UNIVERSIDADE DO EXTREMO SUL CATARINENSE – UNESC

CURSO CIÊNCIA DA COMPUTAÇÃO

MOACYR AZEVEDO COUTO JUNIOR

UM MODELO DE POLÍTICA DE SEGURANÇA: A APLICABILIDADE DA SEGURANÇA DA INFORMAÇÃO EM SERVIDORES LINUX

CRICIÚMA, DEZEMBRO DE 2007

MOACYR AZEVEDO COUTO JUNIOR

UM MODELO DE POLÍTICA DE SEGURANÇA:

A APLICABILIDADE DA SEGURANÇA DA INFORMAÇÃO EM

SERVIDORES LINUX

Trabalho de Conclusão de Curso apresentado para obtenção do Grau de Bacharel em Ciência da Computação da Universidade do Extremo Sul Catarinense.

Orientador: Prof. MSc. Rogério A. Casagrande

CRICIÚMA, DEZEMBRO DE 2007

MOACYR AZEVEDO COUTO JUNIOR

UM MODELO DE POLÍTICA DE SEGURANÇA:
A APLICABILIDADE DA SEGURANÇA DA INFORMAÇÃO EM SERVIDORES LINUX

Submetido ao corpo docente do Departamento de Ciência da Computação da Universidade do Extremo Sul Catarinense como um dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Prof[a] MSc. Ana Cláudia Garcia Barbosa
Coordenadora do Curso de Ciência da Computação

Banca Examinadora:

Prof. MSc. Rogério A. Casagrande (UNESC)
Orientador

Prof. MSc. Paulo João Martins (UNESC)

Prof. Esp. Arildo Sônego (UNESC)

Dedico este trabalho in memorian à Moacyr Azevedo Couto, meu pai, pessoa a que tenho um enorme orgulho, por ter sido um exemplo de pai e amigo, me instruindo, me incentivando, me dando forças em vencer todos os desafios que encontrei em minha vida.

AGRADECIMENTOS

Em especial, agradeço aos meus pais, Moacyr (*in memorian*) e Joêmia, meus irmãos, Amilton e Amílcar, por estarem sempre presentes em minha vida, trazendo-me constantes palavras de amor e incentivo, sem os quais não poderia, jamais, alcançar esta almejada conquista.

Agradeço, também, aos meus grandes amigos, Aramis, Serginho, Andrezinho, Domingos Jorge que, em muitas oportunidades, a despeito da distância de minha terra natal constituíram minha família. Agradeço também minha namorada, Deise, pela compreensão e paciência dispensada nestes momentos finais.

Também sou muito grato a todos os professores da UNIJUÍ, UERJ e UNESC que colaboraram com a concretização dessa vitória por todos esses anos de estudo, transmitindo-me os vossos conhecimentos, e em especial, ao Professor Rogério A. Casagrande pela orientação, compreensão e incentivo na elaboração e conclusão desse trabalho.

Não poderia deixar de agradecer a todos os meus amigos da vida acadêmica e naqueles, também, que de alguma forma, participaram em algum momento da minha vida, incentivando-me e não me deixando abater nas horas difíceis.

E principalmente a DEUS por ter me abençoado, me dando forças pra prosseguir pois sem ele ao meu lado este momento não poderia tornar possível.

“Entrega o teu caminho ao Senhor, confie nele e o mais ele fará.”

Salmos 37:5

RESUMO

A segurança das informações tornou-se um ponto crucial para a sobrevivência de qualquer organização e deve ser tratada como um dos seus objetivos prioritários. Fruto dessa realidade, verifica-se a necessidade de estabelecer princípios de segurança para se contrapor as inúmeras ameaças existentes. Por meio do estudo dos padrões e normas nacionais e internacionais buscou-se atender os requisitos básicos de segurança na proteção da informação, identificando que a segurança da informação envolve a aplicação de procedimentos em diversas áreas distintas de uma organização. Após entender os riscos a que um servidor está sujeito e conhecer técnicas e tecnologias que podem ser empregadas para contribuir com a segurança, pode-se ter uma visão mais ampla sobre as necessidades de segurança de uma organização, facilitando assim o planejamento e implementação de uma política de segurança coerente. Esse trabalho propõe definir uma metodologia de política de segurança para servidores Linux visando obter um nível de segurança adequado.

Palavras-Chaves: Segurança da Informação, Política de Segurança, Servidores Linux.

ABSTRACT

The information security has become very important to the survival of any organization and should be treated as one of its priority objectives. This reality, there is a need to establish principles of security to counter the many threats.This model will be based security norms and standards of national and international, searching to take care of to the security basic requirements of a company in the protection of information, identifying that the information security involves the application of procedures in areas distinct of an organization. After understand the risks to which a server is subject and learn techniques and technologies that can be employed to contribute to the security, can have a broader view on requirements of an organization security, facilitating the planning and implementation of a consistent security policy. The objective of this project, is to provide an approach of methodology implementation of model security policies of Linux server.

Keywords: Information Security, Security Policies, Linux server

LISTA DE ILUSTRAÇÕES

Figura 1. Incidentes reportados ao CERT.br 24

Figura 2. Scan reportados ao CERT.br – Julho a Setembro de 2007 27

Figura 3. Exemplo de um *firewall* 31

Figura 4. Distribuições Linux baseadas em outras distribuições 34

Figura 5. Política de segurança de informações e seus relacionamentos 48

Figura 6. Modelo da cebola para a segurança física 50

Figura 7. Entradas no *logbook* 68

Figura 8. Criação de senha usando acrônico 80

LISTA DE TABELAS

Tabela 1. Recursos a serem protegidos 21

Tabela 2. As 10 distribuições Linux mais utilizadas..... 34

Tabela 3. *Filesystems* mais comuns para o GNU/ Linux 36

Tabela 4. Função específica dos diretórios Linux..... 37

Tabela 5. Configuração das partições no arquivo /etc/fstab 71

Tabela 6. Ferramentas para alterar scripts de inicialização..... 75

Tabela 7. Módulos de autenticação alteráveis 88

Tabela 8. Configuração mais segura para o ssh 90

Tabela 9. Medidas de prevenção nos sistemas de suporte e infra-estrutura a TI..... 97

Tabela 10. Componentes de um sistema de informação 98

Tabela 11. Recursos disponíveis no syslogd 112

Tabela 12. Principais arquivos de log 113

Tabela 13. Periodicidade dos testes 116

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
BSI	British Standards Institute
CGI	Common Gateway Interface
CLT	Consolidação das Leis de Trabalho
DNS	*Domain Name Services*
FTP	*File Transfer Protocol*
HIDS	*Host Intrusion Detection System*
HTTP	*HiperText Transfer Protocol*
IDS	Sistema de Detecção de intrusão
ISO	Internacional Organization for Standardization
IP	*Internet Protocol*
MUA	Agente de Usuário de Mensagem
MTA	Agente de Transferência de Mensagens
NIDS	*Network Intrusion Detection System*
SGSI	Sistema de Gestão da Segurança da Informação
SSH	Security Shell
SSL	Security Sockets Layer
SO	Sistema Operacional
STMP	*Simple Mail Transfer Protocol*
TCP	*Transfer Control Protocol*
TI	Tecnologia da Informação

SUMÁRIO

1 INTRODUÇÃO 15

1.1 OBJETIVO GERAL 16

1.2 OBJETIVOS ESPECIFICOS 16

1.3 JUSTIFICATIVA 16

1.4 ESTRUTURA DO TRABALHO 18

2 SEGURANÇA COMPUTACIONAL 20

2.1 CONCEITOS GERAIS DE SEGURANÇA 20

2.2 OBJETIVOS DA SEGURANÇA COMPUTACIONAL 21

2.2.1 **Confidencialidade** 21

2.2.2 **Integridade de Dados** 22

2.2.3 **Disponibilidade** 22

2.2.4 **Autenticidade** 22

2.3 ANÁLISE DE RISCOS 23

2.4 AMEAÇAS, VULNERABILIDADES E ATAQUES 23

2.4.1 **Vírus e *Worms*** 25

2.4.2 ***Trojans e Backdoors*** 25

2.4.3 ***Spyware*** 26

2.4.4 **Ferramentas de Exploração de Vulnerabilidades** 26

2.4.5 **Ferramentas de Negação de Serviços** 28

2.5 MEDIDAS DE SEGURANÇA 28

2.5.1 **Introdução à Política de Segurança** 29

2.5.2 **Criptografia** 29

2.5.3 **Sistema de Detecção de Intrusão** 30

2.5.4 ***Firewall*** 30

2.5.5 **Análise de Registros de *Log*** 31

3 SISTEMAS OPERACIONAIS DE REDE 32

3.1 SISTEMAS OPERACIONAIS UNIX 32

3.2 SISTEMA OPERACIONAL LINUX 33

3.3 PRINCIPAIS VULNERABILIDADES DO LINUX 38

3.3.1 **Senhas** 38

3.3.2 ***Buffers*** 39

3.3.3 **Serviços Desnecessários e Portas Abertas** 39

3.3.4 **Serviços Desatualizados** 40

3.3.5 **Serviços Essencialmente Inseguros** 41

3.4 SERVIDORES *LINUX* 41

3.3.1 **Serviço de Páginas *Web*** 42

3.4.2 **Serviço de Correio Eletrônico** 43

3.4.3 **Serviço de FTP** 44

3.4.4 **Serviço de Acesso Remoto** 45

3.4.5 **Serviço DNS** 46

4 POLÍTICAS DE SEGURANÇA 47

4.1 DEFINIÇÃO DE UMA POLÍTICA DE SEGURANÇA 47

4.1.1 **Segurança Física** 50

4.1.2 **Segurança Lógica** 51

4.2 IMPLANTAÇÃO DA POLÍTICA DE SEGURANÇA 51

4.3 NORMAS E PADRÕES DE SEGURANÇA 52

4.3.1 **BS 7799** 53

4.3.2 **NBR ISO/IEC 17799** 54

4.3.3 **NBR ISO/IEC 27000** 57

5 MODELO DE POLÍTICA DE SEGURANÇA 58

5.1 SEGURANÇA EM RECURSOS HUMANOS 59

5.2 CONTROLES DE AMBIENTE FÍSICO E DO MEIO AMBIENTE 61

5.2.1 **Segurança de Equipamentos** 62

5.2.2 **Controle de Acesso Físico** 64

5.2.3 **Proteção contra Ameaças Externas e Meio Ambiente** 65

5.3 CONTROLES NA INSTALAÇÃO E IMPLANTAÇÃO DO SERVIDOR 67

5.3.1 **Segurança no Sistema de Arquivos (particionamento)** 69

5.3.2 **Instalação Mínima** 71

5.3.3 **Desabilitar Serviços Desnecessários** 72

5.3.4 **Instalações de Correções de Segurança** 75

5.3.5 **Configuração dos Serviços Utilizados** 76

5.4 POLÍTICA DE CONTROLE DE ACESSO AO SERVIDOR 78

5.4.1 **Política de Uso de Senha** 79

5.4.2 **Política de Gerenciamento de Usuários** 81

5.4.3 **Gerenciamento de Privilégios e Permissões de Acesso** 83

5.4.4 **Controle de Acesso ao Sistema Operacional** 86

5.4.5 **Política de Acesso Remoto** 88

5.5 CONTROLES DE PRESERVAÇÃO DA DISPONIBILIDADE DA INFORMAÇÃO 91

5.5.1 **Política de *Backup*** 92

5.5.2 **Plano de Contigência** 94

5.6 CONTROLES DE MECANISMOS DE PROTEÇÃO AO SERVIDOR 100

5.6.1 **Utilização de *Firewall*** 101

5.6.2 **Utilização de IDS** 104

5.6.3 **Utilização de Antivírus e Anti-r*ootkits*** 106

5.6.4 **Utilização de Controles Criptográficos** 107

5.7 CONTROLES DE MONITORAMENTO, AUDITORIA E TESTE 109

5.7.1 **Política de Monitoramento** 110

5.7.2 **Política de Auditoria e Testes** 114

CONCLUSÃO 117

REFERÊNCIAS 120

APÊNDICE A – COMANDOS E ARQUIVOS DE CONFIGURAÇÃO DO LINUX 127

APÊNDICE B – TUTORIAL DE SEGURANÇA PARA SERVIDORES (CHECK LIST) 131

APÊNDICE C – RESUMO DA APLICAÇÃO DA NBR ISO/IEC 17799 EM SERVIDORES 134

APÊNDICE D – ANÁLISE PRÁTICA 136

APÊNDICE E – ARTIGO CIENTÍFICO 148

1 INTRODUÇÃO

Atualmente as informações contidas em sistemas computacionais são consideradas recursos críticos em qualquer organização. Com isso, a segurança das informações tornou-se um requisito de elevada importância com vista a manter a constante continuidade dos negócios de tais instituições.

Apesar da possibilidade não ser remota de dados importantes serem capturados, observa-se que em muitas instituições não há sequer uma política de segurança que visa à minimização e ao combate a possíveis ameaças, ou mesmo, se existe, a sua aplicação pode não ser executada por todos os agentes responsáveis (usuários e técnicos de informática).

Paralelo a questões de segurança, máquinas que utilizam o Sistema Operacional (SO) Linux estão sendo empregadas, cada vez mais, em organizações, públicas e privadas, como estações de trabalho e principalmente em servidores, distribuindo-se serviços de rede, correio, *web*, dentre outros.

Do exposto, a problemática em questão baseia-se na necessidade de elaborar uma política de segurança específica para ambientes que utilizam servidores Linux. É de vital importância que toda instituição tenha uma política de segurança eficiente de modo a evitar a manipulação não autorizada e a reduzir a probabilidade e o impacto de incidentes de segurança.

1.1 OBJETIVO GERAL

Propor uma política de segurança de informações específica para servidores *Linux*.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos dessa pesquisa consistem em:

a) identificar os principais conceitos de segurança computacional;

b) identificar as principais normas nacionais e internacionais utilizadas como referência na segurança da informação.

c) identificar os principais serviços disponibilizados nos servidores Linux;

d) enunciar as principais ameaças e vulnerabilidades do Linux;

e) descrever os recursos de segurança dos servidores Linux aplicando-os na elaboração da política de segurança sugerida;

f) propor uma política de segurança da informação que atenda aos requisitos básicos de segurança de uma instituição que utilize servidores Linux.

1.3 JUSTIFICATIVA

As ameaças e as vulnerabilidades, relativas ao emprego e ao acesso às informações, devem ser adequadamente consideradas no contexto da informatização de atividades e processos. A eficiência dos recursos de tecnologia na proteção dos dados, constitui fator primordial para o sucesso de qualquer organização.

Mello (2005) destaca que instituições têm como excelentes alternativas em seus projetos a utilização do SO Linux e podem usá-los confiando em sua qualidade, capacidade de segurança e funcionalidade. Sua aplicabilidade é variada podendo servir como estações de trabalho e servidores. Acrescenta, ainda, que ao se pensar na segurança do sistema deve ser buscada a seguinte fórmula: “capacitação (técnicos, usuários) + metodologia (normas, política de segurança) + ferramental = bom projeto de segurança”. As duas primeiras “variáveis” serão os objetos principais de estudo do presente trabalho.

Uma política de segurança de informações deve estabelecer princípios institucionais de como a organização irá se proteger, controlar e monitorar seus recursos computacionais e, conseqüentemente, as informações por eles manipuladas (DIAS, 2000).

Barenboim (2003) relata que o objetivo básico da política de segurança é informar aos usuários o que eles devem fazer com vista a proteger os recursos tecnológicos e de informação de uma organização. Deve deixar claro o mecanismo por meio do qual isso pode ser encontrado, além de explicar como configurar, manter e auditar sistemas para que essa política possa se manter consistente.

Apesar de todo o esforço que possa ser despendido em segurança lógica, limitando acessos e protegendo os dados, a política de segurança deve constar, também, os aspectos relacionados à segurança física.

Portanto, justifica-se o estudo das principais ameaças e vulnerabilidades dos servidores, pois é essencial que se tenha o conhecimento por parte dos administradores de rede de que forma os usuários mal intencionados encontram, examinam e obtêm acesso ao sistema. Apesar dos ataques estarem cada vez mais

sofisticados, há forma de evitá-los, basta conhecer os devidos procedimentos técnicos (STARLIN; NOVO, 2002).

Apesar da diversidade de literatura, não é tão simples encontrar metodologia coerente e básica ou mesmo, um conjunto de diretrizes que auxilie na elaboração de uma política de segurança específica para ambientes que utilizam servidores Linux.

Os contextos que norteiam o tema em questão justificam a motivação da presente pesquisa baseada na possibilidade de se estabelecer uma política eficiente de segurança com o objetivo de preservar ao máximo as informações de uma organização e a continuidade do negócio.

Os controles de segurança podem ser considerados como princípios básicos, fornecendo um bom ponto de partida para a implementação da segurança da informação. São baseados tanto em requisitos legais como nas melhores práticas de segurança da informação normalmente usadas (CACIATO, 2004).

O estudo de normas e padrões, tais como a NBR ISO/IEC 17799, permite fornecer recomendações para a gestão da segurança da informação em sua implementação ou manutenção da segurança das organizações. Portanto, em essência, essas questões são a base para a teoria da segurança, que apresentará os recursos de segurança básicos para ambientes Linux em uma política de segurança coerente e prática.

1.4 ESTRUTURA DO TRABALHO

A presente pesquisa tem como meta elaborar uma política de segurança que se aplica em servidores Linux. Para tanto, o trabalho é dividido em duas partes: a primeira parte, composta pelos Capítulos 2, 3 e 4 formam a base teórica do objeto de

estudo. A segunda parte, consubstanciada em variadas referências bibliográficas, apresenta o cerne do trabalho descrevendo um modelo de política de segurança.

O Capítulo 2 identifica os principais fundamentos relacionados à segurança computacional, abordando seus objetivos, suas principais ameaças e formas de proteção. O Capítulo 3 tem por objetivo descrever algumas características existentes nos sistemas operacionais de rede, em especial o *Linux*, a fim de auxiliar e compreender melhor a sua aplicação em servidores. O capítulo seguinte enumera os principais aspectos que compõe uma Política de Segurança, citando as normas e padrões de segurança utilizados como referência na segurança da informação. O Capítulo 5, em conjunto com o Apêndice A – Comandos do Linux, apresenta os diversos controles implementados na presente política como: segurança em recursos humanos, controles do ambiente físico e do meio ambiente, controles na instalação e implementação do servidor, controles de acesso ao servidor, controles de preservação da integridade e disponibilidade da informação, controles de mecanismos de proteção ao servidor e controles de monitoramento, auditoria e teste. O Apêndice B – Tutorial de Segurança para Linux apresenta um resumo sintético da presente pesquisa, enquanto que o Apêndice C – Aplicação da NBR ISO/IEC 17799 em servidores descreve uma síntese sobre os itens constantes da referente norma que correlacionam com a segurança dos servidores. E complementando o trabalho, o Apêndice D – Análise Prática, tem por objetivo correlacionar a política de segurança proposta com diferentes tipos de ataques em um ambiente de rede simulado, visando obter resultados práticos da metodologia sugerida.

2 SEGURANÇA COMPUTACIONAL

Os dados existentes em diferentes sistemas são considerados recursos críticos, tanto para instituições privadas, bem como, para órgãos públicos e, portanto necessitam de procedimentos e recursos de segurança com o objetivo de se preservarem ao máximo o valor das informações. Os recursos computacionais e as informações sobre as organizações, por possuírem um alto valor agregado, podem ser vítima de ações de pessoas ou grupos com más intenções.

Este capítulo destaca os principais aspectos relacionados aos conceitos de segurança computacional, abordando os seus objetivos, suas principais ameaças e formas de proteção.

2.1 CONCEITOS GERAIS DE SEGURANÇA

Segurança, em seu sentido mais amplo, é a capacidade de se proteger contra alguém ou algo. Segundo Ferreira (2006) segurança é o ato ou efeito de segurar, ou seja, é a condição livre de perigo ou risco.

A segurança computacional está relacionada à proteção de informações, sistemas e recursos computacionais contra erros ou manipulação não-autorizada, de forma a reduzir a probabilidade e os impactos de incidentes de segurança (DIAS, 2000).

Com as devidas circunstâncias, os *bugs*[1] de *software*, acidentes, erros, má sorte ou um usuário mal intencionado implicará que qualquer computador poderá ficar comprometido, submetido a desuso ou algo pior do que isso (CORDEIRO; MOREIRA, 2002).

[1] Erro em um programa de computador que o faz executar incorretamente.

Na análise da segurança de qualquer instituição é necessário, portanto, identificar todos os recursos que devem ser protegidos e onde as informações importantes estão sendo armazenadas. A Tabela 1 apresenta os recursos que mais necessitam de cuidados especiais na proteção.

Tabela 1. Recursos a serem protegidos

AMBIENTE	RECURSOS
Hardware	Processadores, placas, teclados, terminais, estações de trabalho, computadores pessoais, impressoras, unidades de disco, linhas de comunicação, servidores, roteadores.
Software	Utilitários, programas diagnósticos, sistemas operacionais, programas de comunicação, aplicativos.
Dados	Em processamento, em trânsito nos dispositivos e linhas de comunicação, armazenados *on-line* e *off-line*, cópias de segurança, registros de auditoria, bases de dados.
Pessoas	Usuários e funcionários necessários para o funcionamento dos sistemas.
Documentação	Sobre programas, *hardware*, sistemas, procedimentos administrativos.
Suprimentos	Papel, formulários, fitas, disquetes, CD-ROM.

Fonte: DIAS, C. (2000)

2.2 OBJETIVOS DA SEGURANÇA COMPUTACIONAL

O anseio dos usuários da informática está relacionado à que suas informações sejam confiáveis, corretas e mantidas em local seguro e que não haja acesso aos dados por pessoas não autorizadas. Tais expectativas, segundo Candéa (2002) se traduzem nos seguintes objetivos de segurança: confidencialidade, integridade de dados, disponibilidade e autenticidade.

2.2.1 Confidencialidade

Consiste em proteger as informações contra leitura ou cópia por alguém que

não tenha sido autorizado pelo proprietário daquela informação. Deve-se atentar não somente com a proteção da informação como um todo, mas também de partes da informação que possam interferir sobre o todo (SPANCESKI, 2004).

2.2.2 Integridade de dados

Evitar que os dados sejam modificados ou apagados sem a devida autorização, ou seja, é a capacidade se manter uma informação em sua forma original. Bernardes (1999), acrescenta que é a garantia que os dados armazenados não serão modificados, tanto como conseqüência de atos provenientes de uma intrusão quanto a eventos como quedas de energia e falhas nos sistemas.

2.2.3 Disponibilidade

Proteção dos serviços para que não sejam degradados ou indisponíveis sem autorização. A segurança consiste em prevenir que ataques deliberados ou maliciosos evitem ou dificultem o acesso de usuários a seus próprios sistemas (CORDEIRO; MOREIRA, 2002).

2.2.4 Autenticidade

É a garantia de que as informações são realmente procedentes da origem informada em seu conteúdo. Está associada com a identificação correta do usuário ou do computador (CANDÉA, 2002).

2.3 ANÁLISE DE RISCOS

Risco, quando aplicado à tecnologia da informação, é um evento ou condição incerta que, se acontecer, trará um efeito para a segurança da informação (FRANCISCO, 2004).

O processo de análise de riscos apresenta um estudo das vulnerabilidades existentes, analisando-se as ameaças, os mecanismos de segurança existentes e o impacto resultante para a organização. Fruto dessa análise, recomenda-se que a instituição crie ou modifique os mecanismos de segurança existentes (MARTINS, 2003).

2.4 AMEAÇAS, VULNERABILIDADES E ATAQUES

Uma ameaça é caracterizada por uma possível ação, que se realizada, ocasionará possíveis alterações nos dados de um sistema. Essas violações de segurança ocorrem devido à exploração das vulnerabilidades existentes nos sistemas.

> As vulnerabilidades em sistemas computacionais sempre estiveram presentes. Um erro de programação, um erro de configuração ou mesmo um erro de operação, podem permitir que usuários não autorizados entrem no sistema ou mesmo que usuários autênticos executem ações não autorizadas, podendo assim comprometer o funcionamento correto do sistema (CAMARGO, 2000, p. 23).

O ataque, portanto, é a efetivação de uma vulnerabilidade por meio de uma ameaça. Enquanto, as ameaças estão relacionadas geralmente a usuário mal intencionados, podendo ser internamente, oriundas de funcionários ou ex-funcionários insatisfeitos, ou externamente por meio de *hackers* e *crackers*[2].

As invasões aos sistemas podem ser executadas por meio da exploração de

[2] *Hacker é* uma pessoa interessada em descobrir novas brechas dentro dos sistemas, estão sempre buscando novas descobertas e jamais corrompem dados. Os *crackers*, por sua vez, invadem sistemas e destroem ou roubam os dados, suas ações são sempre maliciosas.

técnicas que podem ter como base a engenharia social[3] ou invasões técnicas. Essas invasões exploram deficiências na concepção, implementação, configuração, gerenciamento dos serviços e sistemas, e por acompanharem a evolução tecnológica continuam em permanente evolução (RAMOS, 2005).

A cada dia, listas de discussão, como security@debian.org e *sites* especializados, como o *site* www.securityfocus.com, publicam novas falhas encontradas em sistemas operacionais e programas (PEÑA, 2007).

Dependendo das ações do usuário mal intencionado, as seguintes subcategorias, podem ser usadas para descrever a maioria dos ataques à segurança: interceptação, análise de tráfego, personificação, modificação, inserção, retransmissão e negação de serviços (CARLOS NETO, 2004). A cada dia segundo o Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil (CERT.br), órgão responsável por receber, analisar e responder a incidentes de segurança envolvendo redes conectadas à I*nternet* no Brasil, o número de incidentes vem aumentando. A Figura 1 apresenta o total de incidentes reportados ao CERT.br desde 1999 a setembro de 2007.

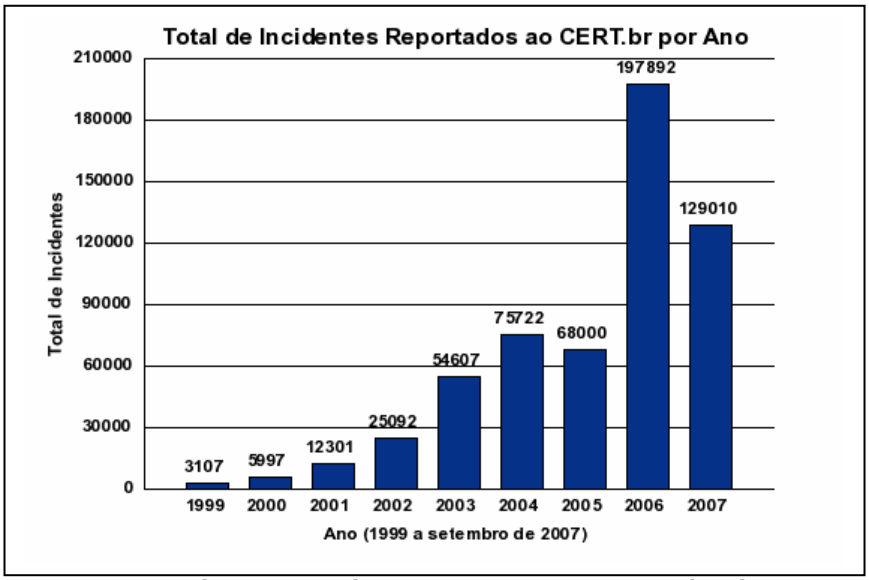

Figura 1. Incidentes reportados ao CERT.Br – 1999 a setembro de 2007
Fonte: CERT.Br (2007)

[3] É o emprego de técnicas de aquisição de informações vitais sobre uma determinada pessoa, empresa, produto ou organização.

Os principais ataques, aos sistemas de informação, descritas por Thomas (2005) são: vírus e *worms*, *trojans* e *backdoors*, *spywares*, ferramentas de exploração de vulnerabilidades e ferramentas de negação de serviços.

2.4.1 Vírus e *Worms*

Os vírus são seções de códigos nocivos, que modificam programas originais por meio da inserção de código nos arquivos afetados. Vírus podem se propagar por meio dos recursos computacionais pela execução de seus programas hospedeiros, afetando assim novos alvos. Uma classe de vírus são os *rootkits* que *comprometem a segurança do Linux,* servem-se para se obter acesso administrativo a um *host*[4], além de conter diversas ferramentas e aplicativos, para serem usadas em uma invasão. Os *worms* (vermes) se diferem de vírus por serem programas completos, executáveis independentemente da existência de um hospedeiro (HIJAZI; MAZZORANA; RAVANELLO, 2004).

2.4.2 Trojans e Backdoors

Os *trojans*, popularmente conhecidos como cavalos-de-tróia, geralmente, usam programas que, camuflam a sua intenção, sendo aparentemente agradáveis para o usuário, como um jogo, uma proteção de tela ou um reprodutor de sons em mp3, entre outros exemplos. Entretanto, ao contrário de programas sérios, realiza uma função não-autorizada de forma oculta. A diferença entre os *backdoors* é que estes não inseridos sem o consentimento do usuário (HATCH; LEE; KURTZ, 2002).

[4] É qualquer máquina ou computador conectado a uma rede. Os hots variam de computadores pessoais a supercomputadores.

Tais programas abrem uma comunicação externa por meio de uma determinada porta, fornecendo acesso remoto a todos que possuam o "cliente", que é o programa usado para se conectar ao "servidor" (usuário invadido) (LOPES, PEREIRA; SILVA FILHO, 2002).

2.4.3 *Spyware*

Segundo Beal (2004 apud THOMAS, 2005) os s*pywares* exploram falhas de alguma aplicação ou do próprio sistema operacional, geralmente se instalando em *browsers*[5] buscando monitorar toda a atividade realizada na *Internet*, capturando endereços de *e-mail*, *sites* que estão sendo visitados, senhas, e até número de cartões de crédito, dentre outras informações.

2.4.4 Ferramentas de Exploração de Vulnerabilidades

Existem inúmeros programas que permitem explorar as vulnerabilidades e que auxiliam os usuários mal intencionados no ataque a determinados sistemas, essas ferramentas possuem metodologias e recursos diferentes. É de grande importância para um administrador de sistema conhecê-las.

O ***scanner*** é uma ferramenta que detecta vulnerabilidades, podendo ser utilizada com o objetivo de analisar um *host* local ou em varredura de rede. Segundo Monteiro (2005) atualmente existem *scanners* que identificam tipos de sistemas operacionais, portas abertas, versões de *software*, vulnerabilidade e até a localização de onde encontrar informações sobre vulnerabilidades.

[5] É um programa que é usado para pesquisar e consultar informação disponibilizada em *sites*. Em língua portuguesa, o significado mais preciso é o de navegador.

A Figura 2 apresenta uma estatística das principais portas analisadas por scanners em servidores reportados ao CERT.br no terceiro trimestre de 2007.

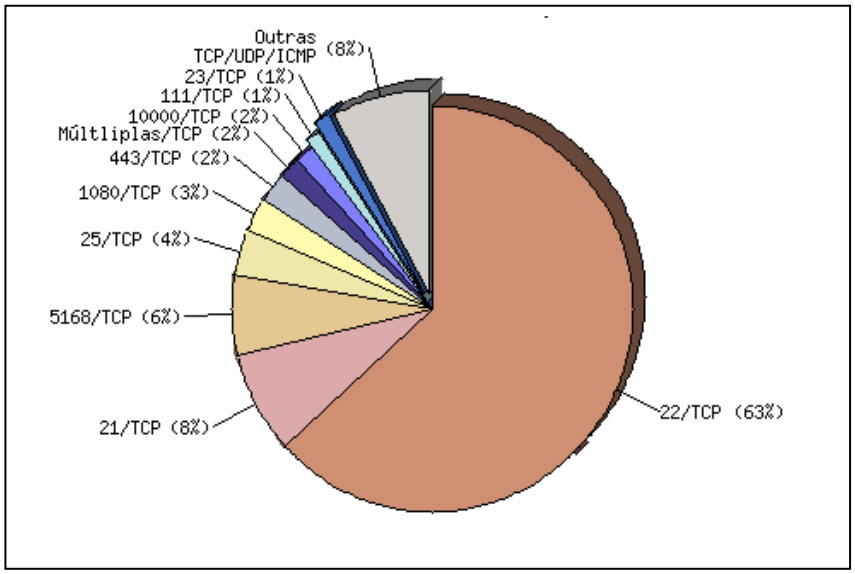

Figura 2. Scan reportados ao CERT.br – Julho a Setembro de 2007
Fonte: CERT.br (2007)

Dentre os diversos tipos de *scanners* de porta, pode-se citar: o Cheops que permite localizar as máquinas da rede, desenhá-las em uma área de trabalho identificando o SO, o Nmap que permite selecionar diversas opções que combinem com os critérios de busca e o Nessus que possui a possibilidade de permanente atualização por meio da instalação de novos *plugins* de vulnerabilidades (MONTEIRO, 2003).

Além dos *scanners*, existem os ***exploits***, que em inglês, significa um ato ou tarefa especial, realizado de modo brilhante ou heróico. No contexto da Segurança da Informação, os *exploits* são programas utilizados para explorar vulnerabilidades de segurança em programas ou sistema operacional específicos (SANTOS, 2004).

Apesar de não explorar diretamente uma vulnerabilidade, os ***spoofings*** são ferramentas que permitem que usuários mal intencionados modifiquem seus endereços de origem, fornecendo o de uma outra máquina confiável como sendo o seu próprio endereço. Tal ferramenta ter por finalidade dificultar a perícia forense[6].

[6] Algumas metodologias utilizadas por peritos para obtenção de evidências de invasão.

Na atualidade, essa definição foi expandida para abranger qualquer método de subverter a relação de confiança ou autenticação baseada em endereço ou em nome de *host* (THOMAS, 2005). Existem diversas técnicas de *spoofing*, tais como: *spoofing* de IP, de ARP e de DNS (SEGURANÇA MÁXIMA PARA LINUX, 2000).

Os ***sniffers***, comumente chamados de farejadores, são ferramentas que capturam pacotes de rede, independentemente do destino legítimo que eles possam ter, com o objetivo de analisar o tráfego de rede. Della Valle e Ulbrich (2003) acrescentam que essas ferramentas podem trazer perigo para a rede, colhendo senhas de usuários e informações dos serviços prestados. Vianna (2004) enumera os principais *sniffers* utilizados: Tcpdump, Ethereal, Ethercap, Dsniff e Spy.

