dificultar esta ação, estes arquivos devem ser armazenados em um meio seguro ou com permissões de apenas leitura, como, por exemplo, em um CD-ROM.

Uma vez que o sistema foi invadido, o administrador deverá tentar identificar como o invasor conseguiu este acesso, qual é a vulnerabilidade existente e o que foi alterado. Após analisar as ações executadas pelo atacante, o administrador deverá tomar as ações necessárias, que vão desde a correção de uma vulnerabilidade encontrada e restauração de um *backup*, até a formatação do sistema se este não for mais confiável.

De acordo com Uchôa (2005), aplicativos muitos utilizados para verificação de integridade no Linux são: *md5sum*, *AIDE* e o *Tripwire*. Também existem ferramentas utilizadas para detectar *rootkits* já instalados no servidor, como, por exemplo, o *chkrootkit*[28].

[28] http://www.chkrootkit.org

CAPÍTULO 7 – CAMADA 4: RECUPERAÇÃO E DISPONIBILIDADE DO SISTEMA

7.1 - Plano de contingência

A existência de um plano de contingência visa facilitar a continuidade das operações da empresa em caso de problemas. O Plano de Contingência é um documento onde estão definidas as responsabilidades e a organização para atender a uma emergência, contendo informações detalhadas sobre os procedimentos a serem adotados.

Um plano de contingência pode ser definido em várias situações. Por exemplo, pode ser criado um plano de contingência com as ações a serem tomadas caso o sistema seja atacado por um invasor, ou no caso de uma falha de *hardware.* Também pode ser criado um plano de contingência durante o planejamento de uma alteração com impactos significativos no ambiente, como, por exemplo, a migração de um sistema ou serviço crítico.

7.2 - *Backup*

O *backup* e *restore* representam uma das principais ações que podem ser tomadas para possibilitar a restauração do sistema a um estado anterior e a recuperação de informações.

Antes de criar rotinas de *backup*, devem ser definidas quais informações são importantes, além de conhecer bem o sistema para que seja feito o *backup*

dos arquivos corretos, evitando surpresas durante a restauração dos arquivos e reduzindo a quantidade de informações a serem armazenadas.

A política de segurança da empresa irá definir o tipo de *backup* e a freqüência com que os *backups* e testes de restauração devem ser realizados, periodicidade de troca das fitas e tempo que os dados deverão ficar armazenados. A estratégia de *backup* depende dos riscos que a empresa estiver disposta a correr.

Segundo Jang (2003), o Linux disponibiliza vários comando que podem ser utilizados para *backup* e *restore*, entre eles, *tar*, *cpio*, *dump* e *restore*.

7.3 - Disponibilidade

No entendimento de Uchôa (2005), para atender aos requisitos mínimos de disponibilidade dos dados, devem ser realizados *backup* periódicos. Em ambientes críticos podem ser utilizadas estratégias de "Alta Disponibilidade", que consistem em mecanismos para fazer com que um serviço esteja no ar o maior tempo possível através da utilização de servidores redundantes, sincronização de dados *on-line*, entre outras técnicas.

A disponibilidade de um sistema computacional, indicada por *A(t)*, é a probabilidade de que este sistema esteja funcionando e pronto para uso em um dado instante de tempo *t (*Arruda et al., (2002).

Adicionando-se mecanismos especializados de detecção, recuperação e redundância ou replicação, pode-se aumentar a disponibilidade do sistema, de forma que este venha a se enquadrar na classe de Alta Disponibilidade. Nesta classe as máquinas apresentam disponibilidade na faixa de 99,99% a 99,999%,

podendo ficar indisponíveis por um período de pouco mais de cinco minutos até uma hora em um ano de operação.

O principal objetivo da Alta Disponibilidade é manter os serviços prestados por um sistema disponíveis, mesmo que o sistema sofra uma falha. Para manter a disponibilidade dos serviços prestados, podem ser utilizadas estratégias de tolerância a falhas, entre elas, redundância de *hardware* e reconfiguração de *software,* que possibilita que um servidor assuma os serviços de um outro servidor que venha a falhar.

