UNIVERSIDADE DO VALE DO ITAJAI
CENTRO DE EDUCAÇÃO SUPERIOR DE CIÊNCIAS TECNOLÓGICAS DA TERRA E DO MAR
CURSO DE CIÊNCIA DA COMPUTAÇÃO

Estratégias de Segurança

Área de Redes de Computadores

Fabricio Bortoluzzi

Itajaí (SC), novembro de 2001.

UNIVERSIDADE DO VALE DO ITAJAI
CENTRO DE EDUCAÇÃO SUPERIOR DE CIÊNCIAS TECNOLÓGICAS DA TERRA E DO MAR
CURSO DE CIÊNCIA DA COMPUTAÇÃO

Estratégias de Segurança

Área de Redes de Computadores

Fabricio Bortoluzzi

Relatório apresentado à Banca Examinadora do Trabalho de Conclusão do Curso de Ciência da Computação para análise e aprovação.

Itajaí (SC), novembro de 2001.

EQUIPE TÉCNICA

Acadêmico

Fabricio Bortoluzzi

Professor Orientador

Prof. Gilberto da Silva Luy, M.Eng.

Coordenador dos Trabalhos de Conclusão de Curso

Profª Anita Maria da Rocha Fernandes, Drª

Coordenador do Curso

Prof. Luís Carlos Martins

DEDICATÓRIA

Dedico este trabalho ao meu irmão Lucas e minha namorada Luciene, ambos acadêmicos da Ciência da Computação, bem como aos meus amigos administradores de redes, que criticaram construtivamente as idéias discutidas neste documento.

AGRADECIMENTOS

Aos meus pais, que, desde minha infância, nunca hesitaram em me fazer entender a importância de um curso de graduação.

Aos meus irmãos Chris e Lu, pela boa companhia que me proporcionam :-)

Aos professores, transmissores de conhecimento, em especial, ao meu orientador Gilberto Luy.

À minha namorada Luciene, que consegue transformar sentimentos de desespero em confiança.

Aos amigos que me incentivaram e ajudaram na coleta de informações para compor este, em especial: Ivan, Márcio, Mateus, Neto e Ronan.

Aos desenvolvedores de sistemas *open source*, por permitirem que possamos buscar no código-fonte o entendimento do funcionamento do *software* que executamos em nossos computadores.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

AIDE	Advanced Intrusion Detection Environment
BSD	Berkley System Distribution
CIFS	Common Internet File System
DNS	Domain Name Service
FTP	File Transfer Protocol
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
GID	Group ID, Group Identificator
HTML	Hyper-text Markup Language
HTTP	Hyper-text Transfer Protocol
IMAP	Internet Message Access Protocol
IDS	Intrusion Detection System
IP	Internet Protocol
IRC	Internet Relay Chat
LAN	Local Area Network
LIDS	Linux Intrusion Detection System
LDAP	Lightweight Directory Access Protocol
NAT	Network Adderss Translator

NFS	Network File System
NIDS	Novel Internet
NIS	Network Information System
TCP	Transmission Control Protocol
POP	Post Office Protocol, Point Of Presence
RIP	Routing Internet Protocol
SMB	Server Message Block
SNMP	Simple Network Management Protocol
SSH	Secure Shell
UDP	User Datagram Protocol
UID	User ID, User Identificator
WINS	Windows Internet Name Service
WWW	World Wide Web

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Os ataques contra sistemas conectados à Internet nos dias de hoje são mais sérios e complexos do que costumava ser no passado. Manter dados, recursos computacionais e, principalmente, a reputação de uma organização protegida se tornou tarefa para profissionais dedicados ao estudo de segurança da informação.

O objetivo deste trabalho é implementar e documentar técnicas que auxiliem nos procedimentos para garantir segurança de sistemas computacionais ligados em rede.

Para a realização do trabalho, levantou-se primeiramente as razões que as pessoas mal intencionadas têm para praticar atos de ataque e invasão dos ambientes computacionais alheios, bem como as ferramentas mais utilizadas por estas pessoas.

Para o entendimento das falhas de segurança envolvidas com o ambiente da tecnologia da informação, fez-se necessário um estudo sobre o comportamento dos elementos vitais de comunicação entre redes locais e Internet, o funcionamento dos sistemas operacionais e *softwares* que servem conteúdo nos *hosts* conectados, as várias topologias de *firewall*, seus benefícios e suas vulnerabilidades.