2.4.5 Ferramentas de Negação de Serviços

O ataque de negação de serviço, ou *Denial of Service* (DoS), é um ataque que explora vulnerabilidade do TCP/IP[7] e da implementação do mesmo pelos sistemas operacionais, com a finalidade de tornar algum serviço ou informação indisponível. Uma variante deste ataque é o *Distributed of Service* (DDoS) que utiliza vários computadores para executar o ataque simultâneo a um único objetivo (MARTINS, 2003).

2.5 MEDIDAS DE SEGURANÇA

Nos sistemas computacionais as ameaças são constantes e a forma de se evitar ou de se dificultar os ataques é identificar e corrigir as vulnerabilidades existentes

[7] Protocolo que provê recursos para a transmissão de mensagens entre equipamentos dentro de uma ou mais redes (NEMETH, 2002).

nos sistemas, algo que na verdade não é tão simples.

Nesse contexto, diversos mecanismos de segurança são realizados para a prevenção dos ataques. Spanceski (2004) enumera que desde procedimentos físicos, como impedir a entrada de usuários em uma determinada sala; treinamento e conscientização dos funcionários ou até realização de *backups*[8] periódicos ou políticas de controle de acesso, são ações implantadas para se chegar a tal finalidade.

Na maioria das vezes, deve-se usar a combinação de várias estratégias de acordo com o nível de segurança que se deseja alcançar. Algumas medidas que podem vir a serem adotadas são: estabelecimento de políticas de segurança, uso de criptografia, utilização de Sistema de Detecção de Intrusão (IDS), uso de *firewall*[9], e análise de *log*[10].

2.5.1 Introdução à Política de Segurança

É um documento publicado que estabelece estratégias e práticas de segurança que irão buscar há confiabilidade e integridade nos sistemas de informação. Tais políticas permitem às organizações fixarem práticas e procedimentos a fim de reduzir a probabilidade de incidentes, bem como minimizar um dano causado pelo mesmo, caso esse venha acontecer (THOMAS, 2005).

2.5.2 Criptografia

Criptografia significa transformar uma mensagem em outra (escondendo a mensagem original), com a elaboração de um algoritmo com funções matemáticas e

[8] Cópia de segurança de um arquivo ou sistema.
[9] São dispositivos que possuem a função de controlar o fluxo de tráfego entre uma rede interna e uma rede externa.
[10] São registros realizados por um sistema operacional ou programa das atividades executadas.

uma senha especial, denominada chave (KON; PINHEIRO JUNIOR, 2004). A criptografia surgiu da necessidade de se enviar informações com caráter sigiloso por meio de sistemas de comunicações não confiáveis.

2.5.3 Sistema de Detecção de Intrusão

São ferramentas úteis que vão monitorar a rede e servidores, analisando tudo o que acontece, qual tipo de tráfego está circulando e que eventos ocorrem dentro da rede ou no próprio *host*.

Os IDS não apenas coletam e sincronizam registros desses eventos, mas também fazem uma análise dos mesmos procurando por sinais de violações de segurança (CORDEIRO; MOREIRA, 2002). Os sistemas de detecção de intrusão serão estudados com mais detalhes no item 5.6.2.

2.5.4 *Firewall*

Um *firewall* é qualquer dispositivo projetado para impedir que estranhos acessem sua rede. Tal dispositivo, segundo o Livro Segurança Máxima (2000), é geralmente um computador independente, um roteador ou um *firewall* em uma caixa.

> *Firewall* são dispositivos de *hardware* e *software* que servem para criar uma barreira de proteção lógica entre a sua rede e uma rede externa com regras para determinar o que pode ou não passar por eles. Dito em outras palavras, os *Firewalls* compõem o portão de entrada da sua empresa com um porteiro inspecionando tudo o que entra e saí, e tomando as devidas providências sobre o tipo de tráfego. (MONTEIRO, 2003, p. 51).

Ribeiro (2004) define que um *firewall* é como se fosse uma “parede a prova de fogo” que protege uma máquina ou uma rede local contra o restante do exterior da rede. A Figura 3 exemplifica a sua localização demonstrando essa definição.

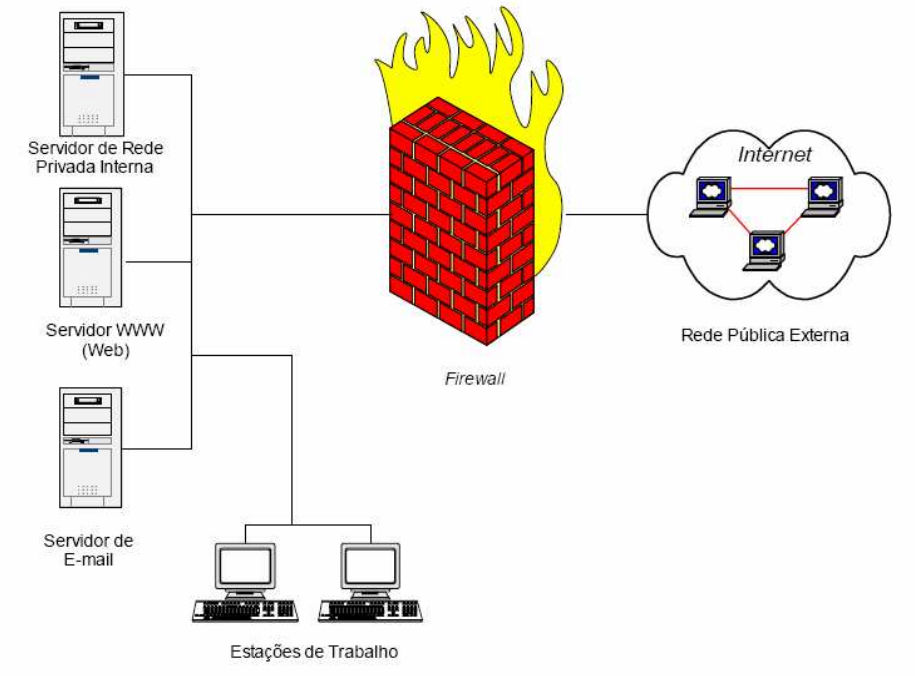

Figura 3. Exemplo de um *firewall*
Fonte: RIBEIRO, S. (2003)

A aplicação do *firewall* na segurança do servidor será objeto de estudo do item 5.6.1.

2.5.5 Análise de Registro de *log*

Esse é um excelente recurso, pois permite que as informações constantes em um arquivo de *log* possam ser editadas e pesquisadas com facilidade. Um eventual sistema de análise usaria essa base de registros do sistema comparando-a com padrões pré-estabelecidos com o objetivo de detectar anomalias nos sistemas utilizados (KLER; PADRO, 2004).

Na arquitetura de segurança do Linux, o sistema de *log* é vital para a análise do comportamento desse SO, por esse motivo será abordado, com mais afinco no item 5.7.1.

3. SISTEMAS OPERACIONAIS DE REDES

Com o advento da *Internet*, surge a necessidade da utilização de Sistemas Operacionais que apresentam eficientes recursos de rede e segurança. Monteiro (2003) explica que SO é um conjunto de programas que gerenciam as funções internas do computador e permitem que o usuário controle a sua operação. Silva (2006, p. 5) complementa: “Ele é responsável pelo gerenciamento de recursos e periféricos (como memória, discos, arquivos, impressoras, CD-ROMs, entre outros), interpretação de mensagens e execução de programas”.

O presente capítulo pretende tratar algumas características existentes nos sistemas operacionais de rede, em especial o Linux, a fim de auxiliar a compreender melhor a sua utilidade. Os conhecimentos relacionados a esse assunto são de grande importância para o entendimento final do trabalho.

3.1 SISTEMAS OPERACIONAIS UNIX

A Bell Laboratories em parceria com a AT&T, começou em 1969 um novo projeto de sistema operacional, denominado UNIX. Esse SO, segundo Monteiro (2003), surgiu como um ambiente de programação para atender técnicos, programadores, engenheiros e cientistas.

Em 1977 o UNIX foi lançado como sistema operacional comercial e sendo distribuído também de forma gratuita para instituições de pesquisa, faculdades e universidades, o que permitiu que fosse estudado e aperfeiçoado (MONTEIRO, 2003).

Sendo um padrão aberto, o *UNIX*, conhecido também como *Single Unix Specification* permitiu que diferentes fabricantes desenvolvessem suas próprias

implementações. Nesse Contexto, Della Valle e Ulbrich (2003) citam alguns desenvolvedores que possuem diferentes versões do sistema:

a) *Sun Microsystems*, com seu Solaris;

b) IBM, com o *AIX*;

c) SGI e o *IRIX*;

d) BsDi com a implementação BSD e a versão gratuita FreeBSD;

e) *Hewlett-Packard* e seu HP-UX;

f) própria *Microsoft* com o finado *Xenix*; e

g) família LINUX, cerne do estudo desse trabalho.

O Linux, motivado pelo estudo do sistema operacional Minix, surgiu em 1991, sendo desenvolvido por Linus Torvalds, na Universidade de Helsinki na Finlândia (SILVA, 2006), Segurança Máxima para Linux (2000) acrescenta que desde então o GNU/Linux cresce e vem tornando um sistema operacional completo sendo utilizado cada vez mais em diferentes ambientes.

3.2 SISTEMA OPERACIONAL LINUX

Na essência, Linux, por si só, é o *Kernel*[11]. Esta junção entre o *Kernel* e as mais diversas ferramentas é denominada distribuição GNU/Linux. Para fins didáticos no decorrer dessa pesquisa será usado o termo Linux para representar o Sistema Operacional. Mota Filho (2006) descreve que as distribuições *Debian*, *Fedora* e *Slackware* são as maiores e mais antigas e que diversas distribuições são derivadas dessas. A Figura 4 apresenta uma pesquisa realizada pelo Site DistroWatch (http://distrowatch.com) realizada em maio de 2007.

[11] Rotinas de instruções básicas, essenciais, necessárias como uma base para quaisquer operações em um sistema de computador (MOTA FILHO, 2006)

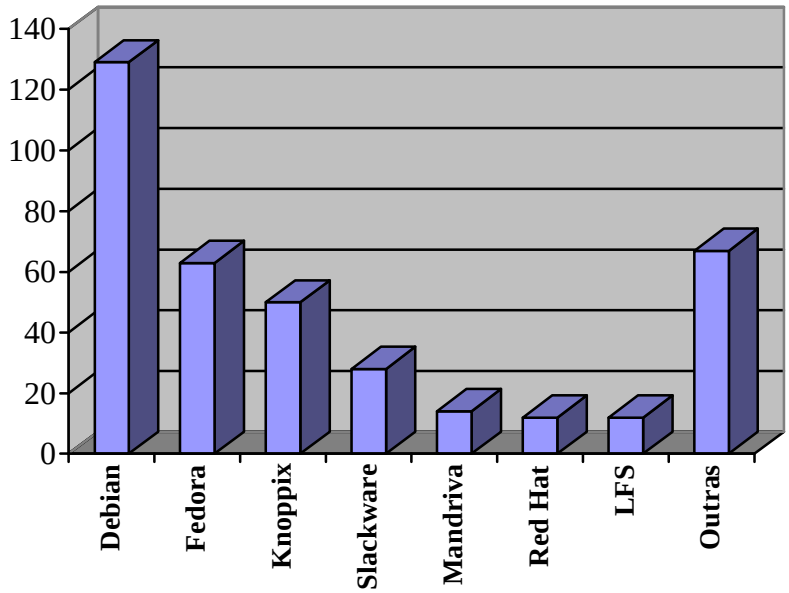

Figura 4. Distribuições Linux baseadas em outras distribuições.
Fonte: DistroWatch (2007)

Atualmente, segundo o site DistroWatch (http://distrowatch.com) existem mais de 350 distribuições GNU/Linux ativas. Essas distribuições variam conforme sua função, podendo ser usadas para usuários comuns ou em servidores. A Tabela 2 apresenta o resultado de uma pesquisa realizada em outubro de 2007 das dez distribuições GNU/Linux mais utilizadas.

Tabela 2 – As 10 distribuições Linux mais utilizadas.

DISTRIBUIÇÃO (Versão Atual)	***SITE***
Ubuntu (versão 7.10)	http://www.ubuntu.com/
OpenSUSE (versão 10.3)	http://www.opensuse.org/
Fedora (versão 8)	http://fedoraproject.org/
Debian (versão 4.0)	http://www.debian.org
Mandriva (versão 2008.0)	http://www.mandrivalinux.com
PCLinuxOS (versão 2007)	http://www.pclinuxos.com/
Mepis (versão 6.5)	http://www.mepis.org/
KNOPPIX (versão 5.1.1)	http://www.knoppix.com/
Slackware (versão 12.0)	http://www.slackware.com/
Gentoo Linux (versão 2007.0)	http://www.gentoo.org/
FreeBSD (versão 6.2)	http://www.freebsd.org/

Fonte: DistroWatch (2007)

Segurança Máxima (2000, p. 23) explica por meio de uma definição bem clara o que é realmente o Linux:

> Linux é um sistema operacional de rede 32-64 bits livre, do tipo UNIX, com o código fonte aberto e otimizado para Internet (freqüentemente utilizado por hackers) que é executado em diferentes hardwares, incluindo processadores Intel (X86) e RISC.

O código fonte aberto permite que qualquer pessoa veja como o sistema funciona (útil para aprendizado), corrija alguma problema ou faça alguma sugestão sobre sua melhoria, oferecendo substanciais benefícios de segurança permitindo examinar o código e ver como o sistema de segurança é implementado (SEGURANÇA MÁXIMA PARA LINUX, 2000).

O Linux trabalha com modularização, carregando somente para a memória o que é usado durante o processamento, liberando totalmente a memória assim que o programa ou dispositivo é finalizado. Devido a isso, os *drivers* dos periféricos e recursos do sistema podem ser carregados e removidos completamente da memória RAM a qualquer momento, o que torna o computador mais rápido (SILVA, 2006).

Embora o Linux seja adequado para uso em *desktops* ele é inerentemente um sistema operacional de rede. A Rede TCP/IP é mais rápida que no Windows e tem sua pilha constantemente melhorada. O GNU/Linux tem suporte nativo a redes TCP/IP e não depende de uma camada intermediária. Em acessos via modem a *Internet*, a velocidade de transmissão é 10% maior (SILVA, 2006).

O Linux é estruturado justamente a partir do sistema de arquivos. De acordo com Ulbricht e Della Valle (2003) qualquer recurso que possa ser manipulada no sistema é tratada com um arquivo. Estão inclusos os dispositivos de *hardware*, como a placa de som e porta impressora, os processos em execução e, obviamente, os arquivos comuns.

Os sistemas de arquivos baseados em disco (*filesystem*) referem-se a forma como os dados são armazenados, organizados e acessados pelo SO. No Linux, os mais conhecidos são: Ext2, Ext3, ReiserFS, XFS, JFS e ISO 9660 (MOTA FILHO, 2006). Um dos recursos que os diferem é a presença de *journaling*[12]. A Tabela 3 apresenta as características mais comuns dos filesystems.

Tabela 3. Filesystems mais comuns para o GNU/ Linux

Filesystems	**Recurso *journaling***	**Descrição**
Ext2	Não	O primeiro filesystem para uso exclusivo no GNU/Linux. Atualmente, não há vantagens no uso do Ext2.
Ext3	Sim	Apresenta melhorias em relação ao Ext2 por possuir o recurso journaling. O Ext 3 só é suportado pelo kernel 2.4 ou superior. Utiliza blocos fixos de 512, 1024, 2048 ou 4096.
ReiserFS	Sim	Escrito por Hans Reiser. Não utiliza blocos com tamanho físico, é mais rápido do que Ext3 na maioria das operações.
XFS e JFS	Sim	São *filesystem* desenvolvidos a partir de produtos proprietários. Não são tão difundidos no mundo GNU/Linunx quanto o Ext2 e o Ext3.

Fonte: MOTA FILHO, J. (2006)

O Linux oferece uma ampla variedade de *softwares*, sendo que a maneira mais segura de adquirir os programas é utilizando os gerenciadores de pacotes, disponíveis em quase todas as distribuições. Os mais conhecidos são o Advanced Package Toll (APT), o Debian PacKaGe (DPKG), o RedHat Package Manager (RPM) e o YellowDog Updater Modified (YUM). Esses gerenciadores permitem realizar as operações básicas na manipulação de pacotes como instalação, remoção, consulta e checagem de arquivos. Os pacotes são consultados na *Internet* para poder baixar o programa no computador e posteriormente ser instalado (TERPSTA, 2005).

Outra característica comum desse SO é a estrutura de diretórios. A perfeita

[12] Técnicas especiais de recuperação de dados em caso de falhas de energia e outros desastres correlatos (MOTA FILHO, 2006).

noção do funcionamento da estrutura ajuda a corrigir problemas ou a implantar novas funcionalidades (MOTA FILHO, 2006). A Tabela 4 apresenta a função dos diretórios.

Tabela 4. Função específica dos diretórios Linux

DIRETÓRIO	DESCRIÇÃO
/bin	Contém arquivo executáveis que podem ser acessados por qualquer usuários.
/boot	Contém o *kernel* e os arquivos que controlam a inicialização do sistema.
/cdrom	É um atalho para o diretório /media/cdrom.
/dev	Contém arquivos que servem de ligação com os dispositivos de hardware.
/etc	É o centro nervoso do SO, contém 95% dos arquivos de configuração.
/home	Contém os arquivos, documentações e configurações dos usuários.
/initrid	Cria um HD virtual na RAM no momento da inicialização, para abrigar o *Kernel*.
/lib	Contém os módulos do *Kernel* e as bibliotecas necessárias ao funcionamento.
/lost + found	Após recuperações de *filesystem*, os arquivos encontrados no disco e que tenham perdido com o seu I-node serão colocados neste diretório.
/media	Trata-se de um ponto de montagem de mídias removíveis (disquetes, *pendrive*,entre outros).
/mnt	Ponto de montagem de *filesystems* localizados em dispositivos de armazenamento não removíveis como o HD.
/opt	Destinado aos programas que não fazem parte da distribuição em questão.
/proc	Diretório virtual. Na verdade, um filesystem virtual, contém referências a informações dinâmicas do sistema.
/root	O usuário *root* é o administrador do sistema. É o único usuário que pode fazer qualquer ação dentro do Linux.
/sbin	Arquinos destinados a administração e manutenção do sistema.
/srv	Abriga as informações que serão servidas pela máquina, como *sites*, arquivos para ftp, entre outros.
/sys	Repositório utilizado pelo *kernel* 2.6 para manter dados atualizados sobre o sistema e os dispositivos de hardware.
/tmp	Diretório utilizado pelo sistema para gravar informações temporárias.
/usr	Contém a maior parte dos dados do sistema, guarda dados compartilhados no modo somente leitura.
/var	Contém dados variáveis, como *logs*, *spool* de impressora, caixas postais em servidores de email, entre outros.

Fonte: MOTA FILHO, J. (2006)

Todo sistema Linux possui váriados controles de segurança. Um dos componentes de arquitetura de segurança existente é que todo o poder administrativo do Linux reside em uma única conta chamada *root.* Com essa conta o administrador controla tudo, incluindo: contas de usuário, arquivos e diretórios e recursos de rede. A

conta permite realizar alterações gerais para todos os recursos ou alterações particulares a somente alguns (SEGURANÇA MÁXIMA PARA LINUX, 2000).

3.3 PRINCIPAIS VULNERABILIDADES DO LINUX

É provável que todo dia uma nova vulnerabilidade seja identificada em algum problema ou sistema operacional. Estas vulnerabilidades permitem que um usuário mal intencionado execute uma operação que não deveria poder executar e geralmente adquire privilégios que não deveria conseguir. Dentre algumas vulnerabilidades conhecidas, pode-se citar: a aquisição de senhas; estouro de *buffers*; utilização de serviços desnecessários, mal configurados e essencialmente inseguros.

3.3.1 Senhas

Kurtz, McClure, Scambray (2000) destacam que senhas fracas, facilmente adivinháveis ou reusadas em nível de estação de trabalho podem levar a comprometimento dos servidores. Uma lista elaborada por atacantes consta, a partir da engenharia social, de elementos como: nome completo do usuário, dos parentes, seus passatempos, modelo de carro ou endereço. Informações como estas são muito úteis e em 50 % dos casos são efetivas (DELLA VALLE; ULBRICHT, 2003).

A senha no Linux identifica o usuário como sendo o verdadeiro dono de uma conta, garantindo, assim, acesso a seus recursos, por isso é tão visada pelos usuários mal intencionados. Dessa forma, existem diversos tipos de ataques para se conseguir uma senha, são eles: dedução, engenharia social, ataques por dicionário, força bruta, monitoramento de toque de teclado e mouse e *login* falso (SILVA, 2006).

Segurança Máxima para Linux (2000) destaca que para se quebrar as senhas, o usuário poderá utilizar ferramentas automatizadas tais como: o Crack, Jhon the Ripper e o Brutus.

3.3.2 *Buffers*

As vulnerabilidades do *buffer* são criadas quando os desenvolvedores usam técnicas impróprias de codificação para executar alguma operação em um programa. Esses procedimentos executados fazem simplesmente o preenchimento de uma variável ou *buffer* do programa com informações excessivas. Dessa forma, pode-se executar um programa de s*hell*, fornecendo, assim, acesso ao sistema ou a um outro aplicativo (HATCH; LEE; KURTZ, 2002). O *buffer overflow* tem sido a forma mais comum de exploração de vulnerabilidade de segurança nos últimos anos (VIANNA, 2004).

3.3.3 Serviços Desnecessários e Portas Abertas

As versões do Linux configuram vários serviços desnecessários quando são instaladas automaticamente ou por padrão, desordenando o sistema e desgastando sua segurança (HATCH; LEE; KURTZ, 2002).

Todo serviço instalado pode introduzir novas, talvez não óbvias ou conhecidas, falhas de segurança em um servidor (PEÑA, 2007). Tais programas podem ser explorados, por exemplo, por um *exploit*, para a execução de um ataque tanto local quanto remoto (CORREIA et al, 2006).

Della Valle e Ulbricht (2003) explicam que um dos primeiros passos de um atacante é analisar os serviços que a máquina dispõem, e acrescentam que a correção é

possível para essas falhas, basta desligar tudo o que não for usar.

Um exemplo de um ataque conhecido que se aproveita de serviços desnecessários, tais como o Echo[13] e Chargen[14], é o Dos. Esse ataque envia um pacote UDP[15] ao serviço Echo com o endereço de origem sendo a porta do Chargen. O serviço Echo e Chargen passam a enviar pacotes entre si causando o esgotamento do sistema e o seu conseqüente travamento. Uma solução óbvia para se evitar tal ameaça é a desabilitação de tais serviços (HATCH; LEE; KURTZ, 2002).

3.3.4 Serviços Desatualizados

Softwares sem aplicação de correções (*patches*), desatualizados, vulneráveis ou deixado com as configurações padrões são um dos principais aspectos que deixam o Linux desprotegido (KURTZ; McCLURE; SCAMBAY, 2000).

Como exemplo de uma vulnerabilidade, publicada no *Site* "*Securityfocus.com*", a Mozilla Foundation lançou treze alertas de segurança especificando vulnerabilidades de segurança no Mozzila Firefox, SeaMonkey, Camino e Thunderbird. Tais vulnerabilidades permitem aos usuários mal intencionados executarem código na máquina por meio da aplicação vulnerável, obter acesso a informações sensíveis e executar código *JavaScript* com privilégios elevados, permitindo o acesso remoto da máquina, sendo afetados sistemas tais como o Ubuntu Linux 6.06 LTS amd64, RedHat Enterprise Linux WS 4, S.U.S.E. Linux Personal 10.1.

Independente da versão que escolher, deve-se certificar de manter os serviços atualizados ou executará versões antigas que possuem erros ou falhas

[13] Proporciona que o servidor envie de volta ao usuário um *reply* de tudo o que é digitado no terminal.
[14] É responsável por gerar um fluxo de dados ACSII.
[15] É um protocolo de transmissão de dados que faz a entrega de mensagens independentes, designadas por datagramas, entre aplicações ou processos, em sistemas *host* (NEMETH, 2002).

conhecidas na segurança viáveis de serem exploradas (HATCH; LEE; KURTZ, 2002).

3.3.5 Serviços Essencialmente Inseguros

Serviços de rede inseguros são aqueles que utilizam nomes de usuário e senha não criptografados para autenticação, pois se um *sniffer* estiver monitorando o tráfego entre o usuário remoto e um servidor desse tipo, os nomes de usuário e senhas podem ser interceptadas facilmente. Telnet[16] e FTP[17] são exemplos de serviços essencialmente inseguros. Outra categoria de serviços inseguros são sistemas de arquivos de rede e serviços de informação, como NFS[18] ou NIS[19], desenvolvidos explicitamente para uso em rede local, mas que, infelizmente, têm seu uso entendido para usuários remotos (RedHat, 2005).

3.4 SERVIDORES LINUX

Um servidor é um sistema de computador que fornece serviços a uma rede de computadores. Hunt (2000) destaca que muitos servidores possuem, geralmente, o propósito de fornecer uma grande quantidade de serviços, entretanto alguns, podem possuir uma única finalidade.

Dallabrida (2004) identifica os diversos tipos de serviços oferecidos pelos servidores: serviço de arquivos; serviço de impressão, serviço *web*, serviço de aplicações, serviço de correio eletrônico, serviço de Fax, serviço de comunicação e

[16] É um protocolo cliente-servidor de comunicações usado para permitir a comunicação entre computadores ligados numa rede (SOUZA, 2001).
[17] É um protocolo de transferência de arquivo (SOUZA, 2001).
[18] É um modelo de sistema de arquivos, que tem como função centralizar arquivos em um servidor, formando assim um diretório virtual (MACHADO, 2002).
[19] É um serviço, desenvolvido pela Sun Microsystens, para distribuição de informações por uma rede (MACHADO, 2002).

serviço de acesso remoto. Outros são encontrados, ainda, dependendo da necessidade.

Dentre esses variados serviços citados, serão descritos abaixo: serviço de páginas *web*, serviço de correio eletrônico, serviço de FTP, serviço de acesso remoto e serviço de resolução de nomes (DNS).

3.3.1 Serviço de Páginas *Web*

Serviço de páginas *web* tornou-se uma parte essencial no dia-a-dia das empresas sendo utilizado para publicação de produtos e oferta de serviços para clientes externos assim como para coordenar e disseminar informações dentro de uma mesma organização (*Intranet*).

Quando um determinado endereço é digitado em um *browser*, ele tenta estabelecer uma conexão TCP/IP com um servidor que "escuta" a porta 80; quando recebem um solicitação de página, esses servidores lêem um arquivo padrão chamado index.html em um diretório pré-estabelecido e devolvem a página ao solicitante (MONTEIRO, 2003).

Dentre os inúmeros *softwares* que provêem serviço w*eb*, o Apache é o servidor WEB mais usado no mundo (algo em torno de 75% das empresas), apresentando como características ser muito rápido e fácil de se configurar (Silva, 2006). Sua popularidade, segundo Hatch, Lee e Kurtz (2002) devem-se ao fato dele ser plenamente configurável, de fonte aberta, gratuito e relativamente seguro, com distribuição quase de imediata das correções, quando uma nova falha é descoberta.

Por conseguinte, os servidores w*ebs* são alvos constantes de ataques. Uma tática, por exemplo, no qual um usuário mal intencionado pode obter informações sobre a máquina é analisar o cabeçalho que o servidor envia, dessa forma poderá empregar o

exploit correto para aquela versão do servidor (HATCH; LEE; KUNT, 2002).

Um outro ponto que deve merecer atenção especial é a capacidade destes servidores serem compatíveis com Common Gateway Interface (CGI) que é uma especificação que permite ao servidor trocar informações com outros programas, portanto os CGI poderão fornecer a qualquer pessoa a capacidade de executar um script de CGI no servidor (NEMETH, 2002).

Uma lista de anúncios que divulgam correções, novas versões e realização de eventos sobre o Apache está disponível em <apache-announce@apache.org> (SILVA, 2006).

3.3.2 Serviço de Correio Eletrônico

O correio eletrônico é a aplicação que fez a *Internet* começar a se popularizar. Monteiro (2003) explica que um servidor de correio eletrônico permite gerenciar os e-*mails* que são enviados e recebidos. Estes servidores podem ser utilizados na *Internet*, onde e-*mails* enviados e recebidos podem ser transitados para qualquer lugar do mundo, ou como correio de *intranet* onde as mensagens trafegam apenas dentro da organização.

O Protocolo de transporte de correio eletrônico mais extensamente utilizado hoje é o *Simple Mail Transfer Protocol* (SMTP). Funciona com um conjunto limitado de regras, tais como: aceita uma mensagem entrante; verifica os endereços da mensagem, se não forem endereços locais, armazena a mensagem para recuperação; e se não forem endereços remotos, encaminha a mensagem (SEGURANÇA MÁXIMA PARA LINUX, 2000).

O serviço de *e-mail* necessita de um *software* de MUA (agente de usuário de correio eletrônico) e um MTA (agentes de transferência de mensagens). O MUA são os responsáveis por permitir que um usuário se conecte e envie uma mensagem, enquanto os MTA são os responsáveis para disponibilizar as mensagens, são esses os papéis desempenhados pelos servidores (SEGURANÇA MÁXIMA, 2000).

O Sendmail é o mais MTA amplamente utilizado, sendo executado por 72% dos servidores de *e-mail* da *internet*, pois é complexo e poderoso. Além desse, também, existe o PostFix e o Qmail (SILVA, 2006).

Hunt (2000) acrescenta que o Sendmail é notoriamente difícil de configurar e os atacantes, portanto, aproveitam dessa dificuldade na expectativa de que o administrador não soube configurar corretamente o servidor.

3.3.3 Serviço de FTP

Segundo Griffith e Norton (2000) o *File Transfer Protocol* (FTP) é um protocolo para *Internet* usado para transferir arquivos de um sistema para outro.

O protocolo FTP utiliza dois canais diferentes para o comando e o fluxo de dados: canal de comando e canal de dados. O canal de comando é o soquete de rede que conecta seu cliente à porta 21 do servidor, enquanto o canal de dados é ativado e interrompido toda vez que o cliente e o servidor possam trocar dados, tais como a transferências de dados com recursos de inserção e captura e listagem de arquivos (HATCH; LEE; KURTZ, 2002).

Tal serviço aborda recursos como controle de autorização de acesso e independência de plataforma, além de possuir acesso interativo, diferentes formatos para os dados e controle de acesso (MARTINS, 2003).

Esse protocolo possui inúmeras vulnerabilidades críticas de segurança, utilizando a autenticação nome de usuário e senha-padrão, isso não permite que o servidor possa determinar se realmente o usuário é quem realmente afirma ser. Por padrão, as senhas são transmitidas em texto claro possibilitando a captura da senha por interceptação; e as sessões de FTP não são criptografadas e portanto não oferecem privacidade (HATCH; LEE; KUNT, 2002).

Um caso comum de erro por parte do administrador de rede, segundo Monteiro (2003) é o esquecimento de desabilitar o compartilhamento de um diretório ou um arquivo após o *download* realizado por parte dos usuários. Esses servidores são alvos constantes de usuários mal intencionados, um servidor com uma pasta pública, aberto a *Internet*, representa um ponto de entrada ou local de armazenamento de dados e *softwares* para esses usarem posteriormente (DELLA VALLE; ULBRICHT, 2003).

3.3.4 Serviço de Acesso Remoto

O Telnet é um protocolo para controlar remotamente outra máquina permitindo a ligação de um computador a um outro (MITNICK, 2005).

Os clientes e servidores trocam dados sem criptografia, criando uma oportunidade para que os dados sejam capturados com o uso de *sniffers*, além disso, também, não emprega autenticação forte e não realiza verificação de integridade de sessão. Devido a tal limitação de não criptografar os dados transmitidos está sendo substituído pelo *secure shell* (ssh) (HATCH; LEE; KUNT, 2002).

O ssh é um programa para efetuar logon em outro programa, executar comandos em uma máquina remota e mover arquivos de uma máquina para outra. Ele fornece autenticação e comunicações seguras sobre rede inseguras (SEGURANÇA

MÁXIMA PARA LINUX, 2000).

3.3.5 Serviço de DNS

O Serviço DNS é um serviço de resolução de nomes para redes TCP/IP. Sem a utilização dos servidores DNS haveria a necessidade de se conhecer a numeração dos *sites*, o que tornaria uma simples navegação tediosa e demorada (MONTEIRO, 2003).

Hunt (2000) explica que para responder uma solicitação por informação DNS, o servidor de nomes local deve ter a resposta à solicitação ou saber qual servidor de nomes a possui, utilizando para isso um sistema hierarquizado de nomes.

O DNS pode ser um banco de dados local que converte automaticamente os nomes em endereços IP ou por meio de servidores DNS desejado. Um servidor DNS mais difundido na *internet* é o Bind (SILVA, 2006).

Segundo Monteiro (2003) um usuário mal intencionado ao identificar que a rede possui um servidor DNS, buscará, basicamente:

a) entrar no servidor LINUX e tomar posse para alterar a base de dados, pois os arquivos DNS são arquivos de textos puros que se encontram no diretório */etc*;

b) tentar derrubar o servidor DNS para desestabilizar a rede.

Um outro exemplo de vulnerabilidade é a possibilidade de um atacante executar “consultar reversas”[20] em todos os endereços de IP para obter seus nomes de *host*, mapeando, dessa forma, a rede sem nem mesmo acessá-la (HATCH; LEE; KUNT, 2002).

[20] É a maneira pela qual um usuário pode obter um nome host a partir de um endereço IP.

4 POLITICA DE SEGURANÇA

Com a grande dependência da Tecnologia da Informação (TI), por parte das organizações, surgiu a necessidade de se considerar como um fator crítico para o sucesso a utilização de soluções viáveis para a proteção da informação. Com esse intuito de exprimir formalmente as regras que devem ser seguidas para se ter acesso aos recursos tecnológicos de uma organização é que se cria uma política de segurança (WANDERLEY, 2005).

Spanceski (2004) acrescenta que uma política de segurança da informação é essencial, pois definem normas, procedimentos, ferramentas e responsabilidades para garantir o controle e a segurança da informação na empresa.

> O propósito principal de uma política de segurança é informar aos usuários, equipes e gerentes, as suas obrigações para a proteção da tecnologia e do acesso à informação. A política deve especificar os mecanismos através dos quais estes requisitos podem ser alcançados (SOUZA, 2001, p. 6).

O presente capítulo apresenta os principais aspectos que devem constar na definição de uma política de segurança, citando as normas e padrões de segurança utilizados como referência na segurança da informação.