Também é possível utilizar RAID (vetor redundante de discos independentes), que permite que, se um disco falhar, outros disco no vetor possam assumir a sua funcionalidade. Alguns tipos de RAID podem oferecer alta disponibilidade, tolerância a falhas e recuperação.

Outra forma de aumentar a disponibilidade é através de *cluster.* Existem vários tipos de *cluster*, dentre os quais os mais conhecidos são os *clusters* que garantem a alta disponibilidade e o balanceamento de carga, além dos que combinam as duas características.

Mesmo sendo um solução de custo elevado, um nível maior de disponibilidade pode ser mais barato do que ter um sistema crítico indisponível. É importante analisar a disponibilidade do seu RAID, questionar a infra-estrutura utilizada e realizar testes de performance nos servidores para avaliar quais são as necessidades do seu sistema.

CAPÍTULO 8 - CONCLUSÃO

Através desta monografia chegou-se a conclusão que, com a rápida evolução tecnológica, as diversas formas de comunicação utilizadas, o lançamento de produtos no mercado com falhas de segurança e as diversas interações com o meio interno e externo que ocorrem em um ambiente cooperativo, dizer que um ambiente é totalmente seguro não é correto. Os administradores de redes devem proteger dezenas ou centenas de pontos potenciais de ataque a rede, enquanto, ao atacante, basta explorar um ponto vulnerável.

O sistema operacional Linux é complexo e exige um grande conhecimento do administrador de rede para a implementação de medidas de segurança adequadas. É importante que o responsável pela segurança conheça as técnicas e tecnologias utilizadas pelos atacantes para tornar o seu sistema mais seguro. Novas falhas de segurança e técnicas de exploração são descobertas a cada dia, sendo importante que os administradores de rede mantenham-se sempre atualizados.

Existem diversas técnicas de ataques remotos e locais que atacantes podem utilizar para invadir até mesmos sistemas com nível de segurança elevado. Cabe ao responsável pela segurança implantar uma série de medidas de segurança em diversas camadas para impedir ou dificultar a realização destes ataques.

Após entender os riscos a que uma rede está sujeita e conhecer técnicas e tecnologias que podem ser empregadas para prover segurança, além de entender como estas tecnologias atuam em diversas camadas, o administrador de

rede terá uma visão mais clara sobre as necessidades de segurança de uma empresa, facilitando assim o planejamento e implementação de uma política de segurança.

Com base em estudos realizados durante o desenvolvimento deste trabalho, o autor desta monografia definiu as quatro camadas de segurança apresentadas. Inclusive, a definição destas camadas e a apresentação dos benefícios da sua utilização foram as principais contribuição deste trabalho.

A utilização de diversas camadas de segurança torna possível proteger, monitorar e recuperar um sistema após um desastre. A abordagem em camadas também torna mais clara a compreensão do leitor e amplia a visão sobre segurança, facilitando a identificação de quais elementos estão sendo protegidos e quais não estão, minimizando a possibilidade de que algum ponto seja esquecido.

A divisão da segurança em camadas também facilita o planejamento e implantação de uma política de segurança, que é o elemento mais importante da segurança em redes. Na política de segurança está definido o que se vai proteger, o quanto deve ser protegido, porque e qual a melhor forma.

Por fim, chegou-se a conclusão que, apesar de não ser possível proteger totalmente os servidores, os administradores, utilizando diversas técnicas e tecnologias em diferentes camadas, podem garantir um alto nível de segurança.

REFERÊNCIAS BIBLIOGRÁFICAS

SCAMBRAY, Joel, MCCLURE, Stuart, KURTZ, George. *Hackers Expostos: Segredos e soluções para a segurança de redes*. 2 ed. São Paulo: Makron Books, 2001.

JANG, Michael. *Dominando Red Hat Linux 9*. Rio de Janeiro: Ciência Moderna, 2003.

BERNSTEIN, Terry. *Segurança na Internet***.** Rio de Janeiro: Campus, 1997.

FURMANKIEWICZ, Edson, FIGUEIREDO, Joana. *Segurança máxima: o guia de um hacker para proteger seu site na Internet e sua rede*. Rio de Janeiro: Campus, 2000.