Também são apresentados os resultados de uma pesquisa onde foram analisados os procedimentos de segurança utilizados atualmente por 15 administradores de redes de universidades, provedores de acesso à Internet e empresas comerciais.

Em função do comportamento da pesquisa, são estudados métodos de segurança inéditos para o ambiente de trabalho dos entrevistados, mais aprimorados que os atuais, propondo a adição de uma nova camada de segurança nestas redes.

Conclui-se que o cumprimento das recomendações especificadas neste trabalho auxilia efetivamente na proteção das redes de trabalho expostas à Internet.

ABSTRACT

Attacks against Internet-connected systems are more serious and complex than they used to be in the past. Keeping data, computational resources and, above all, the reputation of institutions protected against such attacks has become a task to professionals dedicated to the study of information security.

The aim of this work is to document and to implement techniques in order to assist security procedures to assure the safety of computer networks.

The first part of this work tries to explain the reasons that would lead people to perform unauthorized actions with the intention of disturbing or harming the normal operation of third-part computer systems. This section also focus in determining the software and hardware tools used by these individuals.

A study about the behaviour of important elements in network systems and the Internet provided a way of understanding the security flaws related to such technology environment. Operational systems and firewall topologies benefits and vulnerabilities are important clues to solve security puzzles.

Finally, a research has been conducted to assess current network security conditions of local companies, including Internet service providers, academic networks and commercial companies. Based on the results of the research, this work proposes modifications in the current layout of their computer network to improve security.

I - INTRODUÇÃO

1. APRESENTAÇÃO

Embora a pilha TCP/IP (*Transmission Control Protocol* / *Internet Protocol*) viabilize a Internet e ofereça valiosos serviços, existem sérios problemas relacionados com a questão da segurança. Quando uma organização conecta sua rede com a Internet, qualquer computador na rede interna que utilize TCP/IP pode (teoricamente) ser acessado por qualquer usuário da Internet, o que representa um risco para a segurança das informações daquela rede.

Muitos dos problemas atuais de segurança na transmissão de dados pela Internet estão relacionados com o fato de que o protocolo TCP/IP não foi projetado para prover segurança, estando sujeito a várias formas de ataque. Como conseqüência, vários serviços da Internet que são baseados no TCP/IP, como o TELNET, FTP (*File Transfer Protocol*) e o SMTP (*Simple Message Transfer Protocol*) também são inseguros, pois utilizam mensagens que são transportadas em texto plano e, portanto, são vulneráveis a ataques que podem ser classificados em dois tipos: passivos, (como captura de tráfego através de *sniffers*) ou ativos, como o seqüestro de sessões.

2. JUSTIFICATIVA

O desenvolvimento deste trabalho é justificado pela necessidade de proteção das informações que se armazenam e se publicam nos servidores ligados à Internet. Manter a integridade de tais informações requer profissionais adequadamente especializados para esta finalidade, já que se trata de um esforço contínuo, para que a aplicação dos conceitos de segurança não fique desatualizada.

Em complemento, acrescenta-se a notável necessidade de estabelecer políticas de acesso e uso dos recursos computacionais concedidos aos usuários das redes de trabalho, tanto no meio acadêmico como no comercial, esclarecendo quais serviços podem ser acessados e quais não podem (e sob quais penas), bem como as rotinas do administrador de sistemas na distribuição e expiração de senhas, criptografia, geração de relatórios sobre tentativas de mau uso, sistemas de detecção de intrusos entre outros.

3. IMPORTÂNCIA DO TRABALHO

Este trabalho apresenta uma introdução ao conhecimento dos riscos envolvidos na conexão de redes de computadores à Internet bem como fornece abordagens teóricas e práticas sobre topologias de *firewalls*. O Comportamento ético do administrador de sistemas diante dos choques contra *hackers* também recebe tratamento, visando melhorar o cenário da segurança para aqueles que se interessam pelo assunto.

4. OBJETIVOS DO TRABALHO

4.1. Objetivo Geral

Estudar e implementar técnicas que auxiliem nos procedimentos para garantir a segurança dos sistemas computacionais e informações neles armazenadas e distribuídas, preservando, de forma íntegra, os recursos e a reputação dos seus proprietários, diante das deficiências existentes na Internet.