4.1 DEFINIÇÃO DE UMA POLÍTICA DE SEGURANÇA

A definição de uma política de segurança, segundo Dumont (2006), é o elemento mais importante da segurança da informação e deve envolver a segurança física, lógica e outros componentes relacionados à TI. Sem uma política de segurança bem elaborada não se sabe o que se vai proteger, nem porque e qual a melhor forma.

Cabe destacar que a política de segurança de uma organização não é uma política isolada, devendo integrar-se às políticas institucionais, às metas de negócios da

organização e ao plano estratégico de informática (DIAS, 2000).

A Figura 5 demonstra o relacionamento da política de segurança de informações com a estratégia da organização, o plano estratégico de informática e os diversos projetos relacionados.

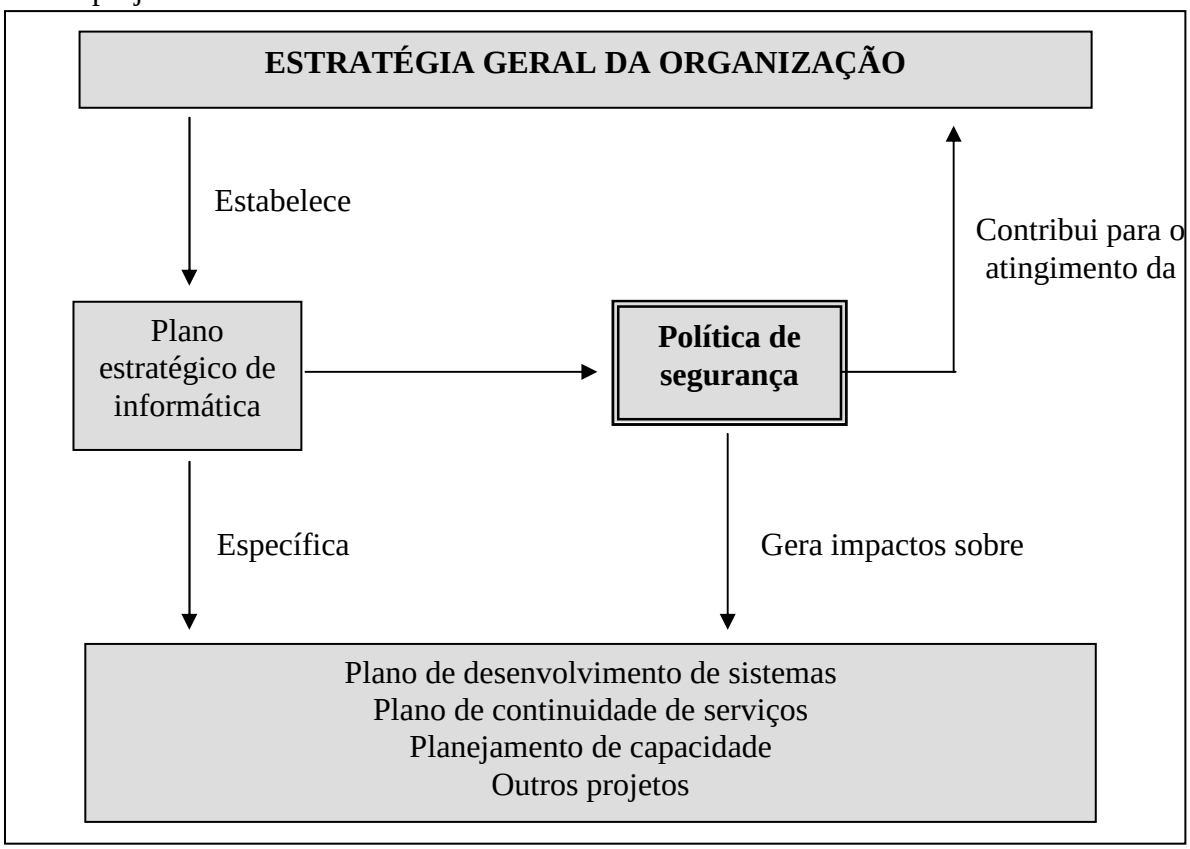

Figura 5. Política de Segurança de informações e seus relacionamentos
Fonte: DIAS, C. (2000)

A avaliação dos possíveis riscos envolvidos no processo leva ao avaliador, por meio da política de segurança, a necessidade de definir inicialmente quais são as metas de segurança. Wanderley (2005) destaca as seguintes determinantes:

a) serviços oferecidos aos usuários: cada serviço apresenta seu próprio risco, devendo o administrador avaliar sua real necessidade de utilização;

b) facilidade de uso: sistemas mais fáceis de usar normalmente dão ao usuário acesso total e irrestrito às informações. No mínimo, seria necessário exigir de cada usuário uma senha. De uma maneira geral, comodidade costuma ser o inversamente proporcional à segurança;

c) perda de informações: perda de dados (os dados podem ser corrompidos ou excluídos), violação de privacidade (leitura de informações por pessoas não autorizadas) e perda de serviços (impossibilidade de acesso à rede).

A política de segurança deve ser estruturada da seguinte forma: deverá conter um objetivo, a área de atuação, os usos corretos dos sistemas, tais como *Internet*, *hardware*, *e-mail*, entre outros, classificação do uso da informação, controle de acesso, mecanismo de proteção, treinamento de usuários, planejamentos de auditorias. A documentação final deverá reunir dentre outros documentos: resultados das auditorias, plano de contingências da organização para área de TI, políticas de acesso, senhas e *backups*, procedimentos para administração de *logs* e outros documentos conforme a finalidade e a estrutura de TI da organização (MONTEIRO, 2003).

Os elementos de uma política de segurança devem manter a disponibilidade à infra-estrutura da organização. Segundo Spanceski (2004) são essenciais para a definição e implantação da política de segurança:

a) vigilância: todos os membros da organização devem entender a importância da segurança para o sucesso da mesma;

b) atitude: correta postura e conduta em relação à segurança de conhecimento de todos;

c) estratégia: deve ser criativo quanto às definições da política e do plano de defesa contra intrusões, possuir a habilidade de se adaptar as mudanças.

d) tecnologia: o ideal é a implantação de uma política de segurança dinâmica em que múltiplas tecnologias e práticas de segurança são adotadas.

Segundo Uchoa (2003 apud NASCIMENTO, 2004), as políticas de segurança diferem em dois ramos: segurança física e lógica.

4.1.1 Segurança física

Segundo Wanderley (2005), dentre os diversos componentes de uma política de segurança pode-se dizer que a segurança física é um dos mais importantes. É ela que vai ditar as normas de acesso físico, normas para garantir a integridade física dos dados e dos equipamentos e infra-estrutura das instalações. Dentre os inúmeros aspectos analisados na segurança física destaca-se: a segurança contra fogo; a climatização; às instalações elétricas; e o controle de acesso físico.

Um bom modelo concebido para a segurança física é conhecido como o modelo da cebola ou a solução em níveis. O gradiente de segurança está definido, cobrindo tudo que vem do exterior para a parte mais interna da segurança (WADLOW, 2000). A idéia é descrever os vários lugares aos quais a segurança física será aplicada e então descrever as transições entre eles e as condições que permitirão alguém realizar as transições. A Figura 6 mostra um exemplo simplificado dessa hierarquia de acesso.

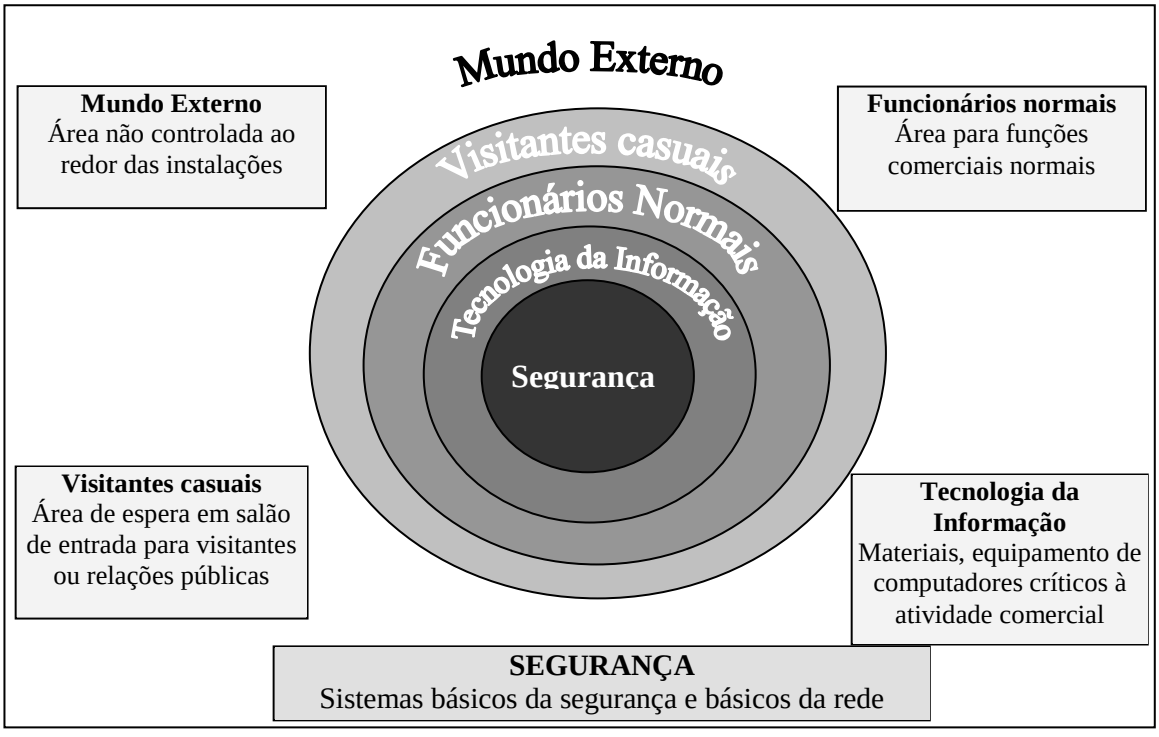

Figura 6. Modelo da cebola para a segurança física
Fonte: Adaptado de WADLOW, T. (2000)

A segurança de acesso físico pode ser implantada a partir de diversas formas, tais como: grades e muros, guardas, crachás, sistema com portas duplas, travas e chaves e controle de acesso biométrico (LEMOS, 2001).

4.1.2 Segurança lógica

Os controles que permitem o acesso lógico, podem ser um conjunto de medidas e procedimentos, adotados pela instituição ou intrínsecos aos *softwares* utilizados, cuja finalidade é proteger dados, programas e sistemas contra tentativas de acessos não autorizados (DIAS, 2000).

A implementação de dispositivos de proteção provendo a segurança lógica deve ser plenamente utilizada buscando minimizar o risco de alteração, destruição ou roubo dos dados. Dentre inúmeras regras, destacam-se: desabilitar *logins* remotos quando não for utilizar em um determinado período, ter uma política de *backup* eficiente, criar regras de segurança para uso de máquinas, manter um rotina de atualização de *software*, utilizar ferramentas de criptografia de discos e pastas, ter uma política de treinamento e uso de senhas nos diversos sistemas da organização (MONTEIRO, 2003).

4.2 IMPLANTAÇÃO DA POLÍTICA DE SEGURANÇA

É possível que a implantação de uma política de segurança venha trazer alguns transtornos iniciais, pois toda mudança é passível de um atraso nas atividades que eram rotineiramente executadas sem controle. Porém, o importante é que ela seja

verdadeira e que atenda a real necessidade da organização e de seus funcionários (WANDERLEY, 2005).

> Para que seja possível a implantação da política é necessário que a alta direção tenha aprovado, comunicado e publicado, de maneira adequada para os funcionários. É necessário que a alta direção esteja sempre preocupada com o processo e estabeleça as linhas mestras para a gestão da segurança da informação. (CACIATO, 2004, p. 18).

Devem ser feitos programas de conscientização e de divulgação da política, de modo que com a divulgação efetiva a preocupação com a segurança deverá tornar-se parte da cultura da organização (SPANCESKI, 2004).

4.3 NORMAS E PADRÕES DE SEGURANÇA DA INFORMAÇÃO

Para implantar uma eficiente política de segurança da informação é recomendável, portanto que a instituição se mantenha atualizada quanto às normas e aos padrões de segurança estabelecidos por organismos nacionais e internacionais (DIAS, 2000).

O *Orange Book* foi desenvolvido pelo Centro de Avaliação do Departamento de Defesa dos EUA e publicado em 1985, é considerado como o marco inicial de um processo mundial e contínuo de busca de um conjunto de medidas que permitam a um ambiente computacional ser qualificado como seguro. Tal norma permitiu e continua permitindo a classificação, por exemplo, do nível de segurança fornecido pelos sistemas operacionais, como são os casos do OpenBSD, do FreeBSD, do NetBSD, do Solaris, do AIX, do QNX, das várias distribuições do Linux e até mesmo das várias versões do Windows (MARTINS, 2002).

Outras normas também tratam diretamente ou indiretamente da segurança

da informação, pode- se citar a **Common Criteria** (ISO/IEC 15408)[21]; **CobiT**[22]; **Itil**[23]; Norma **ISO/IEC TR 13335ª** a qual apresenta uma visão geral dos conceitos e modelos fundamentais usados na gestão de segurança de TI; e Norma **IEC 61508** que enfoca as atividades do Ciclo de Vida de Segurança para os Sistemas Elétricos, Eletrônicos e Eletrônicos Programáveis (LIMA; SILVA; SOUTO, 2006). Vigliazzi (2002) acrescenta que dentre o cabedal de normas, o **RFC 2196 – Security HandBook,** que é guia técnico para implantação de políticas de segurança que possui computadores ou recursos de rede relacionados e que preferencialmente possuem seus sistemas conectados à *Internet*, este trabalho foi desenvolvido pelo trabalho cooperativo de vários autores.

Além dessas, as principais normas utilizadas como referência de Segurança da Informação está a BS 7799, NBR ISO/IEC 17799, NBR ISO/IEC 27001.

4.3.1 BS7799

A Norma BS7799 foi um documento lançado pelo Governo Britânico, por intermédio da *British Standards Institute* (BSI), com abordagens em: políticas de segurança, organização da segurança, controle e classificação de ativos, segurança de pessoal, segurança física e do ambiente, operação e comunicação, controle de acesso, desenvolvimento de sistemas e manutenção, continuidade de negócios e conformidades legais (MONTEIRO, 2003).

Em maio de 2000, o BSI homologou a BS7799-1 que é denominada de Código de Prática para Gestão da Segurança da Informação, e não é passível de certificação. Ela apresenta dez cláusulas que agrupam controles e objetivos de controles

[21] É uma norma internacional que é voltada para a segurança lógica das aplicações e para o desenvolvimento de aplicações seguras.
[22] É um guia que possui recursos que podem servir como um modelo de referência para gestão da TI.
[23] É uma biblioteca de boas práticas de segurança que busca promover a gestão com foco no cliente e na qualidade dos serviços de TI.

com o intuito de direcionar a gestão e o suporte para segurança da informação (MARTINS, 2002).

Já a Norma BS7799-2, que foi publicada em 1998, e revisada em 2002, é um padrão que aponta as especificações necessárias para um Sistema de Gestão da Segurança da Informação (SGSI), que é um sistema de gestão análogo ao sistema da qualidade e como tal é passível de certificação (MARTINS, 2002). Essa certificação se dá a partir das evidências (documentos e práticas) do conjunto de controles implantados e que devem ser continuamente executados e devidamente registrados.

Em março de 2006, foi publicado a Norma BS7799-3 que é um guia de gestão de riscos para SGSI. O guia aborda, dentre vários assuntos, as decisões gerenciais para o tratamento dos riscos, monitoramento e revisão do perfil dos riscos e reavaliação dos riscos.

4.3.2 NBR ISO/IEC 17799

Com a publicação da Norma BS7799-2, a BSI realizou uma campanha para adoção mundial de seu padrão. Em outubro de 2000, na reunião do comitê da *Internacional Organization for Standardization* (ISO) em Tóquio, a norma ISO 17799 foi votada e aprovada pela maioria dos representantes (MONTEIRO, 2003). A Associação Brasileira de Normas Técnicas (ABNT) editou, em 2000, uma norma equivalente, a NBR ISO/IEC 17799 - Tecnologia da Informação – Código de Prática para a Gestão da Segurança da Informação.

Essa Norma, reeditada em 2005 e com uma errata lançada em agosto de 2006, serve como um guia prático para desenvolver eficientes práticas de gestão da segurança, procedimentos de segurança da informação da organização e para ajudar a criar confiança nas atividades interorganizacionais.

A NBR ISO/IEC 17799 tem por objetivo:

> Esta Norma estabelece diretrizes e princípios gerais para iniciar, implementar, manter e melhorar a gestão de segurança da informação em uma organização. Os objetivos definidos nesta Norma provêem diretrizes gerais sobre as metas geralmente aceitas para a gestão da segurança da informação (NBR ISO/IEC 17799, 2005, p. 1).

A norma contém onze seções de controles de segurança da informação, que juntas totalizam trinta e nove categorias principais de segurança e uma seção introdutória que aborda a análise, avaliação e o tratamento de riscos, sendo suas seções:

a) política de segurança da informação;

b) organizando a segurança da informação;

c) gestão de ativos;

d) segurança em recursos humanos;

e) segurança física e do ambiente;

f) gestão das operações e comunicações;

g) controle de acesso;

h) aquisição**,** desenvolvimento e manutenção de sistemas de informação;

i) gestão de incidentes de segurança da informação;

j) gestão da continuidade do negócio;

k) conformidade.

Segundo essa normativa a política de segurança da informação tem por objetivo prover uma orientação e apoio da direção para a segurança da informação de acordo com os requisitos dos negócios e com as leis e regulamentações relevantes.

Convém que a direção estabeleça uma política clara, alinhada com os objetivos do negócio e demonstre apoio e comprometimento com a segurança da informação por meio da publicação e manutenção de uma política de segurança da informação para toda a organização (NBR ISO/ IEC 17799, 2005).

Cabe destacar, segundo essa norma que o documento da política contenha informações relativas a:

a) definição de segurança da informação, suas metas globais, escopo e importância da segurança da informação como um mecanismo que habilita o compartilhamento da informação;

b) declaração do comprometimento da direção, apoiando as metas e princípios da segurança da informação, alinhada com os objetivos e estratégias do negócio;

c) estrutura para estabelecer os objetivos de controle e os controles, incluindo a estrutura de análise, avaliação e gerenciamento de risco;

d) breve explanação das políticas, princípios, normas e requisitos de conformidade de segurança da informação específicos para a organização, incluindo: conformidade com a legislação e com requisitos regulamentares e contratuais, requisitos de treinamento e educação em segurança da informação, gestão da continuidade do negócio, e conseqüências das violações na política de segurança da informação;

e) definição das responsabilidades gerais e específicas na gestão da segurança da informação, incluindo o registro dos incidentes de segurança da informação; e

f) referências à documentação que possam apoiar a política, por exemplo, políticas e procedimentos de segurança mais detalhados de sistemas de informação ou regras de segurança que os usuários devem seguir.

Essa norma chegou em um momento em que as organizações de todo o mundo passaram a investir muito mais em segurança da informação. Pela sua ampla notoriedade, a norma ISO 17799 passou a ser referenciada como sinônimo de segurança

da informação por praticamente toda a mídia especializada em segurança da informação (MARTINS, 2002).

4.3.3 NBR ISO/IEC 27000

Esse conjunto de normas ISO/IEC é o mais importante referencial de Segurança da Informação. Essas normas estão substituindo as normas **BS 7799-2** (referente à Gestão de Segurança da informação) e **ISO 17799** (Código de Boas Práticas da Gestão de Segurança da Informação). Na família 27000, novos segmentos serão abordados sob normas que variam de 27000 à 27009, com previsão de serem publicadas até 2009 (LIMA; SILVA; SOUTO, 2006)

No Brasil, a norma NBR ISO/IEC 27001 foi publicada pela ABNT em março de 2006, sendo preparada para prover um modelo para estabelecer, implementar, operar, monitorar, analisar criticamente, manter e melhorar um Sistema de Gestão de Segurança da Informação (SGSI).

O SGSI “é a parte do sistema de gestão global, baseado na abordagem de riscos do negócio, para estabelecer, implementar, operar, monitorar, analisar criticamente, manter e melhorar a segurança da informação” (ABNT, 2006, p. 3).

Essa norma cobre todos os tipos de organizações, por exemplo, empreendimentos comerciais, agências governamentais, organizações sem fins lucrativos, especificando os requisitos para a implementação de controles de segurança para as necessidades individuais de organizações ou suas partes (ABNT, 2006).

Em conjunto com a ISO/IEC 17799 (Código de Boas Práticas da Gestão de Segurança da Informação) são as principais referências, atualmente, para quem procura tratar a questão da segurança da informação de maneira eficiente.

5 MODELO DE POLITICA DE SEGURANÇA

O modelo de política de segurança proposto neste capítulo define procedimentos específicos de manipulação e proteção da informação para ambientes que utilizam servidores Linux, podendo ser aplicado, também, em servidores que usam outros Sistemas Operacionais fazendo-se necessário às devidas adaptações.

A presente pesquisa buscou estabelecer controles de segurança sob o alicerce de normas e padrões estabelecidos, sendo principalmente norteada pela norma NBR ISO/IEC 17799. Os controles sugeridos não são específicos para um determinado servidor, mas sim, procedimentos comuns a serem realizados em diferentes tipos de servidores Linux, tais como servidor Debian, Fedora, Slackware, RedHat, dentre outros. Utilizou-se pra fins de estudo a distribuição Ubuntu Server 7.04, por ser baseada no Debian, distribuição, segundo o *site* DistroWatch, com maior número de adeptos no mundo.

Para facilitar a compreensão do trabalho para cada controle estabelecido foi referenciada a norma, justificando-se o porquê de se executar tal medida, e sugerido uma solução a ser implementada nos Servidores Linux. Para se chegar a esses resultados foi realizada pesquisa bibliográfica e documental na área da segurança da informação e do sistema operacional Linux. No apêndice A contém exemplos dos comandos do LINUX e das modificações necessárias em alguns dos arquivos de configuração utilizados na presente pesquisa.

A divisão dos controles foi baseada em normas e padrões publicados, técnicas e tecnologias hoje empregadas para prover segurança, e nas ações mais comuns que o usuário mal intencionado executa ao tentar invadir um sistema. Os controles são:

a) segurança em recursos humanos;

b) controle do ambiente físico e do meio ambiente;

c) controles na instalação e implementação do servidor;

d) controles de acesso ao servidor;

e) controles de preservação da integridade e disponibilidade da informação;

f) controles de mecanismos de proteção ao servidor;

g) controles de monitoramento, auditoria e teste.

Por fim, complementando o trabalho foram realizados testes, em uma instituição, sendo eticamente preservada seu nome, verificando a aplicação do presente modelo de segurança versus alguns tipos de ataques aos quais os servidores estão sujeitos.

5.1 SEGURANÇA EM RECURSOS HUMANOS

Com relação à “segurança em recursos humanos”, a norma NBR ISO/IEC 17799 indica os seguintes cuidados, no item 6.1, com o objetivo de assegurar que os funcionários, fornecedores e terceiros entendam suas responsabilidades e estejam de acordo com os seus papéis, e reduzir o risco de roubo, fraude ou mau uso de recursos, convém que:

1) a responsabilização pela segurança da informação deve ser atribuída antes da contratação, de forma adequada, nas descrições de cargos e nos termos e condições de contratação;

2) a análise de todos os candidatos ao emprego, fornecedores e terceiros;

3) a assinatura de acordos sobre os papéis e responsabilidades pela segurança pelos usuários dos recursos de processamento da informação.

Segurança de pessoal corresponde a tudo que envolve os usuários: a forma de contratação, treinamento, a monitoração de suas atividades e até mesmo demissão. Estatísticas mostram que a forma mais comum de ocorrência de crimes de invasão de

computadores envolve as pessoas que possuem acesso legítimo ou que já tiveram acesso recentemente. Alguns estudos mostram que cerca de 80% dos incidentes são causados por esses indivíduos (CORDEIRO; MOREIRA, 2002).

Por ocasião da contratação ou promoção é necessário verificar a idoneidade dos funcionários efetivos, temporários ou terceirizados que de alguma maneira terão acesso às informações consideradas sensíveis ou sigilosas existentes nos servidores.

A causa fundamental da vulnerabilidade na segurança de computadores é delegar pessoas não treinadas para manter a segurança e não prover treinamento para que o trabalho seja executado. Isso se aplica tanto para administradores inexperientes quanto para administradores superconfiantes ou desmotivados (REDHAT, 2005).

Wanderlei (2005) complementa, ainda, que todo investimento em segurança pode se perder se o usuário não estiver preparado para seguir as normas da política da instituição. Seria importante que todo usuário fosse conscientizado da importância da segurança por meio de um treinamento, o que minimizaria os erros provenientes de falha humana.

A Norma NBR ISO/IEC 27001 dispõe no item 5.2.2 que a organização deve assegurar a que todo o pessoal responsável na gestão da segurança da informação tenha competência para desempenhar as tarefas requeridas. Portanto, o treinamento tem por objetivo garantir aos usuários a conscientização das ameaças e das preocupações de segurança da informação a fim de estarem aptos a apoiar a política de segurança da organização durante a execução de suas tarefas (BANDEIRA; SALGADO; SILVA, 2004).

Os termos e condições de trabalho devem determinar as responsabilidades de cada um estabelecendo a fiel execução das normas de segurança da informação e devem incluir as ações a serem tomadas em caso de desrespeito ao acordo. Ao detectar a violação da política de segurança, a primeira tarefa a fazer é realizar um processo de investigação, determinando a razão pelo qual ocorreu, quer seja por negligência, por

acidente ou erro, por desconhecimento da política ou por ação previamente determinada, ignorando a política estabelecida (SPANCESKI, 2004).

O item 8.2.3 da norma NBR ISO/IEC 17799 estabelece que é conveniente a existência de processo disciplinar formal para os funcionários que tenham cometido a violação da segurança da informação. Spanceski (2004) sugere que o não cumprimento das normas estabelecidas na política de segurança poderá causar, de acordo com a infração cometida, as seguintes punições: comunicação de descumprimento, advertência, suspensão ou até demissão por justa causa, quando houver um ato de natureza gravíssima e previsto no art 482 da Consolidação das Leis de Trabalho - CLT.

Cordeiro e Moreira (2002) acrescentam que caso ocorra encerramento da atividade, por demissão ou por qualquer motivo, devem ser tomadas medidas cabíveis quanto à retirada do direito de acesso e da devolução de dispositivos que pertençam à organização (exemplo: crachás, chaves, entre outros). Recomenda-se que se façam ajustes, se for o caso, na segurança física, como exemplo, a substituição de chave.

Dessa forma, o gerenciamento do pessoal é fator chave de sucesso de qualquer organização. Possuir quadros bem selecionados, cientes de suas responsabilidades, com elevado conhecimento técnico profissional e motivados com a perene preocupação em aplicar medidas pró-ativas da segurança da informação trará inúmeros benefícios em prol da organização. Em contrapartida funcionários que, por ventura, não respeitarem a política de segurança prevista devem receber um tratamento justo e compatível com a violação ocorrida.

5.2 CONTROLE DE AMBIENTE FÍSICO E DO MEIO AMBIENTE

A infra-estrutura de informação deve estar bem protegida de ameaças físicas

potenciais, a fim de assegurar o desempenho de todos os componentes do sistema (CORDEIRO; MOREIRA, 2002). Faz-se, então, a necessária adoção de medidas de prevenção à segurança de equipamentos, ao controle de entrada física e à proteção contra ameaça externas e do meio ambiente.

5.2.1 Segurança de equipamentos

No quesito “segurança de equipamento”, a norma NBR ISO/IEC 17799 no item 9.2 estabelece a conveniência dos equipamentos serem protegidos contra possíveis ameaças. Destaca, ainda, a necessidade de proteção dos equipamentos para reduzir o risco de acesso não autorizado às informações e para proteger contra perdas ou danos.

Na instalação e proteção do equipamento deve-se observar:

a) o local de instalação do servidor deve, se possível, utilizar materiais resistentes ao fogo, tubulações de água e esgoto não devem passar pela sala e as condições do ambiente devem ser adequadas quanto à climatização, condicionamento de ar, proteção dos dutos de ventilação, umidificação, aterramento adequado, piso falso para os cabos de força e lógica, para evitar, em casos de incêndio, a propagação do fogo e de gases (WANDERLEY, 2005);

b) os monitores que apresentam informações sensíveis devem posicionar-se de forma que o ângulo de visão seja restrito, de modo a reduzir o risco de que as informações sejam vistas por pessoal não autorizado durante a sua utilização (ABNT, 2005);

c) a localização do equipamento deve ser em local separado e longe do acesso do público (SEGURANÇA MÁXIMA PARA LINUX, 2000);

d) a conexão de qualquer equipamento aos sistemas ou rede deve ser precedida de aprovação prévia e, se necessário, sob supervisão do administrador (FRANCINI, 2004);

e) a utilização, se possível, de sistema de vigilância eletrônica, estendendo o sinal da câmera para um VCR remoto (MONTEIRO**,** 2003);

f) a proibição de consumo de alimentos e bebidas e de fumo nas proximidades dos servidores (FRANCINI, 2004);

g) a instalação de grades de segurança, se for o caso, da parte interna dos aparelhos de ar condicionado, impossibilitando o acesso direto à parte interior da sala quando da remoção destes (CORDEIRO; MOREIRA, 2002);

h) o registro de dados dos componentes do hardware, incluindo modelos e números seriais, com o objetivo de identificar, posteriormente, o servidor caso seja necessário (SEGURANÇA MÁXIMA PARA LINUX, 2000).

i) a autorização prévia para retirada de quaisquer equipamentos, informações ou *software* do local com o registro da saída e da devolução dos equipamentos (ABNT, 2005);

j) a proibição de uso de máquinas fotográficas, gravadores de vídeo ou áudio ou de outros equipamentos de gravação, salvo se for devidamente autorizado (WANDERLEY, 2005).

Quanto à segurança do cabeamento, a norma NBR ISO/IEC 17799 estabelece as seguintes diretrizes:

a) as linhas de energia e de telecomunicações que entram nas instalações de processamento da informação devem ficar abaixo do piso, sempre que possível, ou devem receber;

b) o cabeamento de redes deve ser protegido contra interceptação não autorizada ou danos, por exemplo, pelo uso de conduítes ou evitando trajetos da passagem dos cabos em áreas públicas;

c) os cabos de energia devem ser separados dos cabos de comunicações, para evitar interferências;

d) os cabos e nos equipamentos, devem utilizar marcações claramente identificáveis, a fim de minimizar erros de manuseio e que sejam documentados as conexões;

e) os seguintes controles adicionais para sistemas sensíveis ou críticos devem ser considerados:

- instalação de conduítes blindados e salas ou caixas trancadas em pontos de inspeção e pontos terminais,
- uso de rotas alternativas e/ou meios de transmissão alternativos que proporcionem segurança adequada,
- utilização de cabeamento de fibras ópticas,
- utilização de blindagem eletromagnética para a proteção dos cabos,
- realização de varreduras técnicas e inspeções físicas para detectar a presença de dispositivos não autorizados conectados aos cabos,
- acesso controlado aos painéis de conexões e às salas de cabos.

5.2.2 Controle de Acesso Físico

No quesito “controle de acesso físico”, a norma NBR ISO/IEC 17799 estabelece, no item 9.1.2, que convém que as áreas seguras sejam protegidas por

controles apropriados de entrada para assegurar que somente pessoas autorizadas tenham acesso.

> Convém que sejam levadas em consideração as seguintes diretrizes:
>
> a) restrição de acesso físico aos servidores, permitindo-o acesso somente a pessoas autorizadas, necessitando como requisito básico a identificação de cada indivíduo. Esses acessos devem ser constantemente revistos e atualizados em intervalos regulares (ABNT, 2005);
>
> b) proteção para as salas que deverão possuir portas seguras, podendo utilizar-se de fechaduras eletrônicas com dispositivos de acionamento biométricos (impressões digitais, voz, retina, entre outros) ou cartões magnéticos (MONTEIRO, 2003). Se não for possível o uso de tais tecnologias, uma simples roleta possibilita um meio de controle de acesso;
>
> c) restrição, segundo a RFC 2196, ao acesso físico aos gabinetes de fiação e roteadores;
>
> d) registro da data e hora da entrada e saída de visitantes. Todos os visitantes devem ser supervisionados, a não ser que o acesso tenha sido previamente aprovado para finalidades específicas (ABNT, 2005);
>
> e) concessão restrita às áreas seguras ou às instalações de processamento da informação sensível somente quando necessário para terceiros. O acesso deve ser autorizado e monitorado; (ABNT, 2005).

5.2.3 Proteção contra ameaças externas e meio ambiente

Com relação à “proteção contra ameaças externas e meio ambiente”, a norma NBR ISO/IEC 17799 dispõe, no item 9.1.4, sobre a projeção e aplicação de

proteção física contra incêndios, enchentes, explosões, perturbações da ordem pública e outras formas de desastres naturais ou causados pelos homens. Os seguintes itens são fundamentais para a execução do referido item:

a) os extintores devem ser os equipamentos mais utilizados no combate a incêndios, observando-se suas diversas classes, hidrantes e *sprinklers*[24] (WANDERLEY, 2005);

b) os detectores de fumaça do tipo iônico são recomendáveis, pois possuem tempo de resposta menor, em conjunto com detectores inteligentes ópticos de fumaça, permitindo funcionar por mais tempo em condições de sujeira (SOUZA, 2004);

c) o armazenamento de qualquer material combustível utilizado, principalmente, na limpeza de equipamentos ou para qualquer outra finalidade não pode em hipótese alguma, ser feito na sala (WANDERLEY, 2005);

d) a área utilizada para instalações dos servidores deve ter controles ambientais, que incluem controle de temperatura, umidade e proteção contra eletricidade estática (MARTINS, 2003);

e) a área também deve ser dotada de proteção contra raios e todas as linhas de entrada de força e de comunicações tenham filtros de proteção contra descargas elétricas (ABNT, 2005). Outro fator não menos importante e que deve ser levado em consideração é o aterramento elétrico nos servidores (WANDERLEY, 2005);

f) os equipamentos devem ser protegidos contra falta de energia elétrica utilizando, para tal, *no-breaks*, que permitem o fornecimento de energia por

[24] Sistema automático de combate ao fogo por meio de água. São instalados em vários pontos do forro e assim que a temperatura aumenta uma espécie de gatilho libera água.

um tempo determinado, ou por meio da instalação de um grupo gerador diesel, que deverá entrar em operação toda vez que houver falta da alimentação externa (SOUZA, 2004);

g) a utilização de dispositivos de proteção física antifurtos, quer do sistema do sistema inteiro ou quer de componentes individuais (CACIATO, 2004).