GONCALVES, Marcus: *Firewalls: guia completo.* Rio de Janeiro: Ciência Moderna, 2000.

NAKAMURA, Emilio Tissato, GEUS, Paulo Lício. *Segurança de redes: em ambientes cooperativos*. 2. ed. São Paulo: Futura, 2003.

UCHÔA, Joaquim Quinteiro. *Segurança Computacional.* Lavras: UFLA/FAEPE, 2005. (Curso de Pós Graduação “Latu Sensu” (Especialização) a Distância em Administração em Redes Linux).

NORTHCUTT, Stephen, NOVAK, Judy, MCLACHLAN, Donald. *Segurança e prevenção em redes.* São Paulo: Berkeley, 2001.

WADLOW, Thomas. *Segurança de redes:* projeto e gerenciamento de redes seguras. Rio de Janeiro: Campus, 2000.

MÓDULO *Security Solutions* S.A. *9ª Pesquisa Nacional de Segurança da Informação*, [on-line] outubro 2003. Disponível na Internet via www. Url: http://www.modulo.com.br/pdf/nona_pesquisa_modulo.pdf. Arquivo capturado em 14 mai. 2006.

TEIXEIRA, Roberto, MERCER, Carlos. *Guia do servidor Conectiva Linux*. [on-line] 2004. Disponível na Internet via www. Url: http://www.conectiva.com/doc/livros/online/10.0/servidor/pt_BR/index.html. Arquivo capturado em 09 jun. 2006.

ARRUDA, Felipe, WATTER, Leslie, SZTOLTZ, Lisiane, TEIXEIRA, Roberto. *Guia do servidor Conectiva Linux*. [on-line] 2002. Disponível na Internet via www. Url: http://www.conectiva.com/doc/livros/online/8.0/servidor/book.html. Arquivo capturado em 30 ago. 2006.

FOSTER, Antônio. *Boletim bimestral sobre tecnologia de redes,* [on-line] 1998. Disponível na Internet via www. Url: http://www.rnp.br/newsgen/9811/ipfw.html. Arquivo capturado em 01 jun. 2006.

CSF STORAGE. *Por que implementar cluster.* [on-line]. Disponível na Internet via www. Url: http://www.csfs.com.br/cluster.php. Arquivo capturado em 09 jun. 2006.

NBSO. *Práticas de Segurança para Administradores de Redes Internet.* [on-line] 2003. Disponível na Internet via www. Url: http://www.cert.br/docs/seg-adm-redes/seg-adm-redes.pdf Arquivo capturado em 29/08/2006.

MORGAN, Andrew G. *The Linux-PAM System Administrators' Guide*, Version 0.99.6.0. [on-line] 05/08/2006. Disponível na Internet via www. Url: http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/Linux-PAM_SAG.html. Arquivo capturado em 02 set. 2006.

PERSOAN, Oskar. *The Squid Guide*. [on-line] 2006. Disponível na Internet via www. Url: http://www.deckle.co.za/squid-users-guide/Main_Page. Arquivo capturado em 02 set. 2006.

ANDREASSON, Oskar. *Iptables Tutorial, Version 1.2.0.* [on-line] 2005. Disponível na Internet via www. Url: http://iptables-tutorial.frozentux.net/iptables-tutorial.html. Arquivo capturado em 03 set. 2006.

ROESCH, Marin, GREEN Chris. *Snort Users Manual,* Version 2.6.0. [on-line] 2006. Disponível na Internet via www. Url: http://www.snort.org/docs/snort_htmanuals/htmanual_260/. Arquivo capturado em 03 set. 2006.

LAMPING, Ulf, SHARPE Richard, WARNICKE Ed. *Ethereal User's Guide*. [on-line] 2005. Disponível na Internet via www. Url: http://www.ethereal.com/docs/eug_html/#AppGPL. Arquivo capturado em 02 set. 2006.

TENABLE *Network Security. Nessus 3.0 Client Guide,* Version 13. [on-line] 28/08/2006. Disponível na Internet via www. Url: http://www.nessus.org/documentation/nessus_3.0_client_guide.pdf Arquivo capturado em 02 set. 2006.