4.2. Objetivos Específicos

- Estudar a elaboração de um documento de política de segurança;
- Estudar a pilha TCP/IP implementada em um sistema operacional entendendo seu funcionamento e suas vulnerabilidades;
- Conhecer os ambientes de rede de outros administradores e com eles elaborar um modelo inovador de proteção para suas redes;
- Configurar um sistema de detecção de intrusos.

5. METODOLOGIA

- Coletar informações sobre os diversos sistemas operacionais existentes e que possam contribuir para o desenvolvimento do trabalho;
- Tomar conhecimento sobre quais serviços são explicitamente necessários e quais devem ser proibidos, com base nas redes utilizadas atualmente;

- Implementar no sistema operacional escolhido, as teorias estudadas;
- Estudar, no código-fonte do sistema operacional, o funcionamento do protocolo TCP/IP;
- Obter conhecimento das técnicas de detecção de intrusão, funcionamento e configuração;
- Executar testes.

5.1. Descrição Das Etapas

- Efetuar levantamento bibliográfico sobre os assuntos a serem estudados e entrega da Revisão Bibliográfica em 12 de novembro.
- Estudar os documentos citados na Revisão Bibliográfica a fim de obter informações de como efetuar os objetivos gerais e específicos citados anteriormente neste documento.
- Pôr em prática o conhecimento obtido nos estudos anteriores.
- Avaliação dos resultados, prestando informações sobre os pontos positivos e, eventualmente, negativos da implantação deste projeto de segurança.

6. CRONOGRAMA

6.1. Fase 1

Definição dos detalhes do projeto, de acordo com as instruções obtidas junto ao orientador.

6.2. Fase 2

Efetuar levantamento bibliográfico sobre os assuntos a serem estudados e entrega da Revisão Bibliográfica em 17 de novembro.

6.3. Fase 3

Estudar os documentos citados na Revisão Bibliográfica obtendo informações de como efetuar os objetivos gerais e específicos.

6.4. Fase 4

Pôr em prática os conhecimentos obtidos nas fases anteriores.

6.5. Fase 5

Avaliação dos resultados, descrevendo os pontos positivos e, eventualmente, negativos da implantação deste projeto.

A Tabela 1 apresenta uma descrição visual das fases do cronograma.

Tabela 1 Cronograma De Trabalho

	2000										2001													
F	Ago		Set		Out		Nov		Dez		Jun		Jul		Ago		Set		Out		Nov		Dez	
1	•	•	•	•																				
2				•	•	•	•	•																
3									•	•	•	•	•	•	•	•								
4													•	•	•	•	•	•	•	•	•			
5																	•	•	•	•	•	•	•	•

II - REVISÃO BIBLIOGRÁFICA

Com o crescimento exponencial da Internet, empresas do mundo inteiro vislumbram inúmeras oportunidades de negócios. Algumas empresas já utilizam a Internet como uma ferramenta para agilizar e criar os processos mercantis. Outras a utilizam como uma forma de oferecer novos serviços e de criar novas fontes de renda (BERNSTEIN et al. 1997).

Porém, segundo UNISINOS (1998), a utilização das redes de computadores como meio para realização de transações eletrônicas não é um fato recente na história da informática. Os bancos e outras instituições financeiras mundiais já utilizam, há muito tempo, serviços eletrônicos para efetuar negociações entre si.

Com a popularização deste uso, hoje, através da Internet, transações comerciais eletrônicas podem ser utilizadas pelo usuário para adquirir mercadorias em qualquer ponto do mundo (ibidem).

Atualmente, cada vez mais, empresas estão se apressando para usar a Internet por motivos comerciais e as questões relacionadas à segurança que estão presentes nessas conexões são consideradas extremamente relevantes, mas nem sempre tratadas de forma adequada. Já foi possível testemunhar centenas de ataques à sistemas conectados à Internet, sendo que muitos deles foram bem sucedidos nos seus objetivos. A não inclusão de controles de segurança adequados para conexões com a Internet pode fazer com que uma empresa se torne vulnerável a ataques que poderão deixá-la em uma situação embaraçosa e causar imensos prejuízos financeiros.

A adoção de medidas para a segurança de conexões com a Internet exige investimentos significativos em termos de tempo e esforço. Portanto, é necessário comparar custos e benefícios em relação a todas as medidas de controle de segurança.