Portanto, o estabelecimento de medidas que visem a segurança física em uma política de segurança para servidores Linux, contra ameaças externas ou do meio ambiente, contribuirão para a prevenção do acesso físico não autorizado e para a minimização ou até o impedimento de possíveis danos e interferências com as instalações e as informações da organização.

5.3 CONTROLES NA INSTALAÇÃO E IMPLANTAÇÃO DO SERVIDOR

Existem inúmeros documentos disponíveis na *Internet*, sob a forma de tutoriais e guias, que tratam da instalação do Linux, nas diferentes distribuições. Contudo, muitos deles não focalizam a segurança durante este processo (SEGURANÇA MÁXIMA PARA LINUX, 2000).

Observa-se que muitos administradores preparam seus servidores com a instalação básica e a deixam da maneira como o fizeram pelo motivo de estarem funcionando, entretanto uma vez devidamente trabalhado, aperfeiçoando-se as suas configurações padrões, o Linux tornar-se um sistema mais seguro (CORREIA et al, 2006).

Outro ponto que merece destaque, segundo a NBSO (2003) é a necessidade de se elaborar um documento (*logbook*) que detalhe os componentes instalados no

sistema e todas as modificações na sua configuração global. A Figura 7. apresenta um exemplo de arquivo de texto elaborado com essa finalidade.

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 *.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.

Figura 7. Entradas no logbook
Fonte: NBSO, 2003

Para implementar a segurança de maneira adequada, deve-se primeiro decidir o que se deseja com o sistema, sendo necessário planejar a sua instalação, definindo itens como: o propósito do sistema a ser instalado, os serviços que esse sistema disponibilizará, a configuração de hardware da máquina, a forma de particionamento do disco, entre outros (NBSO, 2003). Portanto, o propósito do presente item é estabelecer alguns procedimentos comuns de instalação e configuração que fortalecerá frente as possíveis ameaças.

Serão estabelecidas as seguintes etapas no controle da instalação e da implantação do servidor: segurança no sistema de arquivos (particionamento),

instalação mínima, desabilitação de serviços desnecessários, instalação de correções de segurança e configuração dos serviços utilizados.

5.3.1 Segurança no Sistema de arquivos (Particionamento)

No que diz respeito à "segurança em sistemas de arquivos", a norma NBR ISO/IEC 17799 diz, no item 10.4, que deve-se proteger a integridade do *software* e da informação.

Quando se utiliza somente uma única partição[25], o sistema operacional pode gravar os dados arbitrariamente, onde conseguir localizar um espaço adequado, dessa mesma forma usuários ao gravarem um arquivo poderão, também, gravá-lo de modo desorganizado (SEGURANÇA MÁXIMA PARA LINUX, 2000). Para se evitar uma única partição ocupando o disco inteiro, a NBSO recomenda dividir o disco em várias partições por diversos motivos relatados abaixo:

a) evitar o loteamento de uma partição pelo mau uso de um programa por parte de um usuário no qual tenha permissão de escrita, o que poderá acarretar o travamento do sistema. Caso os programas do sistema estiverem em outra partição eles provavelmente não serão afetados;

b) diminuir a possibilidade de comprometimento de outras partições caso, por alguma razão uma outra partição seja corrompida;

c) permitir o estabelecimento de configurações individuais no sistema Linux, por meio do comando **mount**, para cada partição;

d) facilitar o procedimento de *backup* do sistema para simplificar funções como: cópias de partições inteiras de uma só vez, exclusão de partições

[25] São áreas em uma unidade de disco que são reservadas para os sistemas de arquivos.

individuais do procedimento e a confecção de *backup* em intervalos diferentes para cada partição.

As partições devem ser estruturadas e dimensionadas de acordo com os requisitos de cada sistema. Em muitos casos, o tamanho ocupado pelo sistema operacional é fornecido na documentação do sistema. Quanto ao tipo de sistema, normalmente é melhor usar o ext3, pois é compatível com o antigo ext2 e tem suporte a *journaling* (MOTAFILHO, 2006).

Deve-se providenciar a partição dos seguintes diretórios: qualquer diretório que um usuário tenha permissão de escrita (*/home, /tmp, /var/temp*), qualquer partição com dados variáveis (*/var*) e qualquer partição onde se pode instalar *software* que não é padrão da distribuição (*/opt* ou */usr*) (PEÑA, 2007).

Recomenda-se, ainda, a conveniência de criação de partições separadas para as seguintes áreas: filas de envio e recepção de *e-mail* (serviço SMTP), filas de impressão (serviço de impressão), repositórios de arquivos (serviço de FTP) e páginas *Web* (serviço HTTP) (NBSO, 2003).

Algumas distribuições oferecem ao usuário rotinas de instalação amigáveis que permitem estabelecer as partições nesse processo, entretanto, pode-se redimensioná-las posteriormente, utilizando-se aplicativos como o Fdisk, Cfdisk, Disk Druid.

Após criada e formatada, a partição será identificada como um dispositivo no diretório */dev* e deverá ser montada, por meio do comando **mount,** descrito no Apêndice “A”- Comandos do Linux que permite seu uso no sistema e contribui para a segurança nas partições.

Uma das configurações possíveis é a opção **nosuid** que não permite que programas tenham acesso no nível *root*. Outra opção é a utilização do **noexec** que não permite a execução de qualquer binário ou arquivo executável dentro da partição.

Muitos usuários mal intencionados podem se aproveitar do diretório */tmp*, onde se pode introduzir *backdoors* ou outro programa malicioso para ter acesso completo ao sistema (CORREIA et al, 2006).

Na inicialização do sistema, o Linux monta todos os sistemas de arquivos disponíveis pelas especificações estabelecidas em */etc/fstab*, caso deseja-se modificar deve-se alterar a configuração do arquivo **fstab**. A Tabela 5, apresenta uma sugestão de configurações para os diretórios.

Tabela 5. Configuração das partições no /etc/fstab

Dispositivo	Ponto de Montagem	Sistema de Arquivo	Opções de montagem	*freq*	*passno*
/dev/hda1	/boot	ext3	defaults, nosuid	0	2
/dev/hda3	/	ext3	Defaults	0	1
/dev/hda4	/home	ext3	defaults, nosuid,noexec	0	2
/dev/hda5	/usr	ext3	defaults, nosuid	0	2
/dev/hda6	/tmp	ext3	defaults, nosuid,noexec	0	2
/dev/hda7	/var	ext3	defaults, nosuid,noexec	0	2
/dev/hda2	None	Swap	Sw	0	0
/dev/hdb	/media/cdrom	iso9660	ro,user,noauto	0	0
/dev/fd0	/media/floppy	Auto	rw,user,noauto	0	0

Fonte: CORREIA, I. et al (2006)

Com essas opções ativadas, ao se instalar um novo pacote com o **apt-get** ou **dpkg** não será possível, pois esses utilitários executam e gravam informações nos diretórios */var* e */tmp* que foram configurados com a opção **noexec**. Para alterar isto, deve-se alterar a configuração ou criar um *script* que altere a configuração (CORREIA et al, 2006).

5.3.2 Instalação mínima

Com relação à "instalação mínima", a norma NBR ISO/IEC 17799 tece considerações no item 11.5.4, acerca da restrição e controle do uso de programas utilitários. Portanto, deve-se instalar o mínimo possível de *softwares* para que o sistema possa funcionar.

Um sistema mais seguro começa pela instalação do mínimo possível de pacotes e componentes, observando-se fundamentalmente o propósito do sistema em questão e do ambiente de rede no qual está inserido. Uma instalação completa, por exemplo, do Red Hat Enterprise Linux contém mais de 1000 aplicações e pacotes de bibliotecas (REDHAT, 2005).

É comum que serviços não utilizados não sejam monitorados por falhas de segurança, o que aumenta a possibilidade de não ser aplicada uma correção devidamente necessária. A redução do número de pacotes instalados diminui a chance do sistema possuir vulnerabilidade a ser explorada por um atacante (NBSO, 2003).

Algumas distribuições permitem que ao administrador escolher entre a instalação típica e especializada. Quando possível, deve-se optar pela personalizada, evitando-se instalar componentes cuja função não seja conhecida ou que não necessite no momento. Em outras, a instalação se dá em duas etapas: a instalação do sistema base (sobre o qual o administrador tem o mínimo de controle) e a instalação de pacotes ou componentes adicionais (NBSO, 2003).

Independente do controle utilizado na instalação a etapa seguinte, descrita no item 5.3.3, é muito importante e deve ser realizada com a maior atenção, pois permite complementar a presente etapa enunciada nesse item.

5.3.3 Desabilitar Serviços Desnecessários

Com relação à “desabilitar serviços e softwares desnecessários”, a norma NBR ISO/IEC 17799 determina, no item 11.5.4, a conveniência da remoção ou desabilitação de todos os *softwares* utilitários e de sistemas desnecessários.

O Linux difere de outros sistemas operacionais, pois nenhuma entidade individual controla o desenvolvimento e os testes dos *softwares*. Muitos desses programas são derivados de empresas independentes, acadêmicas e de desenvolvedores situados em qualquer lugar do mundo. Portanto, cada desenvolvedor é responsável pelo próprio controle de qualidade da segurança do aplicativo, e tal processo pode variar completamente de desenvolvedor para desenvolvedor (SEGURANÇA MÁXIMA PARA LINUX, 2000).

Outra questão a ser levada em consideração é a existência de código aberto na maioria dos programas, permitindo a qualquer usuário conhecedor de programação analisar o código fonte em busca de vulnerabilidades.

Para desabilitar os serviços desnecessários deve-se executar quatro etapas, quais sejam: verificação dos serviços instalados, dos serviços realmente necessários, da existência de dependência, e, por fim, desinstalar ou desabilitar os serviços.

Na verificação de quais programas estão instalados usa-se o comando **dpkg**, após isso, deve ser realizada a análise completa dos programas instalados por padrão, verificando-se o que é realmente necessário. Por exemplo, se o servidor não oferecer um serviço de correio, desative-se o Sendmail, pois tal serviço é um dos serviços de rede mais visados para se efetuar um ataque (TERPSTRA, 2005).

Às vezes, determinado serviço requer que um ou mais *daemons*[26] sejam executados. Por exemplo, o serviço Samba precisa que três *daemons* sejam executados: smdb, nmdb e winbindd. Isso significa que antes de desinstalar deve-se verificar se um serviço contém pré-requisito ou dependências (TERPSTRA, 2005). Pode ser utilizado o próprio comando **dpkg** ou ler o manual do programa. Para ter acesso ao manual utiliza-se o comando **man**.

[26] É um processo executado sem um console associado

Após determinar os serviços desnecessários ao servidor é desejável que os desinstalem ou não os ativem, caso se queira manter o serviço para uso futuro. Para removê-lo, de forma mais simples, usa-se os gerenciadores de pacotes, por meio do comando **apt –get**.

Para desabilitar os serviços deve-se compreender que Linux, os *scripts* de inicialização dos *daemons* estão no diretório */etc/init.d* ou no diretório */etc/rc.d*, dependendo da distribuição usada. Alguns serviços são executados como seu próprio *daemon* e outros são iniciados a partir de um *daemon* especial denominado *inetd* (distribuições SUSE e Debian) ou *xinetd* (RedHat). Para desabilitar um *daemon* de serviço utiliza-se a opção *stop* no arquivo de configuração do *script* de inicialização.

Um complemento de segurança que pode ser executado é a restrição do acesso ao diretório */etc/rc.d/init.d* permitindo apenas ao usuário *root* a permissão para ler, escrever e executar arquivos. Esse procedimento pode ser executado pelo comando **chmod**. Outra opção, no Debian, caso não queira editar renomeando ou removendo manualmente os *links* de */etc/rc.d*, é usar o comando *update-rc.d* (Peña, 2007).

Além do *init.d*, deve-se checar os serviços desnecessários no *inetd*, tais como o echo, chargen, discard, daytime, time, talk, ntalk, r-services os quais são considerados altamente inseguros (Peña, 2007). Para desativar serviços configura-se o */etc/inetd.conf* e acrescentando "#" (caractere de comentário) no início das linhas e depois executa-se o comando **killall** para iniciar o inetd. Outra alternativa, é utilizar o comando **update**, que facilita a tarefa de edição do arquivo inetd.conf.

Existem ainda várias ferramentas que alteram scripts de inicialização. A Tabela 6 lista as ferramentas disponíveis em algumas distribuições Linux.

Tabela 6. Ferramentas para alterar scripts de inicialização

Distribuição	Ferramentas de configuração de serviços
Debian	Rcconf chkconfig
Fedora (Red Hat)	Ntsysv serviceconf chkconfig
SUSE	YaST2 Chkconfig

Fonte: TERPSTA, J. (2005)

5.3.4 Instalações de Correções de Segurança

Com relação à "instalação de correções de segurança", comumente denominado *patch*, a norma NBR ISO/IEC 17799 diz, no item 12.4.1, que para garantir a segurança de arquivos de sistema deve ser realizada a atualização de *software* operacional, de aplicativos e de bibliotecas de programas somente pelos administradores e que os sistemas operacionais sejam atualizados quando existir requisito para tal.

Independente da escolha da distribuição, deve-se certificar em manter os programas atualizados para não executar versões antigas com erros ou falhas conhecidas que possam vir a ser explorados (HATCH; LEE; KURTZ, 2002).

A grande maioria dos fornecedores de *softwares* libera correções para problemas de segurança logo que sejam descobertos em um sistema, sem que se tenha de esperar pela sua próxima versão (NBSO, 2003).

Em primeiro lugar, o administrador deve constantemente acessar *sites* especializados em anúncios de segurança ou de preferência se inscrever em listas de discussão, dos sistemas que está utilizando para se manter devidamente atualizado. Isso permitirá baixar os novos pacotes com atualizações de segurança, as quais são muito importantes na manutenção de um sistema seguro. No Debian, por exemplo, para

receber importantes atualizações de segurança e alertas deve-se enviar um *e-mail* para debian-security-announce-request@lists.debian.org (PEÑA, 2007).

Muitas vezes algumas configurações do sistema são alteradas durante o processo de instalação de correções, dessa forma, é recomendável a revisão da configuração do sistema após instalar a correção, principalmente quanto à desativação de serviços. Recomenda-se, ainda, o registro que todas as instalações de correções no *logbook* (NBSO, 2003).

A melhor opção na atualização do sistema é por meio dos gerenciadores de pacotes. O repositório http:// security.debian.org que contém as atualizações de segurança. Caso se insere ou exclua um repositório deve-se editar o arquivo */etc/apt/souces.list*. Para atualizar o sistema recomenda-se usar, diariamente, o comando **apt-get update**, quando for instalar um pacote ou atualizar o sistema. O comando **apt-get upgrade,** quando necessitar atualizar o sistema como um todo (SILVA, 2006).

5.3.5 Configuração dos Serviços Utilizados

No quesito “configuração dos serviços utilizados”, a norma NBR ISO/IEC 17799, no item 12.4.1, destaca que para minimizar o risco de corrupção nos sistemas operacionais convém a presença de sistema de controle de configuração cuja finalidade é manter o controle do *software* e da documentação do sistema.

Os administradores que não sabem configurar seus sistemas de maneira apropriada são grandes ameaças à segurança dos servidores. Um exemplo real de má configuração de um serviço *proxy* que ocasionou um ataque ao Provedor Norte-Americano de Internet Excite@Home é relatado no livro a Arte de Invadir. Tal feito foi executado pelo conhecido *hacker* Adrian Lamo.

> Adrian descobriu um proxy mal configurado que abriu a porta para as páginas internas da rede de vários departamentos da Excite@Home. Na seção "Ajuda" de um, ele fez uma pergunta sobre problemas para fazer a conexão. Na resposta veio o endereço URL de uma pequena parte do sistema que auxiliava na resolução de problemas de TI. Ao analisar esse URL, ele conseguiu acessar divisões da empresa que usavam a mesma tecnologia. Não lhe pediram autenticação: o sistema tinha sido projetado supondo-se que qualquer um que soubesse solicitar endereços para essas partes do site Web deveria ser um funcionário ou outra pessoa autorizada — uma premissa incerta tão divulgada que ganhou um apelido: *segurança pela obscuridade* (MITNICK, 2005, p. 81).

Segundo GUAZELLI (2005) a melhor oportunidade para se configurar um sistema é logo após a instalação, sendo se possível, com a máquina fora da rede, visando minimizar a exposição do servidor.

A configuração adequada de cada serviço dependerá de sua finalidade e do nível de segurança que se pretende atingir. Recomenda-se analisar as documentações disponíveis em cada serviço alterando as configurações originais e registrando-as no *logbook* (NBSO, 2003).

Muitos serviços são desenvolvidos sob suposição de que são utilizados em uma rede confiável. Porém a afirmação deixa de ser verdadeira a partir do momento em que o serviço for disponibilizado por meio da *Internet*. Alguns serviços que necessitam atenção em sua configuração, sob a ótica da segurança, são: Portmap, NIS, NFS, Apache, FTP e SendMail (REDHAT, 2005). Peña (2007) acrescenta ainda o ssh, squid, BIND, Finger.

Em suma, o objetivo da elaboração dos controles na instalação e na implantação do servidor é estabelecer uma base mínima de segurança desde a sua raiz. Procedimentos sugeridos tais como a montagem equilibrada e coerente das partições de um HD; a análise sobre quais os serviços e softwares que são essenciais ao servidor, para que se evite acesso ao servidor por meio de falhas existentes nesses componentes e a preocupação constante de se manter o sistema atualizado e corretamente configurado

contribuirão, em conjunto com outros controles, para a preservação das propriedades fundamentais da segurança da informação sejam preservadas.

5.4 POLÍTICAS DE CONTROLES DE ACESSO AO SERVIDOR

A obtenção de acesso é um dos principais objetivo de um usuário mal intencionado. Para obtenção de acesso pode-se entender que ele vai necessitar de um nome de usuário e de uma senha válida no sistema, sendo que esse nome de usuário não obrigatoriamente seja o do *root*, superusuário que possui controle total de todo o sistema, pois muitos ataques, utilizando-se de *rootkits*, podem ser efetuados para se obter a elevação de privilégio (HIJAZI; MAZZORANA; RAVANELLO, 2004).

Quando se fala em segurança de servidores, normalmente, acredita-se que os principais ataques deverão ser remotos, porém isto não é uma verdade absoluta, pois um funcionário mal intencionado pode ter acesso físico ao servidor, diretamente no teclado do próprio servidor e realizar procedimentos que afetará a segurança da informação (CORREIA et al, 2006). O Item 5.2 tratou de alguns controles para se evitar que se acesse fisicamente o servidor.

Segundo a RFC 2196, uma política de acesso deve definir os direitos de acesso e privilégios a fim de proteger a organização de danos na informação, especificando diretrizes de uso aceitáveis para os usuários do sistema. Portanto, serão descritos alguns procedimentos que visam a controlar o acesso, quer remoto ou quer local, objetivando-se coibir o acesso indevido aos sistemas. Serão estabelecidos os seguintes controles: política de uso de senha, política de gerenciamento de usuários, política de gerenciamento de privilégios e permissões de acesso, controle de acesso ao sistema operacional e política de acesso remoto.

5.4.1 Política de uso de senha

No quesito “política de uso de senha”, a norma NBR ISO/IEC 17799 descreve, no item 11.2.3, que a concessão de senhas deva ser controlada por meio de processo de gerenciamento formal. Complementa ainda, no item 11.3 que os usuários sejam solicitados a seguir boas práticas de segurança da informação na seleção e no uso de senhas.

As senhas são utilizadas pela grande maioria dos sistemas de autenticação e são consideradas necessárias como meio de autenticação (SPANCESKI, 2004). A segurança da senha é tão crítica que sem ela o sistema nunca estará seguro. De fato, poder-se-ia instalar vários *firewall* e ainda, se as senhas fossem vulneráveis haveria vulnerabilidade no sistema (SEGURANÇA MÁXIMA PARA LINUX, 2000).

A RFC 2196 estabelece que para a criação de uma política de senha consistente deve ser observado as seguintes condicionantes: a importância de senhas robustas, a troca das senhas padrão, a restrição ao acesso do arquivo de senhas, o período de duração das senhas e a possibilidade de bloqueio de senhas após um número determinado de autenticações inválidas.

SPANCESKI (2004) cita alguns cuidados especiais na escolha e no uso das senhas:

a) não utilização de palavras que estão no dicionário (nacionais ou estrangeiros);

b) não utilização de informações pessoais fáceis de serem obtidas, tais como o número de telefone, nome da rua, nome do bairro, cidade, data de nascimento, placa do carro;

c) não utilização de senhas somente com dígitos ou com letras;

d) utilização de senhas com pelo menos oito caracteres;

e) mistura de caracteres maiúsculos e minúsculos, de preferência misturando números, letras e caracteres, incluindo pelo menos um caractere especial e que não haja repetição do mesmo caractere;

f) não anotação de senha em papel ou em outros meios de registro de fácil acesso;

g) não utilização do nome do usuário, primeiro nome ou sobrenome, nome de pessoas próximas, como esposa, filhos, e amigo e de animais de estimação;

h) não fornecimento da senha para ninguém. A lembrança de que a senha é pessoal e intransferível é de fundamental importância;

i) validação das senhas por, no máximo, noventa dias, por exemplo.

A escolha da senha deve permitir a sua utilização posterior, pois não será válida se o usuário criar senha robusta e posteriormente não conseguir lembrá-la. Podem ser usados métodos acrônimos[27] para memorizar a senha. A Figura 8. apresenta um método de criação de senha utilizando-se um acrônimo:

- Frase:
  →"O Sol da liberdade em raios fúlgidos brilhou no céu da pátria neste instante"
- Transformando um acrônimo (primeira letra de cada palavra):
  → **osdlerfbncdpni**
- Adicione a complexidade, substituindo letras por números e símbolos no acrônimo, substitui letra n por 9, letra d por arrouba (@), letra e por 7 :
  →**os@l7rfb9c@p9i**
- Adicione mais complexidade, pelo menos um das letras em caixa alta:
  →**Os@l7rfb (senha oito dígitos)**

Figura 8. Criação de senha usando acrônico
Fonte: REDHAT (2005)

[27] É um agrupamento das letras iniciais de várias palavras.

Um primeiro passo na política de senha é configurá-la para a BIOS. O procedimento citado pode impedir que usuários não autorizados e que obtiveram acesso físico ao servidor, possam inicializar a máquina com mídia removível ou obter privilégios *root* por meio do modo usuário simples (REDHAT, 2005).

Outro passo a ser elaborado é a configuração da senha nos gerenciadores de *boot*. No LILO deve editar o arquivo */etc/lilo.conf* e no GRUB o arquivo */boot/grub/grub.conf*.

No Linux, as senhas são armazenadas de forma criptografada, armazenando-se as senhas já modificadas, na maioria das versões iniciais, no arquivo */etc/passwd*, sendo este visualizado por qualquer usuário (HATCH; LEE; KURTZ, 2002). Com o objetivo de aumentar a segurança as senhas passaram a ser ocultas, sendo armazenadas em */etc/shadow*. Para verificar se está instalado deve-se examinar o */etc/passwd.* Se, porventura, contiver senhas criptografas no segundo campo, é sinal que o pacote não está instalado.

Outro procedimento de segurança é estabelecer datas de expiração para contas de usuários. O comando **passwd** permite especificar datas de expiração para as contas do usuário (SILVA, 2006).

Algumas ferramentas podem ser úteis para verificar se as senhas estão fracas. Os programas são chamados de programas de decrifação de senhas devendo ser usados no processo de auditoria das senhas escolhidas. Pode-se citar: o Crack, Jhon the ripper e Slurpie (HATCH, LEE; KURTZ, 2002).

.

5.4.2 Política de Gerenciamento de Usuários

No que diz respeito à “política de gerenciamento de usuários”, a norma

NBR ISO/IEC 17799 descreve, no item 11.2, a conveniência da implementação de procedimentos formais para controlar a distribuição de direito de acesso a sistemas de informação e serviços.

O sistema operacional Linux é um sistema multiusuário, podendo haver a necessidade da conexão de mais de um usuário ao servidor. A depender do servidor pode-se necessitar de mais de um administrador, por exemplo, para administrar servidores em período integral (24 horas) com a divisão de trabalho por turno de serviço. O conhecimento dos tipos de usuários e a forma de gerenciá-los são essenciais para a segurança do sistema (HATCH, LEE; KURTZ, 2002).

O item 11.2.1 da NBR ISO/IEC 17799 sugere alguns controles úteis para registrar e cancelar os usuários:

> a) utilizar identificador de usuário único para assegurar a responsabilidade de cada um. Convém a permissão de uso de grupos, onde exista a real necessidade para a atividade;
>
> b) verificar se o usuário tem autorização do responsável para usar o sistema;
>
> c) verificar se o nível de acesso concedido ao usuário é devidamente apropriado;
>
> d) conceder aos usuários declaração por escrito dos seus direitos de acesso;
>
> e) manter registro formal de todas as pessoas registradas para usar o serviço;
>
> f) remover imediatamente ou bloquear direitos de acesso de usuários que mudarem de cargo ou deixaram a organização.

As informações sobre todos os usuários em uma máquina Linux são armazenadas no arquivo */etc/passwd*. Para criar um usuário deve-se usar o comando

adduser. Por padrão será criado um diretório */home* como o nome do usuário. Para excluir um usuário usa-se o comando **userdel.**

Um grupo de usuário no Linux é um conjunto de um ou mais usuários. Pode ser conveniente reunir vários usuários para definir as suas propriedades como um grupo assim como o controle sobre o que podem ou não acessar. Os grupos são definidos no arquivo */etc/group*. Para criar um grupo usa-se o comando **addgroup** e para excluí-lo usa-se o comando **groupdel** (HATCH, LEE; KURTZ, 2002).

5.4.3 Gerenciamento de Privilégios e Permissões de acesso

Com relação ao "gerenciamento de privilégios e permissões de acesso", a norma NBR ISO/IEC 17799, no item 11.2.2, estabelece a restrição e o controle quanto à concessão e ao uso de privilégios. Complementa, ainda, no item 11.6.1 que a restrição ao acesso à informação e a funções dos sistemas de aplicações por usuários e pessoal de suporte seja restrito de acordo com o definido na política de controle de acesso.

Ao se falar em privilégio no Linux compreenda-se o conceito de *root*, usuários, grupos e restrições de acesso. Algumas razões levam a criação de usuários ou grupos, buscando-se usar o *root* quando for absolutamente necessário.

Uma das razões está relacionada à viabilidade da conta *root* em obter o poder absoluto. Isso poderá causar danos irreparáveis quando utilizada de maneira indiscriminada ou inadvertidamente. Outra razão está na possibilidade de se abrir o sistema para inúmeras ameaças de segurança, como no caso de o usuário *root* conectar-se a um navegador que poderá processar um miniaplicativo de Java herdando privilégios de acesso, utilizando-o para atacar o sistema (SEGURANÇA MÁXIMA PARA LINUX, 2000).

Uma vez que os privilégios de nível *root* sejam alcançados, será possível reforçar essa posição pela modificação de áreas do sistema por meio de outras falhas de segurança e de técnicas furtivas para ocultar o fato pelo qual foi comprometido (HATCH, LEE; KURTZ, 2002).

Outro ponto que deve ser levado em consideração são os arquivos com permissão Suid bit[28]. A Permissão de arquivos Suid bit**,** utilizado, por exemplo, pelo comando **su, ping e passwd,** possibilita a determinado programa que só pode ser executado pelo usuário root seja executado por qualquer outro usuário comum do sistema. O problema pode ser de grande monta se um usuário mal intencionado souber aproveitá-lo, explorando uma vulnerabilidade (CORREIA et al, 2006).

Portanto, gerenciamento de privilégios e permissão é a capacidade de se administrar corretamente o nível *root* e habilitar permissões de arquivos e diretórios ao demais usuários.

A conta *root* deve ser usada para administrar o sistema e deve ser usada no menor tempo possível. Para o usuário se tornar *root* deverá usar o comando **su**.

Algumas restrições ao nível de *root* podem proporcionar algumas melhorias na segurança, caso o administrador não permita a autenticação por parte de todos usuário, tais como: limitação e desativação de acesso *root* a usuários selecionados e limitação às permissões de *Suid bit* de alguns comandos.

Para limitar o acesso *root* a usuários selecionados, algumas distribuições ativam o suporte ao grupo *wheel*, que só os membros do grupo têm permissão para executar o comando **su** (HATCH, LEE, KURTZ, 2002). Usa-se o comando **chgrp** para mudar o diretório */bin/su* para o grupo *whell*. Havendo a necessidade de limitar o acesso, usa-se o comando **usermod** para adicionar o usuário a esse grupo (REDHAT,

[28] Possibilita que um determinado binário que só possa ser executado pelo usuário *root* seja executado por qualquer outro usuário comum do sistema

2005).

Pode haver, também, a necessidade de impedir os usuários de se autenticarem diretamente como *root. O* administrador de sistema pode configurar a *shell* da conta *root* editando no arquivo */etc/passwd* e alterando de */bin/bash* para */sbin/nologin*. Isto impede o acesso à conta *root* por meio de comandos que requerem uma *shell*, como os comandos **su** e **ssh**, porém ainda permite que clientes FTP, de *e-mail* e o **sudo** acessem a conta *root* (REDHAT, 2005).

Complementando, pode-se bloquear o *loggin* do *root* em todos os terminais de texto, e quando for necessário usar o comando **su** para adquirir o status de *root*. Para realizar essa tarefa deve-se editar o arquivo */etc/securetty* e comentar as linhas referentes aos terminais (Exemplo: # tty1).

Outra necessidade que poderá ocorrer é a retirada da permissão *Suid bit* de algum binário. Para tal usa-se o comando **find**, para localizar as permissões de *Suid bit* e o **chmod** para alterar as permissões.

O comando **sudo** oferece outra opção para se dar o acesso administrativo a usuários. O **sudo** permite que se limite estritamente quais usuários podem chamá-lo e que comandos eles podem executar. Essas configurações estão disponíveis em */etc/sudoers* (CORREIA et al, 2006).

Ao estabelecer permissões de arquivos ou diretórios no sistema de arquivos busca-se colocar controles sobre os usuários, possibilitando que se restrinja o acesso apenas a quem for autorizado. Os modos válidos são leitura (r), gravação (w), execução (x) (HATCH; LEE; KURTZ, 2002). Para identificar as permissões usa-se o comando **ls** e para alterar usa-se o comando **chmod.**

5.4.4 Controle de Acesso ao Sistema Operacional

Com relação ao “controle de acesso ao sistema operacional”, a norma NBR ISO/IEC 17799 expõe, no item 11.5, a utilização de recursos de segurança da informação para restringir o acesso Os recursos englobam autenticação de usuários e restrição do tempo de conexão, quando apropriado.

Apesar de todas as precauções contra ataque externo e com os controles implementados quanto à segurança física não se deve deixar de estabelecer procedimentos que protejam o servidor contra possível tentativa de uma invasão *in loco.*

CORREIA et al (2006) cita um exemplo, envolvendo um administrador de servidor que desprevenido resolveu um problema fora da sala e sem perceber deixou o terminal do servidor logado, como usuário *root*. Nesse caso específico, qualquer pessoa presente no recinto poderia tomar uma atitude hostil contra o sistema.

Para se defender dessa possibilidade algumas medidas simples podem ser tomadas:

a) desabilitar o uso de CTRL + ALT + DEL;

b) limitar o uso de terminais;

c) bloquear o terminal quando permanecer inativo; e

d) utilizar um programa de autenticação de usuário (PAM).

Desabilitar o CTRL + ALT + DEL no sistema Linux impedirá qualquer pessoa a pressionar a seqüência de teclas evitando-se, assim, a reinicialização do servidor. Tal situação é comum acontecer quando se pressiona as teclas pensando que é o do WINDOWS (Correia et all, 2006). Deve-se editar o arquivo */etc/inittab* e comentar a linha referente ao comando e depois atualizar o arquivo com o comando **init q**.

Caso se queira permitir somente que alguns usuários desliguem o sistema, deve-se criar o arquivo */etc/shutdown.allow* e incluir nele os nomes de usuários autorizados a reiniciar o sistema.

Outra opção de segurança é limitar o uso dos terminais texto, pois dependendo do caso, por questões de segurança, não é de bom alvitre deixar o *login* habilitado em todos os terminais texto (CORREIA et al, 2006). Deve-se editar o arquivo */etc/inittab* e comentar a linha do terminal que se deseja impedir o *login* e atualizar o arquivo após o procedimento.

O item 11.5.5 da NBR/ISO 17799 ressalta a importância de desconectar terminais inativos após determinado período de inatividade, cujo procedimento, no Linux, pode ser feito por um programa, como por exemplo com o Vlock, ou pelo uso da variável TMOUT. A variável que não vem selecionada por *default*, tem por finalidade controlar o tempo para o *logout* do terminal. Para indicar o valor de quantos segundos permanecerá ativo, é necessário atribuir seu valor, porém é necessário fazer tal procedimento toda vez que iniciar o uso do terminal. Visando melhor performance pode-se editar o arquivo */etc/profile* inserindo o (TMOUT= 180), por exemplo, se for um intervalo de três minutos (CORREIA et al, 2006) no final do arquivo.

Para auxiliar os métodos de autenticação tradicionais possibilita a execução de funções adicionais convêm usar os Módulos de Autenticação Plugáveis (PAM) (CORREIA et al, 2006). A ferramenta possui variadas opções para gerenciamento de autenticação, gerenciamento de contas, gerenciamento de seção e gerenciamento de senha (SOUZA, 2001).

O PAM tem suporte a muitos programas e já vem disponível em várias distribuições. Para verificar se possui suporte a um determinado programa use o comando **ls.**

Dentre as opções disponíveis, pode-se controlar o horário em que o usuário pode “logue” no servidor, seja remota ou localmente, representando, ser um excelente recurso de segurança (SOUZA, 2001). O PAM possui o arquivo */etc/security/time.conf*, onde realiza-se restrição quanto ao dia, hora e comandos permitidos.

Outra opção, é a possibilidade de limitar a fim de que utilize somente um terminal. Para tanto, deve-se editar o arquivo *limits.conf* (CORREIA et al, 2006).

Outras configurações disponíveis para o PAM podem ser encontradas no *site* primário de distribuição (http://www.kernel.org/pub/linux/libs/pam/). O objetivo das descrições relatas acima foi apresentar alguns recursos usados para se estabelecer restrições (PENA, 2007). Alguns módulos estão descritos na Tabela 7.

Tabela 7- Módulos de autenticação alteráveis (PAM)

Módulo	Finalidade
pam_unix.so	Fazer o policiamento padrão, visa deixar o linux mais seguro
pam_deny	Responsável pela bloqueio quando uma autenticação falhou
pam_warn	Envia aviso de falha ao syslog
pam_wheel	Usuários que terão acesso a root
pam_limits	Restringe os recursos do sistema que os usuários tem permissão
pam_time	Estabelece o controle de acesso por tempo
pam_cracklib	Adiciona verificação de senha
pam_group	Atribui e monitora membros de grupo

Fonte: PEÑA, J. (2007)

5.4.5 Política de acesso remoto

No quesito “política de acesso remoto”, a norma NBR ISO/IEC 17799 descreve, no item 11.4.2, os métodos apropriados de autenticação, utilizando-se da técnica de criptografia, *hardware tokens* ou de um protocolo de desafio/resposta para controlar o acesso de usuários remotos.

O item 11.7.2, complementa, que convém às organizações somente a autorização para atividades de trabalho remoto apenas se elas estiverem certas de que as

providências apropriadas e os controles de segurança estão implementados e que estão de acordo com a política de segurança da organização.

O acesso remoto é a capacidade de obter acesso a um computador ou uma rede a partir de distância remota. Pode ser necessário ao administrador no caso de acessar o servidor quando estiver ausente do local de trabalho. Estes dados transmitidos pelos usuários remotos são de alta confidencialidade, necessitando também como requisito de segurança a integridade (CORDEIRO; MOREIRA, 2002).

Os *sniffers* representam risco significativo de segurança no acesso remoto, principalmente porque eles não são detectados facilmente e, portanto, possibilitam a captura dos dados que trafegam na rede (SEGURANÇA MÁXIMA PARA LINUX, 2000).

Ao criar e implementar uma solução de acesso remoto, o usuário deve se certificar de planejar a implantação mais segura possível. Monteiro (2003) ressalta os seguintes aspectos que devem contar na política de acesso remoto:

a) desabilitar, se for o caso, os *logins* remotos aos finais de semana e durante a noite, por exemplo as portas FTP e Telnet; e

b) fazer um levantamento de quem realmente necessita realizar um login remoto e habilitar o serviço que este usuário necessita somente naquela ocasião.

Deve-se usar preferencialmente o Ssh, que implementa criptografia, ao invés do Telnet.

A Tabela 8 abaixo apresenta modificações que devem ser realizadas no arquivo *sshd_config* no diretório */etc/ssh* para torná-lo mais seguro (PEÑA, 2007).

Tabela 8. Configuração mais segura para o ssh

Configuração	Finalidade
ListenAdress 192.168.0.1	Especifica que o ssh somente funcionará na interface especificada, caso tenha mais de uma interface ou em caso de adição de uma futura interface de rede
PermitRootLogin no	Tenta não permitir o *login* do usuário *root* sempre que possível. Se alguém quiser se tornar o usuário ssh, desta maneira agora dois *logins* são necessários e o ataque de força bruta não terá efeito no *root* via ssh
Listen 666	Altera a porta do programa, assim o intruso não terá completa certeza de onde o *daemon* sshd é executado
PermitEmptyPasswords no	Senhas em brancos tornam o sistema inseguro
AllowUsers usuario ref me@unesc.net	Permite que alguns usuários terão acessos via ssh
AllowGroups wheel admin	Permite somente membros de certos grupos de terem acesso ao ssh no servidor
PasswordAuthentication no	É mais seguro somente permitir o acesso a usuários com chaves ssh colocadas em /.ssh/authorized_keys
-	Desativar outras formas de autenticação que realmente não precisa, se não usar, por exemplo RhostsRSAuthentication, KerberosAuthentication, entre outros
Protocol 2	Desativar o protocolo versão 1, pois representa alguns problemas de designer que torna fácil a descoberta de senhas
Banner /etc/some_file	Adicionar um *banner* para usuários se conectando ao servidor ssh, em alguns países o envio de aviso antes de acessar um determinado sistema alertando sobre acesso não autorizado deverá ser emitido para ter proteção legal
-	Pode-se restringir o acesso ao servidor ssh usando o pam_listfile ou pam_wheel
Porta 22	Alterar a porta 22 padrão que está muito conhecida e utilizada na tentativa de ataque ao servidor. A nova porta deve ter um aporta acima de 1024.

Fonte: PEÑA, J. (2007)

Portanto, buscou-se nesse item estabelecer diversos controles com a finalidade de restringir e limitar ao máximo o acesso apenas aos usuários autorizados. A elaboração de uma boa política de uso de senha, de política de gerenciamento de usuários e de política de gerenciamento de privilégios e de permissões de acesso, dificultam sobremaneira os usuários mal intencionados de atingirem seus objetivos quando atacam o sistema. Além da barreira estabelecida pela segurança física, complementada pelo controle de acesso ao sistema operacional, os acessos indevidos *in loco* no servidor são dificultados.

Por fim, a análise criteriosa da necessidade de se executar um trabalho remoto, associada à utilização de protocolos remotos que utilizam criptografia contribuirão para que os dados sejam mantidos em proteção adequada.

5.5 CONTROLES DE PRESERVAÇÃO DA DISPONIBILIDADE DA INFORMAÇÃO

Sêmola (2003 apud CACIATO, 2004) destaca que a informação gerada ou adquirida por um individuo ou instituição deve estar sempre disponível aos seus usuários no momento em que os mesmos delas necessitem para qualquer finalidade. As informações, armazenadas nos servidores possuem alto valor, valendo-se mais do que os próprios computadores. Proteger essas informações é uma das tarefas mais importantes do administrador de sistema (NEMETH, 2002).

Nesse contexto, a dependência tecnológica pode trazer grandes problemas, pois se a organização não possuir bom planejamento para contornar, interrupção em um serviço, causada pela falta ou funcionamento inadequado de algum componente de suporte ao processo, as perdas podem tornar-se irreversíveis (FRANCISCO, 2004).

Os motivos que podem causar essa interrupção vão desde ações nocivas de humanos, eventos incontroláveis da natureza, equívocos inocentes ao sistema operacional, até falha mecânica e *bugs* de *software* (SEGURANÇA MÁXIMA PARA LINUX, 2000).

Para manter a disponibilidade dos serviços prestados, podem ser utilizadas estratégias de tolerância a falhas, previstas nos **Planos de Contigência**, possibilitando que um servidor assuma os serviços de outro servidor que venha a falhar, além da realização periódica de ***backups*** (DUMONT, 2006).

5.5.1 Política de *Backup*

No quesito "política de *backup*", a norma NBR ISO/IEC 17799 esclarece, no item 10.5, que para manter a integridade e disponibilidade da informação e dos recursos de processamento de informação convém que sejam estabelecidos procedimentos de rotina para a geração de cópias de segurança com a finalidade de possibilitar a geração das cópias de segurança dos dados com vista que haja a recuperação dos dados em um tempo aceitável.

A perda da informação, conforme já citado, pode trazer transtornos para a organização, sendo muita das vezes, difícil de ser substituída. Uma vez que houve a perda da informação por ataque ou por outro motivo, o administrador do servidor, a partir das informações coletadas, poderá restaurar o que foi perdido ou alterado e restabelecer a integridade do sistema. Se corretamente executados, os *backups* permitirão ao administrador restaurar seus sistemas de arquivos à condição que estavam quando o *backup* foi realizado pela última vez (NEMETH, 2002).

A importância dos *backups* na administração de sistema nunca pode ser minimizada. A lista de quais arquivos devem ser realizados os *backups* dependerá do servidor, mas geralmente, inclui os dados, os arquivos de configuração e os *logs*, sendo que os arquivos binários (executáveis e bibliotecas) devem ser geralmente evitados, com exceção a essa regra, da inclusão de cópia completa do sistema logo após a sua instalação, antes que ele seja colocado em rede (NBSO, 2003).

Quanto mais freqüentemente os *backups* forem feitos, menor será a quantidade de dados que poderá ser perdida em uma queda. Entretanto, deve-se fazer um estudo da periodicidade dos *backups,* pois tais processos utilizam recursos do sistema e tempo do administrador. Em sistemas sobrecarregados, é geralmente

adequado fazer o *backup* dos sistemas de arquivos todos os dias úteis. Nos demais sistemas, a periodicidade dependerá do tipo de serviço, podendo ser várias vezes por semana ou até em períodos maiores (NEMETH, 2002).

Visando garantir a maior segurança no armazenamento das mídias dos *backups*, é essencial seu armazenamento em local protegido e distante do ambiente de produção, em função de qualquer incidente que venha a ocorrer (inundação, incêndio e até roubo) (BARROS, 2007).

a) o acesso ao local deve ser restrito, para evitar que pessoas não autorizadas roubem ou destruam os *backups*;

b) o local deve ser protegido contra agentes nocivos naturais (poeira, calor, umidade, dentre outros);

c) o local, se possível, deve ser a prova de fogo.

Uma vez que vários tipos de falhas podem danificar várias partes do *hardware* de uma vez, os *backups* devem ser gravados em algum tipo de mídia removível, a qual varia de acordo com a capacidade e com o armazenamento e velocidade de processamento. O tipo de mídia adequado dependerá do tipo de arquivos e da quantidade a ser armazenada (NEMETH, 2002).

Cordeiro e Moreira (2002) complementam que as etiquetas das mídias de *backup* devem conter o dia e horário da semana, devendo ser realizado, com uma periodicidade mensal, uma verificação dos *backups* com o objetivo de analisar se os dados gravados estão válidos.

A Norma NBR ISO/IEC 17799, além dos procedimentos já citados acima, destaca que:

a) as cópias de segurança necessitam de registros completos e exatos com documentação apropriada sobre os procedimentos de restauração da informação;

b) as cópias de segurança devem ser armazenadas em uma localidade remota, a uma distância suficiente para escapar dos danos de um desastre ocorrido no local principal;

c) as cópias de segurança devem ser protegidas por meios de encriptação onde a confidencialidade é importante.

Vários comandos podem ser utilizados para *backup* e *restore*[29], entre eles, ***tar***, ***cpio***, ***dump*** e ***restore***. Na verdade o **tar** não é compactador e sim "arquivador" (ele junta vários arquivo em um só), mas pode ser usado em conjunto com um compactador (como o gzip ou zip) para armazená-los compactados (DUMONT, 2006).

O **Dump** é um utilitário para agendar *backups* completos[30] ou incrementais[31]. Sua principal característica é realizar *backups* apenas nos arquivos alterados em relação ao último *backup*. Permite, ainda, gravar dados sem se preocupar com o comprimento dos nomes dos arquivos. Para restaurar os *backups* utiliza-se o **restore**. Outro sistema bastante útil, utilizado na realização de *backups* é o AMANDA que permite realizar *backups* de todas as máquinas em rede local (NEMETH, 2002).

5.5.2 Plano de Contigência

Com relação ao plano de contigência ou "plano de continuidade dos serviços", a norma NBR ISO/IEC 17799 descreve, no item 14, que a elaboração do referido plano visa a não permitir a interrupção do negócio e a proteger os processos

[29] Serve para restaurar um arquivo compactado
[30] Fazem backup apenas daqueles arquivos que mudaram desde o ultimo backup completo.
[31] Fazem backup de tudo na unidade de disco rígido.

críticos contra efeitos de falhas ou desastres significativos, e a assegurar a sua retomada em tempo hábil, se for o caso. Complementa, ainda, que o processo do Plano de Continuidade dos Serviços seja implementado para minimizar um impacto sobre a organização e recuperar perdas de ativos da informação a um nível aceitável por meio de ações de prevenção e recuperação.

Francisco (2004) destaca que os principais objetivos a serem atingidos pela Política de Continuidade de Serviços são:

a) garantir a segurança dos empregados e visitantes;

b) minimizar danos imediatos e perdas numa situação de emergência;

c) assegurar a restauração das atividades, instalações e equipamentos o mais rápido possível;

d) assegurar a rápida ativação dos processos de negócio críticos;

e) fornecer conscientização e treinamento para as pessoas-chave encarregadas desta atividade.

Segundo Pereira (2005) as falhas não podem ser restritas somente aos servidores, mas podem ser afetadas por outros meios, tais como: pelos recursos da rede (*hubs*, *switchs*, cabos, roteadores, entre outros), pela rede elétrica e de comunicações e pela localização do servidor (sendo propícios a furacões, enchentes, terremotos, roubos entre outros motivos).

São inúmeros exemplos de casos reais em que a falta de planejamento adequado na continuidade dos negócios contribui para a suspensão do serviço. Wanderley (2005) cita exemplo ocorrido em 2001, na sede do Tribunal de Justiça do Estado do Tocantins, quando o CPD desse órgão pegou fogo. Naquela ocasião, vários conceitos de segurança não foram observados, o que implicou a paralisação do setor de por trinta dias, interrompendo assim toda a movimentação de processos do Órgão.

Portanto, todo sistema está, de forma direta ou indireta, suscepível a falhas, que podem ser contornadas, usando algumas técnicas na garantia da continuidade do serviço (PEREIRA, 2005). O Plano de contigência visa documentar tais técnicas.

De acordo com a NBR ISO/IEC 17799, o processo para a elaboração do plano de continuidade do negócio deve ser composto das seguintes etapas:

> a) entendimento dos riscos que a organização está exposta, no que diz respeito à sua probabilidade e impacto, incluindo a identificação e priorização dos processos críticos do negócio;
>
> b) identificação de todos os ativos envolvidos em processos críticos de negócio;
>
> c) entendimento do impacto que as interrupções terão sobre o negócio;
>
> d) consideração de contratação de seguro compatível, se for o caso, que possa ser parte integrante do processo de continuidade;
>
> e) identificação e consideração da implementação de controles preventivos;
>
> f) detalhamento e documentação de planos de continuidade alinhados com a estratégia estabelecida;
>
> g) testes e atualizações regulares dos planos e procedimentos implantados;
>
> h) garantia de que a gestão da continuidade do negócio esteja incorporada aos processos e à estrutura da organização.

Um plano de contigência consiste em procedimentos de recuperação definidos previamente, tendo por finalidade minimizar os impactos gerados sobre a organização (WANDERLEY, 2005). A NBR ISO/IEC 17799 define que o referido plano deve constar: a identificação da possível falha, a condição de ativação do plano, os procedimentos de recuperação e a designação do responsável pela execução de sua correção constando, quando necessário, o nome do funcionário suplente.

As atividades relacionadas à avaliação e controle de riscos devem ser as mais exatas possíveis, pois definem os cenários e o que podem afetar a organização.Serão determinados quais os possíveis danos relacionados a cada evento e quais as medidas necessárias para prevenir e reduzir os efeitos de uma perda (FRANCISCO, 2004).

Os planos de contingência variarão de acordo com o tipo específico de quebra de segurança e da infra-estrutura presente na organização (SYMANTEC, 2002). A tabela 9 apresenta algumas medidas a serem adotadas em um plano quanto à prevenção de falhas nos sistemas de suporte e na infra-estrutura.

Tabela 9. Medidas de Prevenção nos sistemas de Suporte e Infra-estrutura a TI

Tipo de Falha	Medidas
Falha de sistema HVAC[32]	Identificar os sistemas (elevadores, ar-condicionado, aquecimento central, ventilação, temperatura, entre outros) e avaliá-los quanto: - à sua conformidade com os parâmetros de projeto, observando a existência de sistemas proprietários; - a criticidade deste tipo de sistemas para o funcionamento da rede; - definir regras de utilização destes sistemas, de modo a não pôr em risco o funcionamento da empresa e a segurança dos usuários dos sistemas
Energia elétrica	- prever sistema alternativo de fornecimento de energia; - definir o período de autonomia para o sistema; - prover os recursos necessários para o funcionamento do sistema alternativo durante o período de autonomia pretendido; - identificar as áreas prioritárias para o abastecimento de energia.
Comunicações	- providenciar meios alternativos de comunicação para receber e transmitir as informações; - considerar a hipótese de antecipar processamentos e/ou reativar processos manuais;
Controle Ambiental	- alguns equipamentos necessitam, para o seu correto funcionamento, de determinadas condições de temperatura e umidade. Prevendo uma eventual falha nos mecanismos de controle e reposição dessas condições, deve-se: - criar meios alternativos para fornecer as condições mínimas de funcionamento; - definir períodos de funcionamento no sentido de minorar a degradação das condições ambientais.
Sistemas de combate a incêndios	- devem ser colocados em controle manual; - prever o eventual reforço de meios mecânicos de combate a incêndio.
Transportes	uma eventual falha ao nível dos transportes pode impossibilitar o acesso das pessoas ao seu local de trabalho, inviabilizando o funcionamento da organização: - Viabilizar formas de transporte alternativas.

Fonte:PINHEIRO, J. (2004)

Além dos riscos na infra-estrutura e no suporte os servidores podem estar

[32] Sistemas que utilizam a calefação, ventilação e ar condicionado

sujeitos a falhas de *hardware* ou de *software*. Segundo Weber (2001) a estratégia mais comum em sistemas tolerantes a falhas é o uso de redundância em diferentes níveis, ou seja possuir componentes duplicados que possam entrar em funcionamento após a falha ocorrida, podendo ser automatizada ou por ação de um administrador após sua detecção. A Tabela 10 relaciona alguns componentes básicos de um sistema de Informação que servem para o desenvolvimento de um plano de contigência.

Tabela 10 Componentes de um Sistema de Informação

CATEGORIA	COMPONENTES
Softwares	Configuração Versão Nível de atualização Customizações Fornecedores
Hardware	Configuração Espaço em disco para o S. O. MIPS utilizado Memória CPU Controladora de Disco Controladora de Mídia Magnética Mídia Magnética Fornecedores Infra-estrutura Robôs/Cilos/estantes
Redes/Teleprocessamento	Porta de controlador de Linha Circuito de comunicação Multiplexadores Concentradores Roteadores *Switch* *Hubs* *Bridge* Concessionárias

Fonte: VIGLIAZZI, D. (2002)

Dependendo do sistema pode haver a necessidade de que o serviço seja de alta disponibilidade[33], nesse caso, além da redundância de *hardware*, deve haver *softwares* instalados. Diversas soluções presentes no Linux auxiliam a tarefa: o Drbd tem como principal finalidade a de estar replicando as informações de um servidor em

[33] Consiste em tentar manter um serviço provido por um ou mais servidores o máximo de tempo disponível possível, onde que a queda de um dos servidores, seja suprima automaticamente por outro servidor, de tal forma que não interrompa a disponibilidade do serviço.

outro, utilizando como meio de transmissão a rede; o Heartbeat que é responsável pela verificação e tomada de decisões e o Mon é responsável pelo monitoramento dos serviços e pelo envio, caso configurado, de informações ao Heartbeat (PEREIRA, 2005).

O domínio da área de análise das falhas auxiliam os administradores a avaliarem a relação custo-benefício para o seu caso específico, determinando qual a melhor técnica para seu orçamento. As medidas executadas variam, podendo ser, por exemplo, uma simples necessidade de *backup* dos dados, uma redundância de equipamentos (servidores completos ou de componentes de *hardware*), um espelhamento de discos, utilizando-se, por exemplo, a técnica RAID[34]; ou uma necessidade de técnicas avançadas de redundância, como uso da redundância estática[35] (WEBER, 2002).

A NBR ISO/IEC 17999 sugere que podem ser confeccionadas cópias do plano de contigência que devem ser guardadas em ambiente seguro e remoto, a uma distância suficiente para escapar de qualquer dano de desastre no local principal. Recomenda-se, ainda, que outros materiais necessários para a execução do plano também sejam armazenados em local remoto.

Para validação do plano é necessário o investimento em treinamento dos envolvidos, cujos responsáveis, sejam pessoas aptas e preparadas para desempenhar satisfatoriamente as funções e executar a contento as atividades que lhes couberem (FRANCISCO, 2004). Corroborando com essa idéia, a norma NBR ISO/IEC 17799 estabelece diretrizes que visam por meio de testes programados, a avaliar o plano. Para isso, sugere os seguintes testes:

a) testes de mesa, simulando diferentes cenários, citando os procedimentos

[34] Dois ou mais discos , por exemplo, HD simultaneamente para um mesmo fim.
[35] Todos os elementos executam a mesma tarefa e o resultado é determinado por votação.

de recuperação para diferentes formas de interrupção;

b) simulações (particularmente útil para o treinamento do pessoal nas suas atividades);

c) testes de recuperação técnica, garantindo que os sistemas de informação possam ser efetivamente recuperados;

d) testes de recuperação em um local alternativo, executando os processos de negócio em paralelo com a recuperação das operações;

e) testes dos recursos, serviços e instalações de fornecedores garantindo que os serviços e produtos fornecidos atendam aos requisitos contratados; e

f) ensaio geral, testando se a organização, o pessoal, os equipamentos, os recursos e os processos podem enfrentar interrupções.

Tais procedimentos citados acima permitirão análises rotineiras do plano, e seus resultados contribuirão para que o plano seja, constantemente, revisado e atualizado, como parte do processo cotidiano da organização.

Portanto, identificou-se que com a aplicação de medidas, tais como, a elaboração de um plano de contingência e a realização de *backups* periódicos, podem contribuir para a continuidade do serviço de maneira imediata ou em um curto período de tempo.

5.6 CONTROLES DE MECANISMOS DE PROTEÇÃO AO SERVIDOR

Com os problemas inerentes à segurança, vindo paralelo ao crescimento da *internet*, surgiram ferramentas, dispositivos e práticas de controle de acesso para tornar as redes cada vez mais seguras, dentre estes estão os *firewalls* e os IDS (RIBEIRO, 2004). Se, analogamente, um *firewall* é a porta de um cofre, o IDS é o sensor de

movimento que monitora a sala do cofre. Apesar da porta proteger o interior do cofre, o sensor de movimento, ao detectar uma presença na sala do cofre, é que o dispara o alarme (KLER; PADRO, 2004).

Paralelo à questão, o uso de vermes e vírus podem prejudicar o desempenho de um servidor por estar realizando processos indesejáveis. Embora o Linux não seja extremamente suscetível a vírus, pode sê-lo aos *worms* que são desenvolvidos para explorar vulnerabilidades, conforme os dois casos que ficaram mundialmente conhecidos: o w*orm* Morris em 1988 e o w*orm* Ramen em 2001. Novas vulnerabilidades que podem ser acessadas pela rede são descobertas a cada dia, e poderiam ser usadas para propagar um verme (HATCH; LEE; KURTZ, 2002). O melhor modo de se proteger dos *worms* é se certificando de que o servidor, dentro do possível, esteja seguro.

Apesar de todos os obstáculos dificultando o acesso ao sistema, pode-se ainda, utilizar a criptografia para se manter a confidencialidade da informação.

Os controles de proteção ao servidor abordam mecanismos adicionais de segurança aos quais o servidor poderá utilizar visando dificultar o acesso indevido e a obter a informação. São eles: utilização de *firewall*, utilização de IDS, utilização de antivírus anti-*rootkits* e uso da criptografia.

5.6.1 Utilização de *Firewall*

No quesito “utilização de f*irewall*”, a norma NBR ISO/IEC 17799 diz, no item 11.4.5, sobre a separação entre a rede interna e externa, sugere que para tal perímetro pode ser implementado um *firewall* seguro a fim de controlar o acesso e o fluxo de informações entre os domínios.

Segundo Monteiro (2005), generalizando-se existem dois tipos de *firewall* no mercado: os filtros de pacotes e os *Gateway Firewall*.

Os que realizam filtros de pacotes, são amplamente utilizados por apresentarem baixo custo associado e por estarem normalmente integrados a dispositivos como roteadores, além de serem integráveis ou fazerem parte do próprio *kernel* do Linux (MONTEIRO, 2005).

O outro tipo denominado de *Gateway Firewall* ou *Applications Firewall*, pode apresentar certa dificuldade para ser implementado, mas é uma das melhores opções quando se trata de filtragem de conteúdo, pois permite cadastar regras mais complexas, exigindo que os usuários se autentiquem para estabelecer conexão, analisando-se informações pertinentes à aplicação (MARTINS, 2003).

A escolha está relacionada a fatores como custo, recursos desejados e flexibilidade (NBSO, 2003). Dentre as soluções disponíveis no Linux pode-se optar pelo Ipchains e o Iptables, sendo que este último é o mais usual e popular na comunidade Linux (RIBEIRO, 2004).

Existem outras ferramentas que podem fornecer uma camada extra de proteção para o sistema, controlando o acesso com base no endereço IP. Uma opção é o TCP-Wrappers, que permite filtrar pacotes direcionados a serviços oferecidos por vários *daemons*. Outra opção é o uso do Squid, mais atualmente utilizado no Linux, que é um *software* do tipo *Proxy* que atua como ponto entre duas redes, permitindo controle de tráfego de acordo com o seu conteúdo (DUMONT, 2006).

Outra medida a ser determinada pelo administrador é definir a localização do servidor *firewall*. Existem inúmeras formas de se instalar um *firewall*, geralmente, em conexões à *Internet*, localizado atrás do roteador, pois este permite que seja encaminhado para o *firewall* somente os pacotes destinados a rede interna. De acordo

NBSO (2003) algumas regras se aplicam a grande maioria dos casos:

a) passagem de todo o tráfego pelo *firewall*;

b) existência de filtro de pacotes no perímetro da rede;

c) implantação de servidores externos numa "zona desmilitarizada" (DMZ[36]); e

d) uso de *firewall* internos.

A política de segurança do *firewall* determina como ele irá filtrar o tráfego, ou seja, definirá qual tipo de tráfego será permitido. O primeiro passo é, então, definir o conjunto de regras, identificando que serviços irão trafegar por meio do *firewall* e sob quais circunstâncias deverão ser permitidos (MARTINS, 2003).

Martins (2003) ainda complementa que existem alguns tipos de tráfego que deverão sempre ser rejeitados pelo *firewall*, dentre eles destaca-se:

a) pacotes cujo destinatários é o próprio *firewall*, uma vez que este tipo de tráfego geralmente representa algum tipo de ataque ao próprio *firewall*;

b) pacotes provenientes da rede externa, mas com um endereço de origem da rede interna, que representam um ataque *spoofing*;

c) tráfego ICMP de entrada, já que este tipo de tráfego pode ser utilizado para mapear a rede interna;

d) tráfego de entrada do tipo SNMP de uma origem não autenticada;

e) tráfego de entrada e de saída cujo endereço IP de origem ou de destino seja 127.0.0.1 (local host);

f) tráfego de entrada destinado a um endereço de broadcast.

Por fim, recomenda-se que as regras deverão ser constantemente atualizadas e testadas para que possam, efetivamente, ser aplicadas.

[36] É uma pequena rede que tem por finalidade manter todos os serviços que possuem acesso externo (HTTP, FTP, etc) separados da rede local limitando o dano em caso de comprometimento de algum serviço nela presente por algum usuário externo mal intencionado.

5.6.2 Utilização de IDS

Com relação à "utilização de IDS", a norma NBR ISO/IEC 17799 diz, no item 10.4, com o objetivo de proteger a informação contra ameaças deve-se usar *softwares* de detecção de códigos maliciosos no controle de acesso adequado e nos controles de gerenciamento de mudanças.

Face à ameaça crítica no Linux, dos *rootkits*, uma categoria de *malware*[37] utilizada por usuários mal intencionados, a aplicação de mecanismos como IDS poderá ser uma forma de identificação da sua presença no sistema (CORREIA et al, 2005).

Existem dois tipos principais de IDS. O primeiro tipo são aqueles baseados em redes, o *Network Intrusion Detection System* (NIDS). Esse tipo é instalado em uma máquina, na qual a interface de rede monitora todo o tráfego de dados, Como exemplo pode-se citar o Snort e o NetSTAT. O segundo tipo são os *Host Intrusion Detection System* (HIDS), *são* baseados em host, que geralmente utilizam mecanismos de *log* do sistema operacional e estão ligados aos recursos do sistema, procurando por atividades incomuns, como: tentativas de *login*, acesso à arquivos, alterações de privilégios entre diversos outros tipos de ações (DRESCH, 2004).

Além de origens de dados de *host* e redes, os IDS também examinam dados de auditoria oriundos de registros de aplicações, como no caso de IDS específicos para servidores de aplicação, e dados oriundos de outros IDS na forma de alertas. Tais sistemas ainda podem ser caracterizados pelo seu comportamento quanto à intrusão após sua detecção, o tempo de detecção, a freqüência de uso e o tipo de arquitetura (SCHULTER, 2006).

[37] é um software destinado a se infiltrar em um sistema de computador alheio de forma ilícita, com o intuito de causar algum dano ou roubo de informações

Segundo Chothers citado por Steffen Junior (2003) são necessários três passos para auxiliar na avaliação e escolha de um IDS:

a) o estabelecimento da finalidade do IDS;

b) a identificação dos objetivos específicos para o IDS;

c) a determinar dos requisitos para o IDS.

No caso específico de servidores recomenda-se a instalação de um HIDS em cada um deles, pois tais ferramentas de auto monitoramento ajudam a manter o sistema seguro e consistente (TERSPSTA, 2005).

Dentre as inúmeras ferramentas disponíveis de HIDS cita-se o AIDE, Samhaim, o Osíris (CORREIA et al, 2006). RedHat (2005) acrescenta que o Tripware é um dos IDS mais conhecidos do LINUX. Silva (2006) sugere também o uso do Tiger. Abaixo estão descritas algumas características dessas ferramentas, fornecendo subsídios para uma melhor escolha.

O Advanced Intrusion Deteccion Environment (**AIDE**) cria uma base, com toda a informação do sistema atual, e a compara verificando se algo foi modificado.Com ele pode-se verificar se algum arquivo foi modificado ou deletado, se um binário não foi substituído, se as permissões foram alteradas, além de muitas outras informações. O **Samhaim**, tem sido uma opção cada vez mais utilizada para administradores Linux, pois possui funções semelhantes ao AIDE. O **Osíris** é outra opção que periodicamente monitora as alterações em um ou mais *host*. Sua finalidade é isolar as mudanças que indicam possível invasão. Pode ser configurado para enviar *email* dos *logs* ao administrador e suporta OpenSSL (CORREIA et al, 2006).

O **Tripware** permite ao administrador especificar facilmente os arquivos e diretórios a serem monitorados e a especificar arquivos para os quais alterações

limitadas são permitidas, sem gerar um aviso de alerta. Há duas versões do Tripware: uma versão comercial e outra versão de código-fonte aberto (TERSPTA, 2005).

O **Tiger** fornece, dentre as diversas opções verificações de casos comuns relacionados a furo de segurança, tais como: uso de força bruta das senhas, problemas no sistema de arquivo, comunicação de processos e outras formas de comprometer o superusuário (SILVA, 2006).

5.6.3 Utilização de Antivírus e Anti-Rootkits

No quesito “utilização de antivírus e anti-*rootkits*”, a norma NBR ISO/IEC 17799 determina, no item 10.4 determina que sejam implantados controles de detecção, prevenção e recuperação para a proteção contra códigos maliciosos.

O Linux tem definições claras sobre usuários, grupos, permissões e propriedade de arquivos conforme já foi citado na presente pesquisa. Tais propriedades de administração e de segurança permitem que se um vírus for executado por um usuário somente poderá afetá-los, ao contrário das máquinas que utilizam o Sistema Operacional Windows, em que o vírus pode possuir o controle total sobre a máquina (Hatch; Lee; e Kurtz, 2002).

Peña (2007) relata que já existem vírus para Linux, porém ainda não há relato que eles tenham se espalhado em alguma distribuição. Explica, ainda, que caso um administrador queira instalar sistemas antivírus que o proteja contra vírus enviados para outros sistemas mais vulneráveis pode-se usar por exemplo: o Clam Antivírus, Sanitizer, o Amavis. Na proteção aos *rootkits* sugere-se o uso do Chkrootkits (REDHAT, 2005).

Vianna (2004) acrescenta que é válido instalar estes sistemas em servidores *e-mail* (como o sendmail), pois, assim, evita-se a contaminação no servidor e se dificulta o envio de vírus aos usuários de destino.

5.6.4 Utilização de Controles Criptográficos

Com relação à “utilização de controles criptográficos”, a norma NBR ISO/IEC 17799 estabelece, no item 12.3, que seja desenvolvida uma política para uso de controles criptográficos tendo por objetivo proteger a confidencialidade, a autenticidade ou a integridade das informações.

Dependendo da função do servidor Linux poderá haver a necessidade de proteção dos dados além dos limites de medidas externas e permissões internas. Há muito a ser feito para proteger um sistema e, no fim, as proteções de usuário a um arquivo pode ser o último recurso de defesa contra brechas indesejáveis na privacidade. A criptografia poderá ajudar a proteger as informações contidas no arquivo (TERSPSTA, 2005).

Além de tornar as comunicações em rede mais seguras, a criptografia pode ser utilizada para proteger arquivos do sistema, de forma que, mesmo que um sistema seja invadido, os arquivos criptografados não são acessados pelo usuário mal intencionado (DUMONT, 2006).

Segundo a NBSO (2003), uma medida de segurança é a substituição de protocolos onde haja autenticação por meio de senhas, ou onde senhas trafeguem em claro (telnet, FTP, POP3, IMAP. Rlogin, Rsh e Rexec) por outros que corrijam as

deficiências. Nesse caso podem ser usados os protocolos seguros, como o Security Sockets Layers[38] (SSL) ou Security Shell[39] (SSH).

Uma solução de *software* criptográfico disponível em várias distribuições é o OpenSSL. Tal programa é freqüentemente utilizado como um componente de outros sistemas (Ex: Apache), permitindo realizar operações de criptografia bem simples em arquivos, usando-se ampla variedade de algoritmos criptografados. Para Criptografar um arquivo usa-se o comando **oppenssl**.

Outro método a ser usado na criptografia de arquivos é por meio do GnuPG (gpg) (versão livre da ferramenta pgp) que permite encriptar dados, assim somente o destinatário terá acesso aos dados. Adicionalmente poderá verificar se a origem dos dados é confiável (por meio da assinatura de arquivos). O sistema GPG baseia-se no conceito de chave pública, que é distribuída para as pessoas as quais se deseja trocar as informações.A chave privada deve ser conhecida apenas por quem a criou (SILVA, 2006).

Para se criar um par de chaves, segundo Terpstra (2005) usa-se o comando **gpg – key-gen** e segue-se as etapas abaixo:

a) inserir uma seleção do tipo de chaves (opção 1: DSA and ElGamal; opção 2: DSA; opção 3: ElGamal, opção 4: RSA);

b) inserir uma seleção de chave de sua escolha, tomando o cuidado de usar um número que não seja muito pequeno;

c) inserir uma data de validade; (podendo ser zero);

d) inserir as informações de ID do usuário (nome completo, email e um comentário);

[38] É um protocolo criptográfico que prove comunicação segura na Internet para serviços como email (SMTP), navegação por páginas (HTTP) e outros tipos de transferência de dados.

[39] É simultaneamente, um programa de computador e um protocolo de rede que permite a conexão com outro computador na rede, de forma a executar comandos de uma unidade remota. Possui as mesmas funcionalidades do TELNET, com a vantagem da conexão entre o cliente e o servidor ser criptografada.

e) inserir uma passphrase (uma frase que deverá ser de fácil recordação seguindo boas práticas de escolhas de senhas podendo conter espaços e não há limite de caracteres).

O comando **gpg** possibilita uma série de variedades de situações, tais como: criptografar e decriptografar arquivos, assinar e checar assinaturas, extrair e adicionar chaves públicas ao chaveiro pessoal, entre outros (SILVA, 2006).

Portanto, observou-se que ao conectar a rede interna de uma organização com a *internet*, existe a possibilidade de encontrar invasores anônimos que podem investigar e invadi-la por 24 horas por dia, sete dias por semana. Os controles de mecanismos de proteção visam estabelecer medidas adicionais de segurança. O *firewall* surge não como a única solução de segurança de rede, mas, sim como mais um complemento de segurança. Os IDS auxiliam na administração do sistema, possibilitando ter em mãos o controle daquilo que é alterado no servidor.

Por fim, a utilização de antivírus e *antirootkits* o protege contra as ameaças e o uso da técnica de criptografia oferece a confidencialidade das mensagens transmitidas pela rede ou mesmo o acesso a arquivos de caráter sigiloso.

5.7 CONTROLES DE MONITORAMENTO, AUDITORIA E TESTE

Se um usuário mal intencionado estiver tentando acessar o servidor, provavelmente fará várias tentativas antes de conseguir. Conforme Uchoa (2003 apud Dumont, 2005), em sistema medianamente seguro, uma invasão irá exigir esforço e tempo, de forma que, com o monitoramento eficiente, a invasão pode ser bloqueada em seu início.

Peña (2007) acrescenta que é notório o tratamento de *logs* e alertas como assunto de extrema importância. O monitoramento do sistema permite que sejam controlados os componentes de *hardware* e *software* do computador, identificando possíveis alterações ocorridas. Essa análise é realizada por meio dos arquivos de registros e por uso de *softwares* especiais de controle do sistema.

Paralelo tais questões as instituições estão cada vez mais conscientes e investindo em avaliação dos seus ambientes, buscando qualificar e verificar o nível de segurança que realmente possuem. Uma forma de aplicar esses procedimentos é realizar testes periódicos que validem os mecanismos implementados.

5.7.1 Política de monitoramento

No quesito "Política de Monitoramento", a norma NBR ISO/IEC 17799 apresenta, no item 10.10, com o objetivo de detectar atividades não autorizadas de processamento da informação, que os sistemas sejam monitorados e eventos de segurança da informação sejam registrados. Acrescenta, ainda, que os *logs* sejam utilizados para assegurar que os problemas dos sistemas sejam identificados.

Terspsta (2005, p. 280) relata uma experiência profissional ao trabalhar como Consultor em uma agência do governo Norte Americano:

> As investigações prosseguiram e demonstraram que tinham sido feitas modificações no binário **su**. Essas alterações eram tão sofisticadas que provavelmente não deveriam ter sido feitas pela equipe... Como não havia logs coletados e nem arquivados, foi impossível descobrir o que ocorreu e quando. Nos dias que se seguiram, o hacker continuou o seu trabalho... Novamente, se houvesse logs mostrando como foi feita a invasão, teria sido possível fazer alterações que impedissem novos ataques.

Correia et al (2006) ressalta que um administrador ao possuir controle sobre os *logs* que o Linux oferece, a administração do servidor fica mais fácil, pois os *logs*

ajudam a descobrir por que determinada aplicação não funciona, e alguns até apontam a linha e o que se errou no arquivo, além de descobrir se no sistema houve alguma acesso indevido, registros de tentativas de intrusão e até mesmo problemas de *hardware*.

A NBR ISO/IEC 17799 sugere que os registros contenham quando relevante:

a) identificação do usuário;

b) datas e horários de entrada;

c) identidade do terminal, nome da máquina ou IP;

d) registro das tentativas de acesso aceitos e rejeitados;

e) registros das tentativas de acessos a outros recursos e dados aceitos e rejeitados;

f) alteração de arquivos; e

g) uso de privilégios, aplicativos e utilitários do sistema.

Terspsta (2005) acrescenta que na política de monitoramento deve constar:

a) requisitos, se possível, para *logging* centralizado e seu gerenciamento, pois facilita o uso e evita-se a perda do *log*;

b) realização periódica do *backup* de arquivos de *log*; e

c) proteção dos arquivos de *log*, necessitando possuir acesso restrito;

Segundo NBSO (2003) os *logs* armazenados *off-line* devem ser mantidos no mínimo por seis meses, enquanto os *logs* de conexões de usuários de provedores acesso, segundo o Comitê Gestor da *Internet* , por um período de pelo menos três anos.

Acrescenta, ainda, que o administrador deve executar os seguintes procedimentos: inspecionar, pelo menos uma vez por dia os arquivos de *log*, investigar as causas de qualquer registro que lhe pareça incorreto ou anômalo e identificar o

padrão de comportamento normal dos seus sistemas para poder encontrar anomalias com maior rapidez.

De acordo com Argolo (2005), no Linux, o programa responsável por registar as atividades do sistema é o **syslog** e a maioria dos arquivos de log fica dentro do diretório */var/log*. O arquivo de configuração */etc/syslog.conf* permite a especificação exata de onde deseja que cada tipo de mensagem do sistema fique localizado.

O syslog permite especificar que tipo de programa está enviando a mensagem, a prioridade especifica da importância da mensagem e o destino para o local onde a mensagem está sendo encaminhada, podendo ser um arquivo, um computador remoto, ou um outro destino (SILVA, 2006).

A Tabela 11. lista os nomes válidos dos recursos disponíveis no *syslogd*:

Tabela 11. Recursos disponíveis no *syslogd*

RECURSO	PROGRAMAS QUE A UTILIZAM
Kern	Mensagem do *Kernel*
User	Processos do usuário
Mail	Sendmail e outros *softwares* relacionados ao correio
Daemon	*Daemons* do sistema
Auth	Comandos relacionados à segurança e à autorização
Lpr	Sistema para realizar *spool* de impressora
News	Sistema de notícias Usenet
Uucp	Programa de cópia de Unix para Unix
Cron	O *daemon* cron
Mark	Registro de data/hora gerados a intervalos regulares
Local0 – local7	Uso local
Syslog	Mensagens internas do *syslog*
Authpriv	Mensagens privadas de autorização
ftp	O *daemon* de FTP, FTPD
*	Todos os recursos

Fonte: TERSPSTA, J. (2005)

A tabela 12. apresenta os principais arquivos de *log*, com uma breve descrição:

Tabela 12 - Principais Arquivos de Log

Arquivos de log	Descrição
Utmp	Registra os usuários que estão conectados naquele momento no sistema. É o arquivo acessado pelos comandos w, who, finger por exemplo
Wtmp	Registra as conexões (login) e desconexões (logout) do sistema. É acessado por meio do comando **lastlog**
Btmp	Registra as falhas de conexão. Pode ser acessado pelo comando **lastb**
messages/syslog	Registra eventos e informações do sistema e aplicativos
boot.log/dmesg	Registra as mensagens relativas ao processo de inicialização do sistema
Secure	Mensagens privadas de programas e autorizações de usuários não registrados nesse arquivo
Sulog	Registra o uso do comando su
arquivos históricos	Arquivos como .history, .bash_history, entre outros, registram o histórico dos comandos que foram executados por cada usuário. Esses arquivos podem ser encontrados no diretório pessoal do usuário.

Fonte: ARGOLO, F. (2005)

Outro ponto importante em política de monitoramento é a centralização dos *logs* por meio de um servidor central. Provavelmente o usuário mal intencionado busca apagar as suas evidências, criando-se um servidor separado para esta finalidade a qual se propõe. Tal empreitada será mais difícil, além de facilitar a administração de um único local (TERPSTRA, 2005).

Existem várias ferramentas que permitam monitorar *logs* em tempo real ou processar diversos formatos conhecidos de logs que são bastantes úteis ao administrador. Dentre elas encontra-se o SWATCH, que requer que o administrador especifique um conjunto de padrões a serem monitorados e as ações a serem tomadas quando um destes padrões é registrado nos *logs* (NBSO, 2003).

Outros programas úteis são o Logchek, que envia um email periodicamente ao administrador do sistema alertando-o sobre os eventos que ocorreram desde a última execução do programa; e o Logrotate, usado para fazer *backups* dos *logs* atuais do sistema e criando novos arquivos de *logs* que serão usados pelo sistema (Silva, 2006). Outra opção para ser usada em substituição ao syslog é o syslog-nd que possibilita recursos mais avançados do que o syslog (CORREIA et al, 2006).

Por fim, observou-se que o registro de eventos e sua análise são atividades de vital importância para se identificar o comportamento do servidor. Se for necessário realizar perícia forense computacional, ter estrutura organizada com réplicas e cópias em outro local pode ser de extrema utilidade.

5.7.2 Política de Auditoria e Testes

Com relação à "política de auditoria e testes", a norma NBR ISO/IEC 17799 relata, no item 15.2, que com o objetivo de garantir conformidade dos sistemas com as políticas e normas organizacionais de segurança convém que sejam realizadas auditorias , em intervalos regulares nos sistemas em conformidade com as normas de segurança da informação pertinentes e com os controles de segurança documentados. Complementa, ainda, no item 10.10 que o monitoramento do sistema seja utilizado para checar a eficácia dos controles adotados e para verificar a conformidade com o modelo de política de acesso.

Ainda não se conhece um sistema de segurança perfeito. Sempre existe a possibilidade de que usuários não autorizados possam tentar acessar o sistema, usuários legítimos tenham efetuado ações erradas ou até mesmo atos maliciosos possam ser praticados (KON, 2005). Martins (2003) acrescenta que a política de segurança é dinâmica e, como tal, muda conforme o tempo, à medida que a instituição amadurece e as tecnologias evoluem.

A Auditoria de Sistemas tem por objetivo, justamente, promover a adequação e as recomendações, para o aprimoramento dos controles internos nos sistemas de informação, bem como na utilização dos recursos humanos, materiais e tecnológicos envolvidos nos processamentos dos mesmos (LEMOS, 2001).Após a implantação da

política de segurança, a etapa de monitoramento deve ser uma constante na organização, sendo realizada por meio de auditorias periódicas e ações de melhorias. Esses procedimentos são a chave do sucesso de uma política consistente e atualizada.

Não há regra que defina a periodicidade de revisão da política de segurança, contudo, a sugestão, descrita por Martins (2003) de que as revisões ocorram em intervalos entre seis meses e um ano, podendo haver ocorrência de revisões, quando for necessária uma modificação interna.

A utilização dos serviços da auditoria de sistemas, segundo Lemos (2001) implica certificar-se que:

a) as informações estejam corretas;

b) a existência de um processamento adequado das operações;

c) as informações estejam protegidas contra fraudes;

d) a existência de proteção das instalações e equipamentos;

e) a existência de proteção contra situações de emergência.

A NBR ISO/IEC 17799, no item 15.2.2, complementa que a verificação da conformidade técnica também engloba, por exemplo, testes de invasão e avaliações de vulnerabilidades. A realização desses testes demanda conhecimentos específicos, podendo haver contratação de terceiros que possibilite sua realização.

Existem inúmeros tipos de testes de segurança que são usados com o objetivo de avaliar a segurança implantada na organização. Segundo Ramos (2005) os testes servem como recurso para detectar vulnerabilidades de segurança de sistemas ou de rede, analisando-se a política de segurança.

Geralmente os testes empregam diversas técnicas normalmente empregadas por usuários mal intencionados. Após a verificação, os levantamentos obtidos são documentados e apresentados na forma de um relatório para, posteriormente, buscar a solução para minimizar a vulnerabilidade (RAMOS, 2005).

A Tabela 13 apresenta uma proposta de segurança, baseada no Guideline Network SecurityTesting.

Tabela 13. Periodicidade dos testes

Tipo de Teste	Descrição	Freqüência	Programas
Network Scanning (Varredura)	Possibilita ao administrador realizar um inventário dos serviços de redes disponíveis e certificar-se de que só estão disponíveis os serviços necessários.	Mensalmente	nmap, xprobe, queso, knocker, isic, icmpush, strobe, nbtscan, fragrouter, strobe
Vulnerability Scanning (Análise de Vulnerabilidade)	Permite ao administrador buscar potenciais vulnerabilidades dos serviços providos pelos servidores.	Quinzenalmente	nessus, racces, whisker, nikto, bass e satan
Password Cracking (Bruteforce)	São testes de segurança que podem auxiliar a verificação de como políticas de autenticação estão sendo eficientes, avaliando desde arquivos de senha de usuários até serviços que tenham autenticação.	Quinzenalmente	Crack, Jhon the ripper e Slurpie
Integrity Checkers (Análise de Integridade)	Um mecanismo que permitem analisar a integridade do sistema.	Semanalmente	AIDE, Samhaim, Osíris, Tripware, Tiger
Log Review (Análise de Logs)	Ë um teste que deve ser executado para que, por meio dos registros dos eventos, os administradores possam ser reativos ou mesmo pró-ativos diante de potenciais incidentes de segurança.	Diariamente de forma automatizada	SWATCH, logchek, logrotate
Rootkit Detection (Detecção de vírus/trojans)	Ë um teste de segurança que deve ser executado periodicamente verificando a presença de softwares indesejáveis.	Semanalmente ou quando requerido	lcap, chkrootkit, kstat,

Fonte: CORREIA, L. et al (2005); SILVA, G. (2006)

Portanto, observou-se que, diante das ameaças atuais, a pró-atividade implementada pela análise dos registros, auditoria na política de segurança e realização de testes periódicos de segurança são exemplos de iniciativas que permitem verificar o nível de segurança no sistema. Dessa forma, o Apêndice D – Análise prática, apresenta alguns tipos de testes, tendo por objetivo correlacionar a política elaborada com algumas possíveis formas de ataque, contribuindo para que tais medidas sugeridas sejam analisadas na proteção de um servidor.

CONCLUSÃO

O presente estudo abordou que a existência de inúmeras ameaças à segurança da informação e as constantes vulnerabilidades existentes nos diversos meios tecnológicos (*hardware* e *software*) podem trazer uma série de danos a organização, devido o alto valor que uma informação perdida, roubada ou adulterada possa ter. Dentre as medidas que pode ser adotada para aumentar o nível de segurança da informação, está a implantação de uma política de segurança.

A presente pesquisa realizada sugere uma política de segurança para ambientes de servidores Linux, baseando-se em normas nacionais e internacionais, estabelecendo controles que elevam o nível de segurança desses sistemas. Com base no estudo desenvolvido foi possível estabelecer uma política de segurança que se inicia pela seleção dos recursos humanos buscando-se montar uma equipe de profissionais motivados, competentes e acima de tudo idôneos. No decorrer do estudo foi identificada, ainda, a importância notória de se implantar controles de segurança física cuja finalidade é dificultar o acesso de pessoas não autorizadas aos servidores em questão e conseqüentemente aos arquivos e dados.

Com base na referência bibliográfica estudada conclui-se que a possibilidade de um sistema ser plenamente seguro é um processo extremamente difícil e complicado de ser implantado, implicando na necessidade de se adotar etapas sucessivas de procedimentos visando obter um nível de segurança adequado.

Do exposto, a presente política sugere acrescentar tais controles que visam aumentar a proteção lógica dos dados, tais como: o estabelecimento de procedimentos realizados na instalação do servidor, implantação de um senha segura, utilização de criptografia, elaboração de uma política de *backup*, utilização dos meios de monitoramento do sistema, dentre outras medidas.

A presente pesquisa possibilitou adquirir subsídios necessários ao estudo da segurança da informação, o que possibilitará a aplicação dos conhecimentos obtidos em sua vida pessoal e profissional. Acredita-se ainda, ser o marco inicial de pesquisas realizadas nesta área tão importante da computação. Outro ponto de destaque foi o estudo do Sistema Operacional Linux mostrando ser um sistema extremamente flexível e estável.

Ressalta-se, fruto da pesquisa realizada, a notória importância de instituições utilizarem políticas de segurança, estabelecendo regras e procedimentos a serem executados pelos seus membros com vistas a adquirir um nível de segurança adequado.

Como contribuição científica e acadêmica, abordou características importantes relacionadas a área da segurança computacional, analisando-se as ameaças atuais e medidas pró-ativas a serem executadas por um administrador de um servidor Linux. Com isso, espera-se que esse trabalho seja útil como guia de pesquisa na elaboração de projetos que envolvam a segurança dos recursos computacionais de qualquer organização que utilize o Linux como plataforma de trabalho.

As dificuldades encontradas ao longo do trabalho estão relacionadas à capacidade de sintetizar os diversos aspectos de segurança que envolvem um servidor, fruto de que o presente trabalho foi norteado pela norma NBR ISO/IEC 17799 que apresenta um grande número de controles de segurança. Ela está sendo substituída pela norma NBR ISO/IEC 27002, redigida com o intuito de resumir a referida norma que ainda encontra-se em vigor.

Procurou-se por meio da análise prática correlacionar os itens propostos na metodologia, com vista a dar uma idéia de sua aplicação em situações possíveis de

ocorrer em um ambiente de rede, destacando a aplicação de medidas pró-ativas frente as ameaças atuais.

Como trabalho futuro, nessa mesma linha de pesquisa, sugere-se estender os controles de segurança sugeridos para outros sistemas operacionais, tais como: Windows, Unix, entre outros. Outra pesquisa a ser realizada pode analisar os diferentes métodos de autenticação disponíveis no Linux, pois a autenticação é um método valioso para impedir o acesso indesejado de usuários mal intencionados.

Por fim, conclui-se que apesar de não ser possível proteger totalmente os servidores, a adoção de uma política de segurança específica para ambientes que utilizam servidores Linux, elaborada no presente trabalho, possibilita minimizar os riscos a que um servidor está sujeito, trazendo, assim, benefícios para as instituições que porventura aplicarem as práticas estabelecidas na presente pesquisa.

REFERÊNCIAS

ABNT. **NBR ISO/IEC 17799:** Tecnologia da Informação - Técnicas de Segurança - Código de prática para a Gestão da Segurança da Informação. Rio de Janeiro, 2005.

_____. **NBR ISO/IEC 270001:** Tecnologia da Informação – Técnicas de segurança – Sistemas de Gestão de Segurança da Informação. Rio de Janeiro, 2006.

ATÍLIO, César Eduardo.2003. 80 f. **Padrão ACME! para análise forense de intrusões em sistemas computacionais**. Trabalho de Conclusão de Curso (Graduação) - Curso de Ciência da Computação, UNESP, São José do Rio Preto, São Paulo, 2003.

BANDEIRA, Ronaldo; SALGADO, Ivan Jorge Chueri; SILVA, Rivanildo Sanches da. 2004. 123 f. **Análise de Segurança física em conformidade com a Norma ABNT NBR ISO/IEC 17799**. Trabalho de Conclusão de Curso (Graduação) – Curso de Tecnologia em Segurança da Informação, Faculdades Integradas ICESP, Brasília, Distrito Federal, 2004.

BAREINBOIM, Elias. **Conceitos básicos sobre segurança**. Disponível em: www.olinux.com.br . Acesso em: 10 Dez. 2006.

BARROS, Euriam. **Entendendo os conceitos de backup, restore e recuperação de desastres**. Rio de Janeiro: Ciência Moderna, 2007.

BERNARDES, Mauro César. 1999. 119 f. **Avaliação do uso de agentes móveis em segurança computacional**. Trabalho de Conclusão de Curso (Mestrado) – Curso de Mestrado em Ciências, Universidade de São Paulo, São Carlos, São Paulo, 1999.

CACIATO, Luciano Eduardo.2004. 24 f. **Gerenciamento da Segurança de Informação em Redes de Computadores e a Aplicação da Norma ISO/IEC 17799:2001**. Trabalho de Conclusão de Curso (Especialização), Pontifícia Universidade Católica de Campinas, Campinas, São Paulo, 2004.

CAMARGO, et al. **Segurança em Servidores Web**. Disponível em: http://www.das.ufsc.br/~emerson/academico/artigos/mellomcsbseg06.pdf. Acesso em: 15 Jun. 2007.

CANDÉA, Sérgio Luiz da Cunha. 2002. 76 f. **Coletânea de recomendações básicas de segurança de sistemas, destinadas aos administradores de rede**. Trabalho de Conclusão de Curso (Especialização), Instituto Tecnológico da Aeronáutica, São José dos Campos, São Paulo, 2002.

CARLOS NETO, João. **Segurança em redes móveis *Ad Hoc***. Relatório de Pesquisa – Curso de Doutorado em Ciência da Computação, Universidade São Paulo, São Paulo, São Paulo, 2004.

CORDEIRO, Rômulo Facuri Miranda; MOREIRA, Marcelo Eduardo da Silva. **Desenvolvimento de procedimentos de segurança e implantação de *firewall* no Laboratório de Bioinformática da rede Genoma Centro-Oeste**. Trabalho de Conclusão de Curso (Graduação) – Curso de Ciências da Computação, Universidade de Brasília, Brasília, Distrito Federal, 2002.

CORREIA, et al. **BS7799: da tática à prática em servidores Linux**. Rio de Janeiro: Altas Books, 2006.

DALLABRIDA, Paulo Victor.2004. 69 f. **Usando a tecnologia terminal services para uma sala informatizada**. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistemas de Informação, Instituto Superior Tupy, Joinville, Santa Catarina, 2004.

DELLA VALLE, James; ULBRICHT, Henrique e César. **Universidade H4CK3R**. São Paulo: Digetari Books, 2003.

DIAS, Cláudia. **Segurança e auditoria da tecnologia da informação**. Rio de Janeiro: Axcel Books, 2000.

DRESCH, Beno Eduardo.2004. 82 f. **Interopebalidade e compartilhamento de informações sobre ataques entre sistemas de detecção de intrusão utilizando o snort: modelo de conversão para CIDF**. Trabalho de Conclusão de Curso – Curso de Sistemas de Informação, Instituto Superior Tupy, Joinville, Santa Catarina, 2004.

DUMONT, Carlos Eduardo Silva. **Segurança computacional: segurança em servidores Linux em camadas**. Monografia – Curso de Especialização em administração em rede Linux, Universidade Federal de Lavras, Lavras, Minas Gerais, 2006.

FERREIRA, Aurélio Buarque de Holanda. **Miniaurélio**: O dicionário da lingua portuguesa. Curitiba: Positivo, 2006.

FRANCISCO, Rosemary.2004. 103 f. **A importância de um plano de continuidade do negócio na organização**. Trabalho de Conclusão de Curso (Graduação) – Sistema de Informação com ênfase em redes, Instituto Superior Tupy, Joinville, Santa Catarina, 2004.

GRIFFITH, Arthur; NORTON, Peter. **Guia Completo do Linux**. São Paulo: Berkeley,

2000.

HATCH, Brian; LEE, James; KURTZ, George. ***Hackers linux* expostos**. São Paulo: Makron Books, 2002.

HIJAZI, Houssan Ali; MAZZORANA, Sidney Miguel; RAVANELLO, Anderson Luiz. 2004. 85 f. **Honeypots e Aspectos Legais**. Trabalho de Conclusão de Curs o (Especialização) – Curso de Especialização em Informática Aplicada, Pontifícia Universidade Católica do Paraná, Curitiba, Paraná, 2004.

HUNT, Craig. **Servidores de redes com Linux**. São Paulo: Market Books, 2000.

KLER, Evelyn Ruth; PRADO, Gelson.2004. 75f. **Sistema de Detecção de Invasão**. Trabalho de Conclusão de Curso (graduação) – Curso de Administração com ênfase em análise de sistemas, Faculdade Internacional de Curitiba, Curitiba, Paraná, 2004.

KON, Fábio; PINHEIRO JUNIOR, José de Ribamar Braga. **Segurança em Grades Computacionais.** São Paulo: USP, 2005.

KURTZ, George; McCLURE, Stuart; SCAMBRAY, Joel. ***Hackers* expostos**. São Paulo: Makron Books, 2000.

LEMOS, Aline Morais de. 2001. 76 f. **Política de Segurança da Informação**. Trabalho de Conclusão de Curso(Graduação) – Curso de Administração, Universidade Estácio de Sá, Rio de Janeiro, Rio de Janeiro, 2001.

LIMA, Welton D. de; SILVA, Miris A. da; SOUTO, Cristiane C. 2006. 223 f. **Estudo e aplicação da Norma NBR ISO/IEC 17799: 2005 em segurança da informação**. Trabalho de Conclusão de Curso – Curso de Sistemas de Informação, Centro Universitário Unieuro, Brasília, Distrito Federal, 2006.

LOPES, Euler Nascimento; PEREIRA, Roverson dos Santos; SILVA FILHO, Marcus Vinícius. **Hacker curso completo**: práticas de ataque & segurança. Rio de Janeiro: Book Express, 2002.

MARTINS, Alaíde Barbosa. **Uma abordagem metodológica baseada em normas e padrões de segurança**: estudo de caso Cetrel S/A. Faculdade de Ciência e Tecnologia, Salvador, 2002.

MARTINS, José Carlos Cordeiro. **Gestão de projetos de segurança da informação**. Rio de Janeiro: Brasport, 2003.

MELLO, Sandro. **Implementando projeto de redes com *Software* Livre**. São Paulo: 4Linux, 2005.

MITNICK, Kevin D. **A arte de invadir**. São Paulo: Pearson Prentice Hall, 2005.

MONTEIRO, Emiliano Soares. **Segurança em ambientes corporativos**. Florianópolis: VisualBooks, 2003.

MORINOTO, Carlos E. **Segurança - Usando o Nessus e o Ethereal.** Disponível em: http: www.guiadohardware.net/tutoriais/110/. Acesso em: 20 Jun. 2007

MOTA FILHO, João Eriberto. **Descobrindo** o **linux**: entenda o sistema operacional GNU/Linux. São Paulo: Novatec, 2006.

NASCIMENTO, Marcos Aparecido de Figueiredo. 2005. 71 f. **Estudo de caso sobre migração do SO Windows para o Linux, no 41º Batalhão de Infantaria**

Motorizado, em Jataí-Go, com o menor impacto para o usuário. Trabalho de Conclusão de Curso (Especialização) – Curso de Especialização em Administração em Rede Linux, Universidade Federal de Lavras, Lavras, Minas Gerais, 2005.

NEMETH, Evi. **Manual de administração do sistema Unix.** Porto Alegre: Bookman, 2002.

NBSO. **Prática de Segurança para Administradores de Redes Internet Versão 1.2**. Disponível em: http: http://www.cert.br/docs/. Acesso em: 20 Jun. 2007

PAZ, Carlos. **Detección de puertos.** Disponível em: http: www.linuxdata.com.ar. Acesso em: 15 Jun. 2007.

PEÑA, Javier Fernandez-Sanguino. **Securing Debian Manual**. Disponível em: http: www.debian.org/doc/manuals/securing-debian-howto/. Acesso em: 28 Set. 2007.

PEREIRA, Roberto B. de Oliveira. 2005. 75 f. **Alta Disponibilidade em Sistemas GNU/Linux**. Trabalho de Conclusão de Curso (Especialização) – Curso de Especialização em Redes Linux, Universidade Federal de Lavras, Lavras, Minas Gerais, 2005.

PINHEIRO. José M. Santos. **Conceitos de redundância e contigência**. Disponível em: http://www.projetoderedes.com.br/artigos/artigo_conceitos_de_redundancia.php. Acesso em: 16 Ago. 2007.

RAMOS, Débora Lopes.2005. 84 f. **Estudo sobre as principais ameaças e técnicas para obtenção de falhas de segurança em sistemas cooperativos**. Trabalho de Conclusão de Curso (Graduação) – Curso de Ciências da Computação, Universidade Federal de Pelotas, Pelotas, Rio Grande do Sul, 2005.

REDHAT. **Guia de Segurança: Red Hat Enterprise Linux 4**. Disponível em: http: http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/pt_br/security-guide/. Acesso em: 15 Ago. 2007.

RIBEIRO, Sildenir Alves. 2004. 70 f. **Firewall em Linux**. Trabalho de Conclusão de Curso (Especialização) – Curso de Especialização em Administração em Rede Linux, Universidade Federal de Lavras, Lavras, Minas Gerais, 2004.

SANTOS, Davi Trindade. 2004. 61 f. **Pesquisa de vulnerabilidade e desenvolvimento de um scanner**. Relatório de Estágio Curricular – Curso de Ciências da Computação, Universidade Estadual de Londrina, Londrina, Paraná, 2004.

SCHULTER, Alexandre. 2006. 89 f. **Integração de sistemas de detecção de intrusão para segurança de grades computacionais**. Trabalho de Conclusão de Curso (Especialização) – Curso de pós-graduação em Ciência da Computação, Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina, 2006.

SEGURANÇA MÁXIMA: o guia de um *hacker* para proteger seu *site* na *internet* e sua rede. Rio de Janeiro: Campus, 2000.

SEGURANÇA MÁXIMA PARA LINUX: o guia de um *hacker* para proteger seu servidor e sua estação de trabalho. Rio de Janeiro: Campus, 2000.

SILVA, Gleydson Mazioli da. **Guia Foca GNU/Linux**. Disponível em: http://www.guiafoca.org. Acesso em: 20 Abr. 2006.

SOUZA, Adalberto Diniz.2001. 95 f. **Administração de serviços com segurança em servidores *Linux***. Trabalho de Conclusão de Curso (Graduação) – Curso de Ciência da Computação, Universidade Federal de Lavras, Lavras, Minas Gerais, 2001.

SOUZA, Leonardo Henrique de Lima de. 2004. 39 f. **Segurança física de redes de computadores**. Trabalho de Conclusão de Curso (Especialização) - Curso de Especialização em informática, Universidade Estácio de Sá, Rio de Janeiro, Rio de Janeiro, 2004.

SPANCESKI, Francini Reitz.2004. 102 f. **Política de segurança da informação**: desenvolvimento de um modelo voltado para instituições de ensino. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistema de Informação, Instituto Superior Tupy, Joinville, Santa Catarina, 2004.

STARLIN, Gorki; NOVO, Rafael. **Segurança contra *hacker***. Rio de Janeiro: Book Express, 2000.

SYMANTEC. **Guia para o planejamento de Contigência**. Disponível em: https: www-secure.symantec.com/region/br/enterprisesecurity/content/framework/BR_573.html. Acesso em: 20 Jun. 2007

TERPSTRA, John. **Segurança para Linux**. Rio de Janeiro: Elsevier, 2005.

THALENBERG, Marcelo. **Uso Corparativo de *email* e ferramentas de produtividade**. Disponível em: http://www.mtcriativa.com.br/pesquisa/Uso%20do%20e-mail.pdf. Acesso em: 10 Dez. 2006

THOMAS, Leandro Anchieta.2005. 89 f. **Estudo e implementação de uma ferramenta *Honeypot* para análise de intrusão**. Trabalho de Conclusão de Curso (Graduação) – Curso de Ciência da Computação, Universidade de Rio Verde, Rio Verde, Goiás, 2005.

VIANNA, Willian da Silva. 2004. 145 f. **Proposta de implementação de segurança para redes locais com acesso a internet**. Trabalho de Conclusão de Curso (Especialização) – Curso de Administração em Rede Linux, Universidade Federal de Lavras, Lavras, Minas Gerais, 2004.

VIGLIAZZI, Douglas. **Soluções para segurança de redes Windows**. Florianópolis: Visual Books, 2002.

WANDERLEY, Danillo Lustosa. 2005. 48 f. **Política de Segurança**. Trabalho de Conclusão de Curso(Especialização) – Curso de Especialização em Administração em Rede Linux, Universidade Federal de Lavras, Lavras, Minas Gerais, 2005.

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

WEBER, Taisy Silva. **Tolerância a falhas: conceitos e exemplos**. Disponível em http://www.mtcriativa.com.br/pesquisa/Uso%20do%20e-mail.pdf. Acesso em: 27 Abr. 2007

_____. Um roteiro para exploração dos conceitos básicos de tolerância a falhas. Disponível em www.inf.lasalle.tche.br/~barreto/disciplinas/tf/mat/tf_cap2.pdf. Acesso em 27 Abr. 2007.

APÊNDICE A – COMANDOS E ARQUIVOS DE CONFIGURAÇÃO DO LINUX

COMANDO	DESCRIÇÃO
mount	Serve para montar um sistema de arquivo
	`mount [`*`dispositivo`*`] [`*`ponto de montagem`*`] [`*`opções`*`]`
	exemplo: `# mount -o remount,rw,nosuid /home` (não permite que binários da pasta /home tenham acesso no nível root) `# mount -o remount,rw,noexec /tmp` (não permite a execução de qualquer binário ou arquivo executável dentro da partição /tmp)

	Script (/root/noexec):
	`#!/bin/bash`
	`case$1 in`
	`start)`
	`mount –o remount,rw,noexec /var`
	`mount –o remount,rw,noexec /tmp`
	`mount`
	`echo “Particoes Sem permissao de execução”`
	`;;`
	`stop)`
	`mount –o remount,rw,exec /var`
	`mount –o remount,rw,exec /tmp`
	`mount`
	`echo “Particoes Com permissao de execução”`
	`;;`
	`*)echo use $0{start|stop}`
	`exit 0`
	`;;`
	`esac`
	`exit 1`
	exemplo:
	`# ./noexec start` (deixa as partições sem permissão de execução)
	`# ./noexec stop` (deixa as partições sem permissão de execução)

vi	Editor de texto que serve, como exemplo, para alterar os arquivos de configuração
	`vi [nome do arquivo]`
	exemplo: `# vi /etc/fstab` (edita o arquivo fstab)
dpkg	Programa responsável pelo gerenciamento de pacotes em sistemas Debian.
	`dpkg –[opçao] [pacote]`
	exemplo: `# dpkg –l` (lista os pacotes existentes) `# dpkg –p grub` (remove o pacote e mostra informações detalhadas sobre ele)
man	Página do manual do programa, traz uma descrição básica do comando/programa e detalhes sobre o funcionamento de opção.
	`# man [secao] [comando/arquivo]`
	exemplo: `# man –k sendmail` (abre a página do manual do sendmail)
apt–get	Gerenciadores de pacotes do Debian.
	`apt –get [opcao] [pacotes]`
	exemplo: `# apt –get remove –purge wget` (remove completamente o pacote incluindo o arquivo de configuração)

COMANDO	DESCRIÇÃO
chmod	Muda a permissão de acesso a um arquivo ou diretório
	`Chmod [opções] [permissões] [diretório/arquivo]`
	Exemplo: `# chmod –R 700 /etc/rc.d/init.d/*` (altera a permissão para apenas o root ler,executar e escrever o init.d) `# chmod –s –Rv/` (retirar todas as permissões de Suid bit dos binários) `# chmod +s /bin/su` (estabelece a permissão de Suid bit no comando su) `# chmod g+w /etc/passwd` (altera as permissões do arquivo ou diretório no grupo) `# chmod 700 /etc/passwd` (altera as permissões do arquivo ou diretório dano permissão ao dono e retirando os dos outros utilizando sistema octal)
update-rc.d **update-inetd**	É uma ferramenta que edita daemons
	`# update-inetd [opções] [serviço]` # update-rc.d [opções] [serviço] [opções]
	Exemplo: `# update-rc.d –f mysql remove` (remove o mysql do arquivo rc.d) `# update-inetd --disable finger` (desabilita o finger do daemons inetd)
killall	Permite finalizar processos através do nome.
	`killall [opções] [sinal] [processo]`
	Exemplo: `# killall –HUP inetd` (finaliza o processo inetd)
passwd	Muda a senha do usuário ou grupo
	`$ passwd [usuário/grupo] [opções]` `$ passwd -x[tempo da senha] -w[tempo de aviso de expiração]-i [tempo após expiração para ativar a conta][usuário]`
	Exemplo: `$ passwd –x 10 –w 3 -i 2 tsux` (a senha do usuário "tsux" expirará após 10 dias e será avisado com 3 dias de antecedência para trocar senha. Após o período o usuário tem 2 dias antes da conta ser desativada)
adduser	Adiciona um usuário ou grupo no sistema
	`adduser [opções] [usuário/grupo]`
	Exemplo: `$ adduser tsux` (em resposta, O Linux solicitará a senha e a confirmação) `$ passwd`
userdel	Apaga um usuário do sistema
	`userdel [-r] [usuário]`
	`$ userdel tsux` (apaga o usuário e também o diretório Home do usuário)
addgroup	Adiciona um novo grupo de usuários no sistema
	`addgroup [opções] [grupo]`
	Exemplo: `$ addgroup turno1` (em resposta, O Linux solicitará a senha e a confirmação) `$ passwd`
groupdel	Apaga um grupo do sistema
	`Groupdel [grupo]`
	Exemplo: `$ groupdel tsux` (apaga o grupo)

COMANDO	DESCRIÇÃO
su	Permite que um usuário torne-se root Su [usuário] Exemplo: $ su (Em resposta, O Linux solicitará a senha root ao usuário)
chgrp	Muda o grupo de um arquivo/diretório. Chgrp [*opções*] [grupo] [arquivo/diretório] Exemplo: # chgrp Wheel /bin/su (muda o grupo que pertence o arquivo /bin/su)
usermod	Usado para modificar as informações sobre um usuário usermod [*opções*] [usuário] Exemplo: $ usermod – G wheel tsux (adiciona o usuário "tsux" ao grupo whell)
find	Procura por arquivos/diretórios no disco find [*diretório*] [*opções/expressão*] Exemplo: # find / -perm -4000 > /home/lista.suid (lista detalhada dos binários com permissão de Suid bit)
ls	Lista os arquivos de um diretório. ls [*opções*] [*caminho/arquivo*] [*caminho1/arquivo1*] Exemplo: # ls -1 /bin/su (ler a permissão de Suid bit no comando su) # ls –l /etc/pam.d (lista os programas compatíveis)
tar	Agrupa os arquivos tar [opções] [arquivo-destino] [arquivo-origem] Exemplo: tar –cf backup.tar /etc/* (realiza a cópia do diretório /etc inteiro no arquivo backup.tar no diretório corrente)
gzip	Compacta os arquivos gzip [opções] [arquivos] gzip -9 backup.tar (compacta o arquivo backup.tar usando a compactação máxima)
openssl	Criptografa um arquivo # openssl enc -[nome cifra] -[opção cript/decript] -in[arquivo-origem] –out [arquivo destino criptografado] Exemplo: # openssl enc –bf -e –in example1.txt –out example1.bf (faz uma criptografia usando a cifra Blowfish (bf) especificando o arquivo example1.txt para entrada e o arquivo example1.bf para exibição de saída do texto criptografado – e depois insere-se a senha com a confirmação) Obs: Para decriptografar usa a opção –d em vez de –e alterando os arquivos de origem e de destino
gpg	Criptografa um arquivo # gpg –[opção cript/decript] [arquivo-origem] Exemplo: # gpg – e arquivo.txt (Será pedida a identificação de usuário, digite o nome que usou para criar a chave. O arquivo criado será encriptado usando a chave pública do usuário e terá a extensão .gpg adicionada (arquivo.txt.gpg))

ARQUIVO	DESCRIÇÃO

time. conf	Arquivo de configuração que restringe as conexões (**/etc/security/time.conf)** `[serviço]; [terminal]; [usuário]; [horário]` Exemplo: `login;tty*;root;!A1000-2359` (root não se loga em nenhum terminal, em nenhum dia ou horário); `login & rsh;*;tsux;!Fr1800-2300` (usuário tsux não conecte-se ao servidor nem remotamente nem localmente, as sextas-feiras, no período das 1800 às 2300);
limits.conf	Define limites de uso dos recursos do sistema para cada usuário ou grupos de usuários Primeiramente deve editar o /etc/pam.d/login e inserindo a seguinte linha: `session required pam_limits.so` (habilitar o módulo) Com o módulo habilitado, edita-se o arquivo limits.conf localizado no diretório /etc/security e inserindo a seguinte linha `usuário hard maxlogins 1` (usuário só utilize somente um terminal)

lilo.conf	Arquivo de configuração do gerenciador de partida `lilo`.
	Para Inserir uma senha deve-se adicionar uma linha password e outra restricted no arquivo lilo.conf
	`...`
	`read-only`
	`password = <nova-senha>` (adicione a linha colocando a senha)
	`restricted` (adicione a linha)

grub.conf	Arquivo de configuração do gerenciador de partida `lilo`. Para inserir uma senha como root digite: `$ sbin/grub-md5-crypt` (solicitará uma senha retornando um hash MD5 da senha) → abaixo da linha “time out” insira a linha password substituindo o “password hash” pelo valor retornado no hash MD5 no arquivo grub.conf `$ passwords -md5 <password hash>` (substitua password pelo valor retornado)

syslog.conf	Arquivo de configuração do syslog
	`[recurso].[prioridade]`
	`[destino]`
	Exemplo:
	`mail.info`
	`/var/adm/mail`

`

APÊNDICE B – TUTORIAL DE SEGURANÇA PARA SERVIDORES (CHECK LIST)

✔ 1 SEGURANÇA EM RECURSOS HUMANOS

1.1 SELEÇÃO DOS RECURSOS HUMANOS

1.1.1 verificar nível de conhecimento profissional/técnico e idoneidade do candidato.

1.2 TERMOS E CONDIÇÕES DE CONTRATAÇÃO

1.2.1 realizar um termo de contrato descrevendo as responsabilidades pela segurança da informação.

1.3 TREINAMENTO DE PESSOAL

1.3.1 manter a equipe capacitada e atualizada.

1.4 PROCESSO DISCIPLINAR

1.4.1 instaurar processo de investigação para apurar o ocorrido;

1.4.2 estabelecer das punições: comunicação de descumprimento, advertência, suspensão ou demissão.

1.5 MEDIDAS DE ENCERRAMENTO OU MUDANÇA DE CONTRATAÇÃO

1.5.1 retirar o direito de acesso;

1.5.2 devolver os dispositivos que pertencem à organização;

1.5.3 ajustar a segurança física, se for o caso.

✔ 2 SEGURANÇA FÍSICA E DO MEIO AMBIENTE

2.1 SEGURANÇA DE EQUIPAMENTOS

2.1.1 identificar os requisitos na instalação e proteção do servidor;

2.1.2 identificar medidas para a segurança do cabeamento.

2.2 CONTROLE DE ACESSO FÍSICO

2.2.1 restringir o acesso físico;

2.2.2 controlar a entrada e saída dos usuários e visitantes.

2.3 PROTEÇÃO CONTRA AMEAÇAS EXTERNAS E INTERNAS

2.3.1 prover segurança contra incêndios, enchentes e outros eventos da natureza;

2.3.2 prover meios de proteção contra descargas elétricas e meios para garantia da energia elétrica;

2.3.3 elaborar controles ambientais;

2.3.4 utilizar dispositivos anti-furtos.

✔ 3 CONTROLE NA INSTALAÇÃO E IMPLANTAÇÃO DO SERVIDOR

3.1 SEGURANÇA DO SISTEMA DE ARQUIVO

3.1.1 dividir do disco em várias partições;

3.1.2 configurar as partições (*nosuid* e *noexec*).

3.2 INSTALAÇÃO MÍNIMA

3.2.1 planejar a instalação do SO;

3.2.2 elaborar o *logbook;*

3.2.3 escolher a opção personalizada durante a instalação, quando for possível.

3.3 DESABILITAR SERVIÇOS DESNECESSÁRIOS

3.3.1 verificar os serviços instalados;

3.3.2 verificar os serviços realmente necessários;

3.3.3 verificar a dependência dos serviços;

3.3.4 desinstalar ou desabilitar os serviços.

3.4 INSTALAÇÃO DAS CORREÇÕES DE SEGURANÇA

3.4.1 acessar sites especializados e participar de listas de segurança;

3.4.2 atualizar o sistema periodicamente;

3.4.3 rever a configuração do sistema.

3.5 CONFIGURAÇÃO DOS SERVIÇOS UTILIZADOS

3.5.1 configurar o sistema, se possível, logo após a instalação;

3.5.2 registrar as configurações no *logbook;*

3.4.3 configurar o sistema de acordo com a sua finalidade e nível de segurança.

---

4 POLÍTICA DE CONTROLES DE ACESSO AO SERVIDOR

4.1 POLÍTICA DO USO DE SENHA

4.1.1 estabelecer uma senha forte e com período de validade;

4.1.2 estabelecer a senha da BIOS, senha dos gerenciadores de boot (LILO e GRUB) e senha do Linux;

4.1.3 verificar por meio de softwares se as senhas estão fracas.

4.2 POLÍTICA DE GERENCIAMENTO DE USUÁRIOS

4.2.1 criar usuários ou grupos de usuários do sistema;

4.2.2 verificar autorização e nível de acesso do usuário;

4.2.3 manter um registro formal das pessoas registradas para usar o sistema;

4.2.4 remover ou bloquear direito de acesso de usuários afastados.

4.3 POLÍTICA DE GERENCIAMENTO DE PRIVILÉGIOS E PERMISSÕES DE ACESSO

4.3.1 configurar a permissão de *Suid bit;*

4.3.2 usar o menor tempo possível o usuário *root;*

4.3.3 limitar o acesso *root* a usuários selecionados;

4.3.4 bloquear o *loggin* do *root* em determinados terminais;

4.3.5 limitar usuários que podem executar o comando *sudo*;

4.3.6 estabelecer permissões de arquivos ou diretórios no sistema de arquivo.

4.4 CONTROLE DE ACESSO AO SISTEMA OPERACIONAL

4.4.1 desabilitar o uso do CTRL + ALT + DEL;

4.4.2 limitar o uso de terminais;

4.4.3 bloquear o terminal quando permanecer inativo;

4.4.4 utilizar um programa de autenticação de usuário (PAM).

4.5 POLÍTICA DE ACESSO REMOTO

4.5.1 realizar um levantamento da real necessidade de se habilitar o serviço remoto;

4.5.2 desabilitar, se for o caso, os logins remotos aos finais de semana e durante a noite;

4.5.3 usar protocolo que implementa criptografia.

---

5 CONTROLES DA PRESERVAÇÃO DA DISPONIBILIDADE DA INFORMAÇÃO

5.1 POLÍTICA DE BACKUP

5.1.1 relacionar os arquivos (dados, configuração e *logs*) para *backup*;

5.1.2 estabelecer a periodicidade do backup;

5.1.3 verificar as condições e as mídias de armazenamento dos backups;

5.1.4 identificar as mídias dos *backups* realizados;

5.1.5 guardar as cópias de segurança em local remoto;

5.1.6 Criptografar as cópias quando for o caso;

5.2 PLANO DE CONTIGÊNCIA

5.2.1 identificar os riscos que o servidor e a instituição estão sujeitos;

5.2.2 elaborar um plano que contenha: a identificação da possível falha, a condição de ativação do plano, os procedimentos de recuperação e a designação do responsável pela execução da correção;

5.2.3 guardar em local seguro e remoto uma cópia do plano de contigência;

5.2.4 estabelecer testes visando validar o plano.

---

6 CONTROLES DE MECANISMOS DE PROTEÇÀO AO SERVIDOR

6.1 UTILIZAÇÃO DE FIREWALL

6.1.1 escolher o *firewall* relacionado a custo, recursos desejados e flexibilidade;

6.1.2 identificar a localização do *firewall*;

6.1.3 estabelecer o conjunto de regras a serem usadas no *firewall*;

6.1.4 atualizar e testar constantemente as regras.

6.2 UTILIZAÇÃO DE IDS

6.2.1 avaliar e escolhe o IDS baseado na finalidade, nos objetivos específicos e nos requisitos para o IDS;

6.2.2 estabelecer, preferencialmente, um HIDS.

6.3 UTILIZAÇÃO DE ANTIVÍRUS E ANTI-ROOTKITS

6.3.1 instalar sistemas antivírus que protejam contra vírus enviados por outros sistemas;

6.3.2 instalar sistemas anti-rootkits.

6.4 UTILIZAÇÃO DE CONTROLES CRIPTOGRÁFICOS

6.4.1 utilizar criptogafia quando necessitar confidencialidade dos dados;

6.4.2 utilizar protocolos que utilizem criptogafia.

---

7 CONTROLES DE MONITORAMENTO, AUDITORIA E TESTES

7.1 POLÍTICA DE MONITORAMENTO

7.1.1 analisar periodicamente os *logs* de acordo com sua finalidade;

7.1.2 utilizar, se possível, servidores centralizados de *logs*;

7.1.3 realizar *backup* dos arquivos de *logs*;

7.1.4 proteger os arquivos de *logs* armazenando-os por um período de tempo definido;

7.1.5 utilizar ferramentas auxiliares que permitam monitorar *logs*;

7.2 POLÍTICA DE AUDITORIA E TESTES

7.2.1 realizar auditorias periódicas na política de segurança;

7.2.2 utilizar ferramentas específicas para testes buscando avaliar a segurança e localizar vulnerabilidades.

APÊNDICE C – APLICAÇÃO DA NBR ISO/IEC 17799 EM SERVIDORES

Item da Norma	Item	Aplicação na Política de Segurança
6.1	Infra-estrutura da segurança da Informação	Recursos Humanos (5.1)
8.2.3	Processo Disciplinar	
9.1.2	Controles de entrada física	Controle de Ambiente Físico e do Meio Ambiente (5.2)
9.1.4	Proteção contra ameaças externas e do meio ambiente	
9.2	Segurança de Equipamentos	
9.2.3	Segurança do cabeamento	
10.4	Proteção contra códigos maliciosos e códigos móveis	Controles na Instalação e Implantação do Servidor (5.3)
11.5.4	Uso de utilitários de sistema	
12.4.1	Controle de Software operacional	
11.2	Gerenciamento de acesso do usuário	Política de Controle de Acesso ao Servidor (5.4)
11.2.1	Registro de usuário	
11.2.2	Gerenciamento de privilégios	
11.2.3	Gerenciamento de senha do usuário	
10.5	Cópias de Segurança	Controles de Preservação da disponibilidade da Informação (5.5)
10.5.1	Cópias de segurança das informações	
14	Gestão da continuidade do negócio	
14.1	Aspectos da gestão da continuidade do negócio, relativos à segurança da informação	
14.1.2	Continuidade de negócios e análise de riscos	

Item da Norma	Item	Aplicação na Política de Segurança
11.4.5	Segregação de redes	Controle de Mecanismos de proteção ao servidor (5.6)
10.4	Proteção contra códigos maliciosos e código móvel	
12.3	Controles criptográficos	
10.10	Monitoramento	Controle de Monitoramento, auditoria e teste (5.7)
15.2	Conformidade com normas e políticas de segurança da informação e conformidade técnica	
15.2.2	Verificação da conformidade técnica	

APÊNDICE D – ANÁLISE PRÁTICA

Tem por objetivo analisar a política de segurança em um ambiente de rede simulado. Com esta análise foi possível correlacionar o emprego da presente política de segurança frente à diferentes tipos de ataques, demonstrando sua aplicabilidade em variadas situações.

Foram realizadas quatro avaliações que buscaram complementar o presente estudo. O teste Nr 1 tem por finalidade comparar três servidores usando-se uma ferramenta de exploração de vulnerabilidade (nmap) . O teste Nr 2 buscou por meio de um acesso físico instalar um *rootkit (cb-r00tkit)* para posteriormente realizar um ataque. O teste Nr 3 apresenta um *sniffer* buscando colher dados durante a interceptação. Por fim, o teste Nr 4 demonstra um ataque com a finalidade de derrubar um serviço FTP, ressaltando-se o uso de ferramentas de monitoramento.

Com exceção do teste Nr 1 as análises compararam um servidor sem controles de segurança (Servidor) e com os controles de segurança sugeridos (Servidor_Teste) estudando a política sugerida em resposta frente ameaças possíveis. O objetivo do presente teste não foi testar se a ferramenta empregada para o ataque é satisfatória, mas sim os princípios realizados pelo ataque com suas variantes e a forma como os servidores se comportaram.

A tabela abaixo apresenta a estrutura de *hardware* e Sistema Operacional instalado nas máquinas.

Máquina	**Configuração**	**Finalidade**
Server1	Sempron 2600+ memória 512 MB HD 80 GB SO Server Ubuntu 7.04	Utilizado como servidor PROXY, DHCP, DNS em uma instituição (teste 1) – instalado em maio 2007

Máquina	Configuração	Finalidade
Server2	Atholn XP 1800 (+) memória 1 GB HD 60 GB SO Server Ubuntu 7.04	Utilizado como servidor web (intranet) e de correio em uma instituição (teste 1) – instalado em maio 2007
Atacante	Notebook Satélite M45-S2693 Intel Pentium M 1,73 Ghz memória 1024 MB HD 100Gb SO Desktop Ubuntu 7.04	Utilizado pra prover os ataques (teste 2, teste 3 e teste 4)
Servidor Servidor_Teste	Athlon XP 2400 (+) memória 512 MB HD 40 GB SO Server Ubuntu 7.04	Utilizado para simulador um servidor (teste 1 à teste 4) Fornecendo um serviço *web* e FTP Para o Teste Nr 1 o servidor está com a instalação *default*
Host1	Intel Celeron 2,26 Ghz memória 512 MB HD 40 GB SO Desktop Ubuntu 7.04	Utilizado como uma máquina da rede (teste2)

Obs: foi utilizado um Hub de 8 portas 3Com para montar o ambiente de rede.

1. TESTE NR 1 – USO DO SCANNER DE VULNERABILIDADE

AVALIAÇÃO DE SERVIÇOS INSTALADOS	
AMBIENTE	Composto pelo Server 1, Server 2 e Servidor (recém concluída a instalação default) e atacante.
OBJETIVO	Identificar nos servidores em questão os serviços instalados e as vulnerabilidades (portas abertas).
DESCRIÇÃO DA TÁTICA EMPREGADA	Usar o Nmap 4.20 instalado na máquina atacante para análise dos servidores.

RESULTADOS OBTIDOS:

MÁQUINA	RESULTADO	AVALIAÇÃO
Servidor	Starting Nmap 4.20 (http://insecure.org) at 20071111 20:51 BRST All 1697 scanned ports on 10.1.1.25 **are closed**	Observa-se que ao instalar um servidor no modo *default* todas as portas estão fechadas.
Server1	Starting Nmap 4.20 (http://insecure.org) at 20071110 15:51 BRST PORT STATE SERVICE **22/tcp open ssh** **53/tcp open domain** **80/tcp open http** **139/tcp open netbiosssn** **445/tcp open microsoftds** **3128/tcp open squidhttp**	O Server1 tem por função prover os serviços de Proxy, DHCP, DNS o que justifica as portas 3128 e 53 estarem abertas. As demais portas , perfazendo um total de **4** portas, deveriam estar fechadas pois apresentam uma oportunidade para um usuário mal intencionado explorar suas vulnerabilidades.
Server2	Starting Nmap 4.20 (http://insecure.org) at 20071110 15:41 BRST PORT STATE SERVICE **21/tcp open ftp** **22/tcp open ssh** **25/tcp open smtp** **80/tcp open http** **110/tcp open pop3** **111/tcp open rpcbind** **139/tcp open netbiosssn** **445/tcp open microsoftds** **609/tcp open npmptrap** **2049/tcp open nfs**	O Server2 tem por função prover os serviços de Correio e Web o que justifica as portas 25, 110 e 80 estarem abertas. As demais portas , perfazendo um total de **7** portas, deveriam estar fechadas pois apresentam uma oportunidade para um usuário mal intencionado explorar suas vulnerabilidades.

AVALIAÇÃO: Observa-se no presente teste que os servidores (Server 1 e Server 2) possuem suas funções bem definidas provendo serviços específicos, porém apesar de tal definição conclui-se que a aplicação do item 5.2.3 Desabilitar Serviços Desnecessários diminuiria a vulnerabilidade dos sistemas.

2. TESTE NR 2 – ATAQUE FÍSICO COM INSTALAÇÃO DE UM ROOTKIT

ATAQUE FÍSICO COM INSTALAÇÃO DE UM ROOTKIT		
AMBIENTE	Composto pelo Servidor (provendo serviço Web) e usuário mal intencionado (funcionário da empresa).	
OBJETIVO	Analisar a segurança do servidor contra um ataque que tem por objetivo capturar um arquivo balancete.txt da pasta /home/ web.	
DESCRIÇÃO DA TÁTICA EMPREGADA	**ETAPAS**	**PASSOS EFETUADOS PELO ATACANTE**
	1	Aquisição do status *root* (simulação de que o administrador se ausentou da sala como o terminal plugado com o status de *root)*
	2	Instalação do Jhon the Ripper (v 1.6)
	3	Captura da senha usando um dicionário composto de 213.560 palavras (incluso palavras oriundas da engenharia social). Senhas: Servidor (usuário root – botafogo/ usuário administrador – guerreiro) Servidor_TSux (usuário root - 2E@ilM7w / usuário amilton – t&Qf2U7l).
	4	Instalação de um rootkit. cb-r00tkit (permite acessar remotamente usando o serviço ssh e permite bloquear os acessos a máquina)
	5	Acesso ao servidor pelo rootkit
	6	Captura do arquivo balancete.txt

RESULTADOS COLHIDOS:

Servidor	**Servidor_TSux**
Etapa 1 – Aquisição de *status root*	
Acesso físico liberado usuário *root* conectado `root@servidor:`	Acesso físico com restrições Usuário amilton (administrador) conectado `amilton@servidor:` Se o acesso fosse depois de 3 minutos não seria possível pois terminal estaria encerrado (tempo inatividade)
Etapa 2 – Instalação do Jhon the Ripper	
`root@servidor: apt-get install jhon` (poderia usar também um disquete e descompactar o arquivo)	`amilton@servidor:` Unable to lock the administration directory, are you root? (sem o conhecimento da senha não é possível instalar o pacote)
Etapa 3 – Captura da senha	
#**arquivo passwd** root:x:0:0:root:/root:/bin/bash administrador:x:1001:1001:PedroSantiago,,32339977,34317798:/home **# arquivo shadow** (senha criptografada) root:1pISblmeb$APzuVSJX2SNorc6rioiKT.:13828:0:99999:7::: administrador:1WIAvTiIU$Xe06REL7H1Zt96iwAnHyQ0:13833:0:99999:7:::	Não foi possível instalar o John (poderia ser usada a força bruta remotamente)

Servidor	Servidor_TSux
Etapa 3 – Captura da senha (continuação)	
`root@servidor:~# john -wordfile:/usr/share/john/dicionario.txt /etc/shadow` Loaded 2 passwords with 2 different salts (FreeBSD MD5 [32/32]) botafogo (root) guerreiro (administrador) guesses: 2 time: 0:00:00:16 100% c/s: 5864 trying: guerreiro	Não foi possível instalar o john
Etapa 4 – Instalação de um rootkit	
`root@servidor: ./ Install`	Não foi possível instalar o *rootkit*
Etapa 5 – Acesso ao servidor	
`root@atacante:/home/couto# ssh root@192.168.1.102 -p 2006` The authenticity of host '192.168.1.102 (192.168.1.102)' can't be established. RSA1 key fingerprint is 44:2e:e5:48:98:6f:8c:ea:55:11:31:43:66:5a:30:f6. Are you sure you want to continue connecting (yes/no)? yes Warning: Permanently added '192.168.1.102' (RSA1) to the list of known hosts. root@192.168.1.102's password: Linux host1 2.6.20-15-generic #2 SMP Sun Apr 15 07:36:31 UTC 2007 i686 The programs included with the Ubuntu system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright. Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law. No mail.	Não foi possível o acesso ao servidor
Etapa 6 – Captura do arquivo balancete.txt	

`root@Servidor:/home/administrador# scp /home/administrador/web/balancete.txt host1@192.168.1.100:/tmp`
The authenticity of host '192.168.1.100 (192.168.1.100)' can't be established.
RSA key fingerprint is bc:f4:4e:49:30:bc:9d:46:47:df:3c:50:da:48:f2:98.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.100' (RSA) to the list of known hosts.
`host1@192.168.1.100's password:`
balancete.txt 100% 21KB 21.5KB/s 00:00
`root@servidor :~# exit` logout
Connection to 192.168.1.102 closed.
	Não foi possível o acesso ao arquivo

AVALIAÇÃO: Avulta de importância a aplicação, em primeiro lugar, da seleção dos recursos humanos (item 5.1), ficando evidenciado, no suposto teste acima, a necessidade de contratação de uma equipe de funcionários motivados e idôneos. Os controles aplicados na segurança do ambiente físico e do meio ambiente (item 5.2) também apresentam a sua destacada importância fruto de estabelecer restrições ao acesso físico da sala onde localiza-se o servidor.

Na aplicação dos controles de acesso no servidor (item 5.3) foram observados os seguintes aspectos: execução da política de senha (item 5.3.1) ao não ser identificada as senhas do sistemas, aplicação da política de gerenciamento de usuário (5.3.2) ao criar os diversos usuários do sistema, no gerenciamento de privilégios e permissões (5.3.3) ao administrar corretamente o nível de *root* e habilitar permissões de arquivos e diretórios conforme foi observado na etapa 3, no controle de acesso ao sistema operacional estabelecendo restrição de tempo no uso do terminal.

3. TESTE NR 3 – USO DO SNNIFERS NA COLETA DE DADOS

USO DO SNNIFERS NA COLETA DE DADOS		
AMBIENTE	Servidor, Host 1 e atacante.	
OBJETIVO	Analisar a segurança do servidor contra um ataque que tem por objetivo deletar um arquivo do banco de dados (banco.sql) da pasta /home/web.	
DESCRIÇÃO DA TÁTICA EMPREGADA	**ETAPAS**	**PASSOS EFETUADOS PELO ATACANTE**
	1	Instalação de um Snnifer na rede (ethereal)
	2	O administrador acessa remotamente, de dentro da própria rede, o servidor usando telnet
	3	Captura e análise dos pacotes
	4	Remotamente acessa a pasta /home
	5	Deleta o arquivo

RESULTADOS COLHIDOS:

Servidor	**Servidor_TSux**
Etapa 1 – Instalação de um sniffer	
`Da máquina atacante:` `root@atacante:apt –get install ethereal`	
Etapa 2 – Administrador acessa remotamente	

Da máquina host1
`root@host1: telnet 192.168.1.102`
`root@host1: telnet 192.168.1.102 agenda.txt`
	Da máquina host1
	`dellane@host1: ssh 192.168.1.102`

Etapa 3 – Captura e análise dos pacotes	
A Figura 1 e 2, mostra, respectivamente, o tráfego do Servidor e do Servidor_TSux com a máquina host1 no momento que é feita requisição da senha para conectar o host1 com o servidor.	

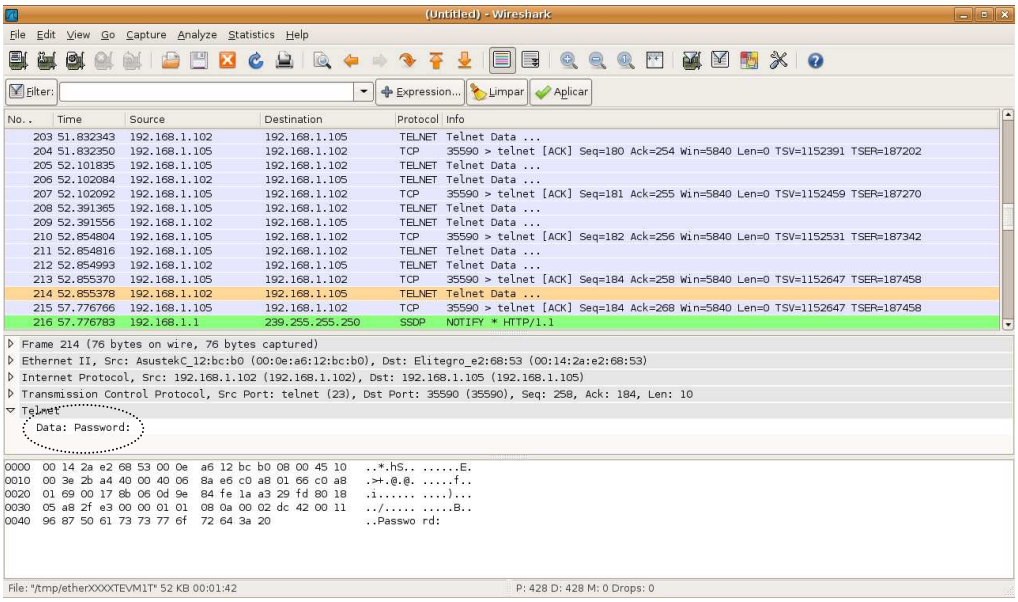

Figura 1 – Tela do Ethereal na captura dos dados usando-se o telnet

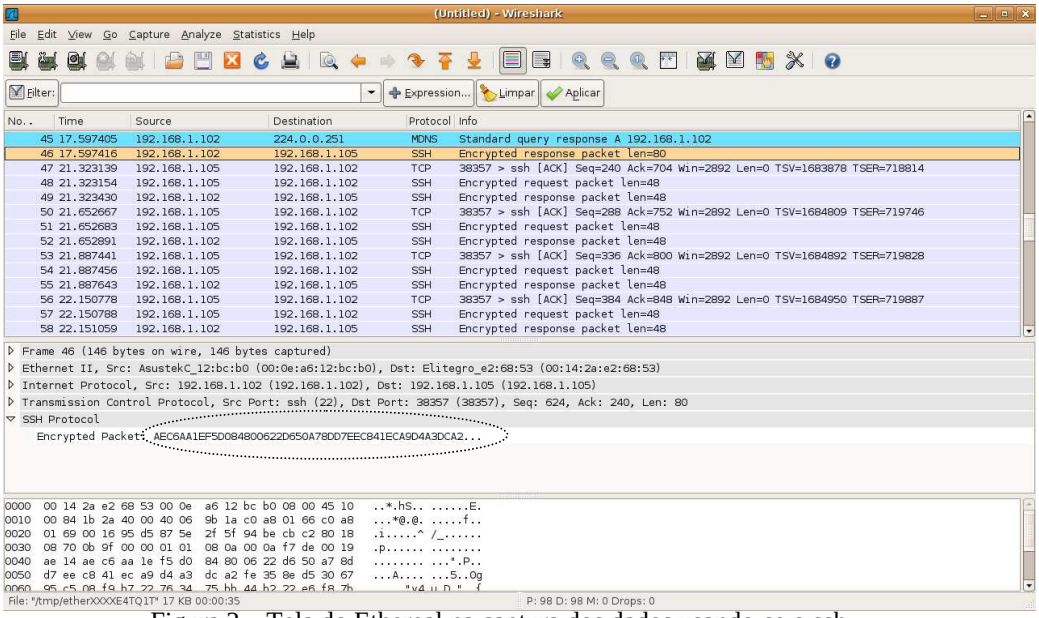

Figura 2 – Tela do Ethereal na captura dos dados usando-se o ssh

Servidor	Servidor_TSux
Etapa 3 – Captura e análise dos pacotes (continuação)	
Senha administrador: guerreiro Senha *root*: botafogo	Não foi possível identificar a senha devido o protocolo criptografar os dados
Etapa 4 – Remotamente acessa a pasta /home	

Da máquina atacante:
`root@atacante:telnet 192.168.1.102`
Trying 192.168.1.102...
Connected to 192.168.1.102.
Escape character is '^]'.
Ubuntu 7.04
`Servidor login: administrador`
`Password:`
Last login: Sat Nov 17 16:26:56 2007 from host1-desktop.local on pts/0
Linux Servidor 2.6.20-15-generic #2 SMP Sun Apr 15 07:36:31 UTC 2007 i686
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.
`administrador@Servidor:~$ su root`
`Password:`
	Não foi possível acessar o servidor

Etapa 5 – Deleta o arquivo	
`root@Servidor:/usr/bin# shred -u /home/administrador/web/banco.sql` `(o comando sobrescreve 25 vezes o arquivo)`	Não foi possível acessar o servidor

AVALIAÇÃO: Em primeiro lugar observou-se que deve ser utilizada a política de acesso remoto (item 5.4.5), visando estabelecer o acesso remoto somente em casos de real necessidade, no teste exposto o administrador foi consultar sua agenda. Além dessa

preocupação identifica-se haver necessidade do uso de protocolos seguros como está descrito no item 5.6.4 com o objetivo de impedir que dados trafeguem em claro pela rede.

Nesse caso suposto a finalidade do ataque era deletar o banco de dados, arquivo que registra inúmeros dados e que, em sua maioria, sua perda traz inúmeros e irreparáveis prejuízos. Com o comando (sherd) utilizado o arquivo torna-se irreparável sua recuperação. Do exposto, há a necessidade de se realizar periódicos *backups*, seguindo uma política de *backup* (item 5.5.1) possibilitando reduzir o impacto de uma possível perda de dados.

Outro resultado obtido está relacionado a permissão de acessos as pastas e arquivos caso o atacante não adquirisse a condição de *root* poderia em determinados arquivos não ter acesso a leitura do arquivo.

4. TESTE NR 4 – ATAQUE DE FORÇA BRUTA

ATAQUE DE FORÇA BRUTA		
AMBIENTE	Servidor e atacante.	
OBJETIVO	Analisar a segurança do servidor FTP contra um ataque que tem por objetivo realizar várias requisições usando um nome aleatório de usuário.	
DESCRIÇÃO DA TÁTICA EMPREGADA	**ETAPAS**	**PASSOS EFETUADOS PELO ATACANTE**
	1	Instalação do Nessus usando o pluggin FTP Service Allows Any Username.
	2	Realização das inúmeras requisições
	3	Acesso ao servidor
	4	Análise de log.

RESULTADOS OBTIDOS:

Servidor	**Servidor_TSux**
Etapa 1 – Instalação de brute force (neste caso foi utilizado o Nessus com um pluggin)	

`Da máquina atacante:`
`root@atacante:apt –get install nessus`
`(acrescentando bibliotecas adicionais)`	

Etapa 2 – Realização das requisições	

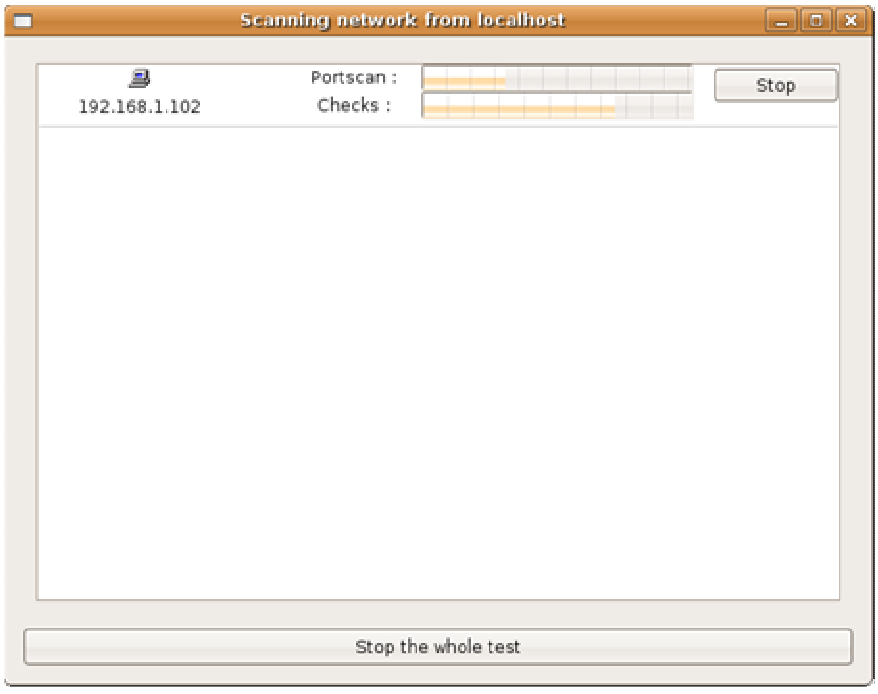

Figura 13 – Tela do Nessus escaneando o FTP

Servidor	Servidor_TSux
Etapa 2 – Realização das requisições (continuação)	
Efetuando as requisições conforme a Figura	Caso a porta fosse alterada na configuração o ataque poderia não causar efeito
Etapa 3 – Acesso ao servidor	
Sem acesso	Sem acesso
Etapa 4 – Análise de Log	
Não é realizada	Nov 18 20:18:46 servidor proftpd[7177]: IPv6 getaddrinfo 'servidor' error: No address associated with hostname Nov 18 20:18:46 servidor proftpd[7177] servidor (atacante.local[192.168.1.101]): FTP session opened. Nov 18 20:18:46 servidor proftpd[7177] servidor (atacante.local[192.168.1.101]): no such user 'gq5sz2j6' Nov 18 20:18:46 servidor proftpd[7177] servidor (atacante.local[192.168.1.101]): USER gq5sz2j6: no such user found from atacante.local [192.168.1.101] to 192.168.1.102:21 Nov 18 20:18:46 servidor proftpd[7177] servidor (atacante.local[192.168.1.101]): FTP session closed. Nov 18 20:22:53 servidor proftpd[7280]: IPv6 getaddrinfo 'servidor' error: No address associated with hostname Nov 18 20:22:53 servidor proftpd[7280] servidor (atacante.local[192.168.1.101]): FTP session opened. Nov 18 20:22:53 servidor proftpd[7280] servidor (atacante.local[192.168.1.101]): FTP session closed. Nov 18 20:23:40 servidor proftpd[7301]: IPv6 getaddrinfo 'servidor' error: No address associated with hostname Nov 18 20:23:40 servidor proftpd[7301] servidor (atacante.local[192.168.1.101]): FTP session opened. Nov 18 20:23:40 servidor proftpd[7301] servidor (atacante.local[192.168.1.101]): FTP session closed. Nov 18 20:23:43 servidor proftpd[7306]: IPv6 getaddrinfo 'servidor' error: No address associated with hostname Nov 18 20:23:44 servidor proftpd[7306] servidor (atacante.local[192.168.1.101]): FTP session opened. Nov 18 20:23:44 servidor proftpd[7306] servidor (atacante.local[192.168.1.101]): no such user 'ANXr1IQq' Nov 18 20:23:44 servidor proftpd[7306] servidor (atacante.local[192.168.1.101]): USER ANXr1IQq: no such user found from atacante.local [192.168.1.101] to 192.168.1.102:21 Nov 18 20:23:44 servidor proftpd[7306] servidor (atacante.local[192.168.1.101]): FTP session closed.

AVALIAÇÃO: Destaca-se na execução deste teste a necessidade de se configurar corretamente um serviço, conforme foi identificado no item 5.3.5, diversas alterações

realizadas nas configurações dos serviços podem impedir o sucesso de uma ataque, neste caso específico foi alterada a porta e alterado o número de processos simultâneos.

Outro ponto merece destaque é a utilização da política de monitoramento (item 5.7.1) que permitiu ao administrador identificar a suspeita do ataque realizado. Nesse caso específico poderiam também ser implementados controles adicionais de segurança como uso de *firewall* (item5.6.1) e de IDS (5.6.2). Por fim, o referido teste usou como ferramenta de ataque um analisador de vulnerabilidade (Nessus) permitindo identificar possíveis vulnerabilidades no sistema, tais analisadores são sugeridos na política de auditoria e testes (item 5.7.2) .

Um Modelo de Política de Segurança: a Aplicabilidade da Segurança da Informação em Servidores Linux

Moacyr A. Couto Jr[1], Rogério A. Casagrande[1]

Departamento de Ciência da Computação
Universidade do Extremo Sul Catarinense (UNESC) – Criciúma, SC – Brazil

mcouto.junior@gmail.com, roc@unesc.net

Abstratct*. The objective of this paper describe a project in development, is to provide an approach of methodology implementation of model security policies of linux server. This model will be based security norms and standards of national and international, searching to take care of to the security basic requirements of a company in the protection of information.*

Keywords: Information Security, Security Policies, Linux server.

Resumo. O presente artigo descreve um projeto em desenvolvimento, que consiste em fazer uma abordagem metodológica de implantação de um modelo de política de segurança para servidores Linux. Este modelo será baseado em normas e padrões de segurança de níveis nacionais e internacionais, buscando atender aos requisitos básicos de segurança de uma empresa na proteção da informação.

Palavras-Chaves: Segurança da Informação, Política de Segurança, Servidores Linux.

1. Introdução

A segurança das informações tornou-se um ponto crucial para a sobrevivência de qualquer instituição e deve ser tratada como um dos seus objetivos prioritários. Apesar desta realidade, em inúmeras empresas não há sequer uma política de segurança, que visa a manutenção e combate a possíveis ameaças, ou mesmo, se existe, a sua aplicação pode não ser executada por todos os agentes responsáveis.

Verifica-se, ainda, a existência de uma vasta literatura tratando do tema de Segurança da Informação. Apesar desta diversidade, não é fácil encontrar uma metodologia coerente e básica ou mesmo, um conjunto de diretrizes, que auxilie na elaboração de uma política de segurança específica para ambientes que utilizam servidores *Linux*.

Em função do exposto, procura-se, nesta pesquisa, analisar e documentar uma política de segurança, consistente e prática, específica para servidores *Linux*. As próximas seções deste artigo apresentam os fundamentos metodológicos seguido dos resultados preliminares.

2. Fundamentação teórica

2.1 Segurança da Informação

Na fundamentação teórica foram abordados assuntos como: segurança computacional, sistema operacional de rede, vulnerabilidades e políticas de segurança.

Segurança, em seu sentido mais amplo, é a capacidade de se proteger contra alguém ou algo. Segundo Ferreira (2006) segurança é o ato ou efeito de segurar, ou seja, é a condição livre de perigo ou risco.

A segurança computacional está relacionada à proteção de informações, sistemas e recursos computacionais contra erros ou manipulação não-autorizada, de forma a reduzir a probabilidade e os impactos de incidentes de segurança (DIAS, 2000).

Com as devidas circunstâncias, os *bugs*[40] de *software*, acidentes, erros, má sorte ou um usuário mal intencionado implicará que qualquer computador poderá ficar comprometido, submetido a desuso ou algo pior do que isso (CORDEIRO; MOREIRA, 2002).

O anseio dos usuários da informática está relacionado à que suas informações sejam confiáveis, corretas e mantidas em local seguro e que não haja acesso aos dados por pessoas não autorizadas. Tais expectativas, segundo Candéa (2002) se traduzem nos seguintes objetivos de segurança: confidencialidade, integridade de dados, disponibilidade e autenticidade.

Os principais ataques, aos sistemas de informação, descritas por Thomas (2005) são: vírus e *worms*, *trojans* e *backdoors*, *spywares*, ferramentas de exploração de vulnerabilidades e ferramentas de negação de serviços.

Nos sistemas computacionais as ameaças são constantes e a forma de se evitar ou de se dificultar os ataques é identificar e corrigir as vulnerabilidades existentes nos sistemas, algo que na verdade não é tão simples.

Nesse contexto, diversos mecanismos de segurança são realizados para a prevenção dos ataques. Spanceski (2004) enumera que desde procedimentos físicos, como impedir a entrada de usuários em uma determinada sala; treinamento e conscientização dos funcionários ou até realização de *backups*[41] periódicos ou políticas de controle de acesso, são ações implantadas para se chegar a tal finalidade.

Na maioria das vezes, deve-se usar a combinação de várias estratégias de acordo com o nível de segurança que se deseja alcançar. Algumas medidas que podem vir a serem adotadas são: estabelecimento de políticas de segurança, uso de criptografia, utilização de Sistema de Detecção de Intrusão (IDS), uso de *firewall*[42], e análise de *log*[43].

2.2 Sistemas Operacionais de Rede

Com o advento da *Internet*, surge a necessidade da utilização de Sistemas Operacionais que apresentam eficientes recursos de rede e segurança. Monteiro (2003) explica que SO é um conjunto de programas que gerenciam as funções internas do computador e permitem que o usuário controle a sua operação. Silva (2006, p. 5) complementa: “Ele é responsável pelo gerenciamento de recursos e periféricos (como memória, discos, arquivos, impressoras, CD-ROMs, entre outros), interpretação de mensagens e execução de programas”.

---

[40] Erro em um programa de computador que o faz executar incorretamente.
[41] Cópia de segurança de um arquivo ou sistema.
[42] São dispositivos que possuem a função de controlar o fluxo de tráfego entre uma rede interna e uma rede externa.
[43] São registros realizados por um sistema operacional ou programa das atividades executadas.

A Bell Laboratories em parceria com a AT&T, começou em 1969 um novo projeto de sistema operacional, denominado UNIX. Esse SO, segundo Monteiro (2003), surgiu como um ambiente de programação para atender técnicos, programadores, engenheiros e cientistas. Sendo um padrão aberto, o *UNIX*, conhecido também como *Single Unix Specification* permitiu que diferentes fabricantes desenvolvessem suas próprias implementações, dentre esses encontram-se o Solaris, Linux e o FreeBSD.

O Linux, motivado pelo estudo do sistema operacional Minix, surgiu em 1991, sendo desenvolvido por Linus Torvalds, na Universidade de Helsinki na Finlândia (SILVA, 2006), Segurança Máxima para Linux (2000) acrescenta que desde então o GNU/Linux cresce e vem tornando um sistema operacional completo sendo utilizado cada vez mais em diferentes ambientes.

Atualmente, segundo o site DistroWatch (http://distrowatch.com) existem mais de 350 distribuições GNU/Linux ativas. Essas distribuições variam conforme sua função, podendo ser usadas para usuários comuns ou em servidores.

O Linux trabalha com modularização, carregando somente para a memória o que é usado durante o processamento, liberando totalmente a memória assim que o programa ou dispositivo é finalizado. Devido a isso, os *drivers* dos periféricos e recursos do sistema podem ser carregados e removidos completamente da memória RAM a qualquer momento, o que torna o computador mais rápido (SILVA, 2006).

O Linux oferece uma ampla variedade de *softwares*, sendo que a maneira mais segura de adquirir os programas é utilizando os gerenciadores de pacotes, disponíveis em quase todas as distribuições. Os mais conhecidos são o Advanced Package Toll (APT), o Debian PacKaGe (DPKG), o RedHat Package Manager (RPM) e o YellowDog Updater Modified (YUM). Esses gerenciadores permitem realizar as operações básicas na manipulação de pacotes como instalação, remoção, consulta e checagem de arquivos.

Outra característica comum desse SO é a estrutura de diretórios. A perfeita noção do funcionamento da estrutura ajuda a corrigir problemas ou a implantar novas funcionalidades (MOTA FILHO, 2006).

É provável que todo dia uma nova vulnerabilidade seja identificada em algum problema ou sistema operacional. Estas vulnerabilidades permitem que um usuário mal intencionado execute uma operação que não deveria poder executar e geralmente adquire privilégios que não deveria conseguir. Dentre algumas vulnerabilidades exploradas no Linux, pode-se citar: a aquisição de senhas; estouro de *buffers*; utilização de serviços desnecessários, mal configurados e essencialmente inseguros.

2.3 Servidores Linux

Um servidor é um sistema de computador que fornece serviços a uma rede de computadores. Hunt (2000) destaca que muitos servidores possuem, geralmente, o propósito de fornecer uma grande quantidade de serviços, entretanto alguns, podem possuir uma única finalidade.

Dallabrida (2004) identifica os diversos tipos de serviços oferecidos pelos servidores: serviço de arquivos; serviço de impressão, serviço *web*, serviço de aplicações, serviço de correio eletrônico, serviço de Fax, serviço de comunicação e serviço de acesso remoto. Outros são encontrados, ainda, dependendo da necessidade.

2.4 Política de Segurança

Com a grande dependência da Tecnologia da Informação (TI), por parte das organizações, surgiu a necessidade de se considerar como um fator crítico para o sucesso a utilização de soluções viáveis para a proteção da informação. Com esse intuito de exprimir formalmente as regras que devem ser seguidas para se ter acesso aos recursos tecnológicos de uma organização é que se cria uma política de segurança (WANDERLEY, 2005).

A definição de uma política de segurança, segundo Dumont (2006), é o elemento mais importante da segurança da informação e deve envolver a segurança física, lógica e outros componentes relacionados à TI. Sem uma política de segurança bem elaborada não se sabe o que se vai proteger, nem porque e qual a melhor forma.

Neste contexto, observou-se que, com o intuito de exprimir formalmente as regras que devem ser seguidas para se ter acesso aos recursos tecnológicos de uma organização é que se cria uma Política de Segurança (WANDERLEY, 2005).

Spanceski (2004) acrescenta que uma política de segurança da informação é essencial, pois definem normas, procedimentos, ferramentas e responsabilidades para garantir o controle e a segurança da informação na empresa.

Ao implantar uma eficiente política de segurança é recomendável o estudo de normas e padrões de segurança (DIAS, 2000), as principais normas usadas como referência na segurança da informação são: a BS 7799, NBR ISO/IEC 17799, NBR ISO/IEC 27001.

A NBR ISO/IEC 17799, reeditada em 2005, serve como um guia prático para desenvolver eficientes práticas de gestão da segurança, procedimentos de segurança da informação da organização e para ajudar a criar confiança nas atividades interorganizacionais. A norma contém onze seções de controles de segurança da informação, que juntas totalizam trinta e nove categorias principais de segurança e uma seção introdutória que aborda a análise, avaliação e o tratamento de riscos.

3. Modelo Proposto

Por meio do estudo dos padrões e normas identificou-se que o modelo de política de segurança deve possuir os seguintes controles: segurança em recursos humanos; segurança física e do ambiente; controles na instalação e implantação do servidor; políticas de controles de acesso ao servidor; controle de preservação e disponibilidade da informação; controle de mecanismo de proteção ao servidor; controle de monitoramento, auditoria e teste. Os controles sugeridos são:

- **1 SEGURANÇA EM RECURSOS HUMANOS**
  - 1.1 SELEÇÃO DOS RECURSOS HUMANOS
    - 1.1.1 verificar nível de conhecimento profissional/técnico e idoneidade do candidato.
  - 1.2 TERMOS E CONDIÇÕES DE CONTRATAÇÃO
    - 1.2.1 realizar um termo de contrato descrevendo as responsabilidades pela segurança da informação.
  - 1.3 TREINAMENTO DE PESSOAL
    - 1.3.1 manter a equipe capacitada e atualizada.
  - 1.4 PROCESSO DISCIPLINAR

1.4.1 instaurar processo de investigação para apurar o ocorrido;

1.4.2 estabelecer das punições: comunicação de descumprimento, advertência, suspensão ou demissão.

1.5 MEDIDAS DE ENCERRAMENTO OU MUDANÇA DE CONTRATAÇÃO

1.5.1 retirar o direito de acesso;

1.5.2 devolver os dispositivos que pertencem à organização;

1.5.3 ajustar a segurança física, se for o caso.

✔ 2 SEGURANÇA FÍSICA E DO MEIO AMBIENTE

2.1 SEGURANÇA DE EQUIPAMENTOS

2.1.1 identificar os requisitos na instalação e proteção do servidor;

2.1.2 identificar medidas para a segurança do cabeamento.

2.2 CONTROLE DE ACESSO FÍSICO

2.2.1 restringir o acesso físico;

2.2.2 controlar a entrada e saída dos usuários e visitantes.

2.3 PROTEÇÃO CONTRA AMEAÇAS EXTERNAS E INTERNAS

2.3.1 prover segurança contra incêndios, enchentes e outros eventos da natureza;

2.3.2 prover meios de proteção contra descargas elétricas e meios para garantia da energia elétrica;

2.3.3 elaborar controles ambientais;

2.3.4 utilizar dispositivos anti-furtos.

✔ 3 CONTROLE NA INSTALAÇÃO E IMPLANTAÇÃO DO SERVIDOR

3.1 SEGURANÇA DO SISTEMA DE ARQUIVO

3.1.1 dividir do disco em várias partições;

3.1.2 configurar as partições (*nosuid* e *noexec*).

3.2 INSTALAÇÃO MÍNIMA

3.2.1 planejar a instalação do SO;

3.2.2 elaborar o *logbook;*

3.2.3 escolher a opção personalizada durante a instalação, quando for possível.

3.3 DESABILITAR SERVIÇOS DESNECESSÁRIOS

3.3.1 verificar os serviços instalados;

3.3.2 verificar os serviços realmente necessários;

3.3.3 verificar a dependência dos serviços;

3.3.4 desinstalar ou desabilitar os serviços.

3.4 INSTALAÇÃO DAS CORREÇÕES DE SEGURANÇA

3.4.1 acessar sites especializados e participar de listas de segurança;

3.4.2 atualizar o sistema periodicamente;

3.4.3 rever a configuração do sistema.

3.5 CONFIGURAÇÃO DOS SERVIÇOS UTILIZADOS

3.5.1 configurar o sistema, se possível, logo após a instalação;

3.5.2 registrar as configurações no *logbook;*

3.4.3 configurar o sistema de acordo com a sua finalidade e nível de segurança.

✔ 4 POLÍTICA DE CONTROLES DE ACESSO AO SERVIDOR

4.1 POLÍTICA DO USO DE SENHA

4.1.1 estabelecer uma senha forte e com período de validade;

4.1.2 estabelecer a senha da BIOS, senha dos gerenciadores de boot (LILO e GRUB) e senha do Linux;

4.1.3 verificar por meio de softwares se as senhas estão fracas.

4.2 POLÍTICA DE GERENCIAMENTO DE USUÁRIOS

4.2.1 criar usuários ou grupos de usuários do sistema;

4.2.2 verificar autorização e nível de acesso do usuário;

4.2.3 manter um registro formal das pessoas registradas para usar o sistema;

4.2.4 remover ou bloquear direito de acesso de usuários afastados.

4.3 POLÍTICA DE GERENCIAMENTO DE PRIVILÉGIOS E PERMISSÕES DE ACESSO

4.3.1 configurar a permissão de *Suid bit;*

4.3.2 usar o menor tempo possível o usuário *root;*

4.3.3 limitar o acesso *root* a usuários selecionados;

4.3.4 bloquear o *loggin* do *root* em determinados terminais;

4.3.5 limitar usuários que podem executar o comando *sudo*;

4.3.6 estabelecer permissões de arquivos ou diretórios no sistema de arquivo.

4.4 CONTROLE DE ACESSO AO SISTEMA OPERACIONAL

4.4.1 desabilitar o uso do CTRL + ALT + DEL;

4.4.2 limitar o uso de terminais;

4.4.3 bloquear o terminal quando permanecer inativo;

4.4.4 utilizar um programa de autenticação de usuário (PAM).

4.5 POLÍTICA DE ACESSO REMOTO

4.5.1 realizar um levantamento da real necessidade de se habilitar o serviço remoto;

4.5.2 desabilitar, se for o caso, os logins remotos aos finais de semana e durante a noite;

4.5.3 usar protocolo que implementa criptografia.

---

5 CONTROLES DA PRESERVAÇÃO DA DISPONIBILIDADE DA INFORMAÇÃO

5.1 POLÍTICA DE BACKUP

5.1.1 relacionar os arquivos (dados, configuração e *logs*) para *backup*;

5.1.2 estabelecer a periodicidade do backup;

5.1.3 verificar as condições e as mídias de armazenamento dos backups;

5.1.4 identificar as mídias dos *backups* realizados;

5.1.5 guardar as cópias de segurança em local remoto;

5.1.6 Criptografar as cópias quando for o caso;

5.2 PLANO DE CONTIGÊNCIA

5.2.1 identificar os riscos que o servidor e a instituição estão sujeitos;

5.2.2 elaborar um plano que contenha: a identificação da possível falha, a condição de ativação do plano, os procedimentos de recuperação e a designação do responsável pela execução da correção;

5.2.3 guardar em local seguro e remoto uma cópia do plano de contigência;

5.2.4 estabelecer testes visando validar o plano.

---

6 CONTROLES DE MECANISMOS DE PROTEÇÀO AO SERVIDOR

6.1 UTILIZAÇÃO DE FIREWALL

6.1.1 escolher o *firewall* relacionado a custo, recursos desejados e flexibilidade;

6.1.2 identificar a localização do *firewall*;

6.1.3 estabelecer o conjunto de regras a serem usadas no *firewall*;

6.1.4 atualizar e testar constantemente as regras.

6.2 UTILIZAÇÃO DE IDS

6.2.1 avaliar e escolhe o IDS baseado na finalidade, nos objetivos específicos e nos requisitos para o IDS;

6.2.2 estabelecer, preferencialmente, um HIDS.

6.3 UTILIZAÇÃO DE ANTIVÍRUS E ANTI-ROOTKITS

6.3.1 instalar sistemas antivírus que protejam contra vírus enviados por outros sistemas;

6.3.2 instalar sistemas anti-rootkits.

6.4 UTILIZAÇÃO DE CONTROLES CRIPTOGRÁFICOS

6.4.1 utilizar criptogafia quando necessitar confidencialidade dos dados;

6.4.2 utilizar protocolos que utilizem criptogafia.

---

7 CONTROLES DE MONITORAMENTO, AUDITORIA E TESTES

7.1 POLÍTICA DE MONITORAMENTO

7.1.1 analisar periodicamente os *logs* de acordo com sua finalidade;

7.1.2 utilizar, se possível, servidores centralizados de *logs*;

7.1.3 realizar *backup* dos arquivos de *logs*;

7.1.4 proteger os arquivos de *logs* armazenando-os por um período de tempo definido;

7.1.5 utilizar ferramentas auxiliares que permitam monitorar *logs*;

7.2 POLÍTICA DE AUDITORIA E TESTES

7.2.1 realizar auditorias periódicas na política de segurança;

7.2.2 utilizar ferramentas específicas para testes buscando avaliar a segurança e localizar vulnerabilidades.

4. Análise Prática

Foram realizadas quatro avaliações que buscaram complementar o presente estudo. O teste Nr 1 tem por finalidade comparar três servidores usando-se uma ferramenta de exploração de vulnerabilidade (nmap) . O teste Nr 2 buscou por meio de um acesso físico instalar um *rootkit (cb-r00tkit)* para posteriormente realizar um ataque. O teste Nr 3 apresenta um *sniffer* buscando colher dados durante a interceptação. Por fim, o teste Nr 4 demonstra um ataque com a finalidade de derrubar um serviço FTP, ressaltando-se o uso de ferramentas de monitoramento.

Com exceção do teste Nr 1 as análises compararam um servidor sem controles de segurança (Servidor) e com os controles de segurança sugeridos (Servidor_Teste) estudando a política sugerida em resposta frente ameaças possíveis. O objetivo do presente teste não foi testar se a ferramenta empregada para o ataque é satisfatória, mas sim os princípios realizados pelo ataque com suas variantes e a forma como os servidores se comportaram.

TESTE	DESCRIÇÃO	RESULTADOS OBTIDOS	
NR 1	Analisador vulnerabilidade (NMap)	Servidor recém instalado (Ubuntu 7.04) – portas fechadas Servidor 1 - Serviços de Proxy, DHCP, DNS (portas abertas: **22**/ 53/ **80**/ **139/445/**3128) Servidor 2 - Serviços Correio e Web (portas abertas: **21/ 22/** 25/ 80/ 110/**111/ 139/ 445/ 609/2049)**	
NR 2	Ataque físico / instalação de chrootkit	**Servidor (sem política implantada)**	**Servidor (com política implantada)**
		-Quebra da senha – 16 seg -Instalação do rootkit com êxito -Captura do arquivo	-Quebra da senha – sem êxito -Não foi possível instalar -Captura do arquivo não possível
NR 3	Captura senha (Snnifer) Ethereal	-Captura da senha (telnet) -Acesso remoto possível -Arquivo deletado (*shred* –u)	-Captura da senha –não possível (ssh) -Acesso remoto não possível
NR 4	Brute force (FTP Service Allows Any Username)	-Efetuando as requisições -Sem acesso ao serviço -Sem identificação do ataque	-Não localizou a porta do FTP que foi alterada na configuração do serviço -Identificação do ataque (análise do log)

Observa-se no **Teste Nr 1** que os servidores (Server 1 e Server 2) possuem suas funções bem definidas provendo serviços específicos, porém apesar de tal definição conclui-se que a aplicação do item 3.3 Desabilitar Serviços Desnecessários diminuiria a vulnerabilidade dos sistemas.

Com relação ao **Teste Nr 2** avulta de importância a aplicação, em primeiro lugar, da seleção dos recursos humanos (item 1.1), ficando evidenciado, no suposto teste acima, a necessidade de contratação de uma equipe de funcionários motivados e idôneos. Os controles aplicados na segurança do ambiente físico e do meio ambiente (item 2) também apresentam a sua destacada importância fruto de estabelecer restrições ao acesso físico da sala onde localiza-se o servidor.

Na aplicação dos controles de acesso no servidor (item 4) foram observados os seguintes aspectos: execução da política de senha (item 4.1) ao não ser identificada as senhas do sistemas, aplicação da política de gerenciamento de usuário (4.2) ao criar os diversos usuários do sistema, no gerenciamento de privilégios e permissões (4.3) ao administrar corretamente o nível de *root* e habilitar permissões de arquivos e diretórios conforme foi observado na etapa 3, no controle de acesso ao sistema operacional estabelecendo restrição de tempo no uso do terminal.

No **Teste Nr 3** em primeiro lugar observou-se que deve ser utilizada a política de acesso remoto (item 4.5), visando estabelecer o acesso remoto somente em casos de real necessidade, no teste exposto o administrador foi consultar sua agenda. Além dessa preocupação identifica-se haver necessidade do uso de protocolos seguros como está descrito no item 4.5.3 com o objetivo de impedir que dados trafeguem em claro pela rede.

Nesse caso suposto a finalidade do ataque era deletar o banco de dados, arquivo que registra inúmeros dados e que, em sua maioria, sua perda traz inúmeros e irreparáveis prejuízos. Do exposto, há a necessidade de se realizar periódicos *backups*, seguindo uma política de *backup* (item 5.1) possibilitando reduzir o impacto de uma possível perda de dados. Outro resultado obtido está relacionado a permissão de acessos as pastas e arquivos caso o atacante não adquirisse a condição de *root* poderia em determinados arquivos não ter acesso a leitura do arquivo.

Destaca-se na execução do **teste Nr 4** a necessidade de se configurar corretamente um serviço, conforme foi identificado no item 3.5, diversas alterações realizadas nas configurações dos serviços podem impedir o sucesso de uma ataque, neste caso específico foi alterada a porta e alterado o número de processos simultâneos.

Outro ponto merece destaque é a utilização da política de monitoramento (item 7.1) que permitiu ao administrador identificar a suspeita do ataque realizado. Nesse caso específico poderiam também ser implementados controles adicionais de segurança como uso de *firewall* (item 6.1) e de IDS (6.2). Por fim, o referido teste usou como ferramenta de ataque um analisador de vulnerabilidade (Nessus) permitindo identificar possíveis vulnerabilidades no sistema, tais analisadores são sugeridos na política de auditoria e testes (item 7.2) .

5. Conclusão

O presente estudo abordou que a existência de inúmeras ameaças à segurança da informação e as constantes vulnerabilidades existentes nos diversos meios tecnológicos (*hardware* e *software*) podem trazer uma série de danos a organização, devido o alto valor que uma informação perdida, roubada ou adulterada possa ter. Dentre as medidas que pode ser adotada para aumentar o nível de segurança da informação, está a implantação de uma política de segurança. Ressalta-se, fruto da pesquisa realizada, a notória importância de instituições utilizarem políticas de segurança, estabelecendo regras e procedimentos a serem executados pelos seus membros com vistas a adquirir um nível de segurança adequado.

Procurou-se por meio da análise prática correlacionar os itens propostos na metodologia, com vista a dar uma idéia de sua aplicação em situações possíveis de ocorrer em um ambiente de rede, destacando a aplicação de medidas pró-ativas frente as ameaças atuais.

Por fim, conclui-se que apesar de não ser possível proteger totalmente os servidores, a adoção de uma política de segurança específica para ambientes que utilizam servidores Linux, elaborada no presente trabalho, possibilita minimizar os riscos a que um servidor está sujeito, trazendo, assim, benefícios para as instituições que porventura aplicarem as práticas estabelecidas na presente pesquisa.

6. Referências Bibliográficas

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR ISO/IEC 17799:** Tecnologia da Informação - Técnicas de Segurança - Código de prática para a Gestão da Segurança da Informação. Rio de Janeiro, 2005.

CANDÉA, Sérgio Luiz da Cunha. 2002. 76 f. **Coletânea de recomendações básicas de segurança de sistemas, destinadas aos administradores de rede**. Trabalho de Conclusão de Curso (Especialização), Instituto Tecnológico da Aeronáutica, São José dos Campos, São Paulo, 2002.

CARLOS NETO, João. **Segurança em redes móveis *Ad Hoc***. Relatório de Pesquisa – Curso de Doutorado em Ciência da Computação, USP, 2004.

CORDEIRO, Rômulo Facuri Miranda; MOREIRA, Marcelo Eduardo da Silva. **Desenvolvimento de procedimentos de segurança e implantação de *firewall* no Laboratório de Bioinformática da rede Genoma Centro-Oeste**. Trabalho de Conclusão de Curso (Graduação) – Curso de Ciências da Computação, Universidade de Brasília, Brasília, Distrito Federal, 2002.

DALLABRIDA, Paulo Victor.2004. 69 f. **Usando a tecnologia terminal services para uma sala informatizada**. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistemas de Informação, Instituto Superior Tupy, Joinville, Santa Catarina, 2004.

DIAS, Cláudia. **Segurança e auditoria da tecnologia da informação**. Rio de Janeiro: Axcel Books, 2000.

DUMONT, Carlos Eduardo Silva. **Segurança computacional: segurança em servidores Linux em camadas**. Monografia – Curso de Especialização em administração em rede Linux, Universidade Federal de Lavras, MG, 2006.

FERREIRA, Aurélio Buarque de Holanda. **Miniaurélio**: O dicionário da lingua portuguesa. Curitiba: Positivo, 2006.

HUNT, Craig. **Servidores de redes com Linux**. São Paulo: Market Books, 2000.

MONTEIRO, Emiliano Soares. **Segurança em ambientes corporativos**. Florianópolis: VisualBooks, 2003.

MOTA FILHO, João Eriberto. **Descobrindo** o **linux**: entenda o sistema operacional GNU/Linux. São Paulo: Novatec, 2006.

SEGURANÇA MÁXIMA PARA LINUX: o guia de um *hacker* para proteger seu servidor e sua estação de trabalho. Rio de Janeiro: Campus, 2000.

SILVA, Gleydson Mazioli da. **Guia Foca GNU/Linux**. Disponível em: http://www.guiafoca.org. Acesso em: 20 Abr. 2006.

SPANCESKI, Francini Reitz. **Política de segurança da informação**: desenvolvimento de um modelo voltado para instituições de ensino. Trabalho de Conclusão de Curso – Curso de Sistema de Informação, Instituto Superior Tupy, Joinville, SC, 2004.

THOMAS, Leandro Anchieta.2005. 89 f. **Estudo e implementação de uma ferramenta *Honeypot* para análise de intrusão**. Trabalho de Conclusão de Curso (Graduação) – Curso de Ciência da Computação, Universidade de Rio Verde, Rio Verde, Goiás, 2005.