1. RISCOS DA CONEXÃO COM A INTERNET

Para ZWICKY et al. (2000), quando se está conectado à Internet, coloca-se estes três ítens em risco:

- Dados
- Recursos
- Reputação

1.1. Dados

As propriedades que devem ser satisfeitas para que se tenha segurança dos dados são:

- Confidencialidade
- Integridade
- Disponibilidade

1.1.1. Confidencialidade

Existe uma tendência em se focar os riscos associados com a confidencialidade da informação, e comprovadamente, este é o maior dos riscos. Muitas organizações têm alguns dos seus maiores segredos - o projeto dos seus produtos ou registros financeiros - em seus computadores. Por outro lado, pode-se achar relativamente fácil separar fisicamente os computadores que têm informações confidenciais dos que devem estar conectados à Internet (ZWICKY et al. 2000).

Supondo que se possa separar os computadores desta forma e que nenhuma informação acessível seja sigilosa. Neste caso, por que se preocupar com a segurança? Porque confidencialidade não é a única coisa que se deve tentar proteger. Ainda há necessidade de proteção com a integridade dos equipamentos e a disponibilidade que eles provêem (ibidem).

1.1.2. Integridade

Mesmo que as informações não sejam particularmente sigilosas, pode-se sofrer conseqüências caso elas sejam destruídas ou alteradas. Geralmente, do ponto de vista do cliente, a perda das informações confidenciais implica na perda de confiança na empresa (ZWICKY et al. 2000).

1.1.3. Disponibilidade

A disponibilidade das informações contidas em um sistema diminui à medida em que se percebem riscos que possam comprometê-las. Freqüentemente ocorre a necessidade de acesso a documentos de uso interno por parte de um executivo em viagem. Em casos como este, muitas vezes é necessário reorganizar a política de segurança da empresa de forma que ela possa atender aos usuários "temporariamente externos", sem gerar oportunidades para intrusos (ZWICKY et al. 2000).

1.2. Recursos

Os intrusos, ou *hackers*, freqüentemente argumentam que utilizam somente recursos excedentes; como conseqüência, suas invasões não custam nada às vítimas. Há, para ZWICKY et al. (2000) dois problemas com este argumento:

- Primeiro: é impossível para um intruso determinar com certeza, que recursos em um sistema são excedentes. Pode-se pensar que o sistema tenha imensas quantidades de espaço de armazenamento vazio e horas de tempo de processamento não utilizadas. De fato, em muitas situações isto ocorre, porém, no momento em que todos os recursos são necessários, deve-se garantir que todos os recursos estejam íntegros.

- Segundo: É de direito da empresa utilizar os recursos como ela bem quiser, mesmo que isto signifique deixar disponível uma grande quantidade de recursos sem utilização.

1.3. Reputação

Quais seriam as conseqüências se um intruso acessasse as informações de um dos representantes de uma organização e as utilizasse na Internet? Para ZWICKY et al. (2000), algumas vezes, tais impostores podem tomar mais do que tempo. Um invasor que se empenha em destruir a identidade de uma organização pode, por exemplo, alterar o conteúdo do seu *web site* ou forjar mensagens eletrônicas. Outros exemplos piores podem danificar permanentemente a reputação da organização.

É possível forjar mensagens eletrônicas sem a invasão de um sistema, porém é muito mais fácil de comprovar sua falsidade (ibidem).

2. MOTIVOS QUE LEVAM OS *HACKERS* ÀS SUAS PRÁTICAS

De acordo com ANTIHACKERS (2000), há muito tempo se ouve falar de adolescentes que passam a noite inteira invadindo sistemas de computadores. Entretanto, muito pouco se fala dos mais perigosos *hackers*. O motivo pelo qual os jovens ganham destaque na mídia é a sua captura, pois eles não possuem conhecimento suficiente para que se mantenham ocultos por muito tempo.

Por inexperiência, deixam rastros por onde passam, pelo descuido e inconseqüência, ou porque simplesmente não têm motivos para se esconderem. Do outro lado, estão os *hackers* profissionais, extremamente cuidadosos em suas tentativas sendo muito mais difíceis de se detectar e capturar (ibidem).

Em geral, se trabalha com a seguinte classificação de atacantes potenciais: (BERNSTEIN, 2000)

- Curiosos: são estudantes que passam tempos na Internet procurando alguma forma de diversão, e normalmente aprendem com programas e ferramentas prontas que estão na *Web* e são facilmente capturados;

- *In-house*: são funcionários, ou ex-funcionários que procuram causar problemas para a empresa onde atuam ou atuaram. Muitas vezes são movidos por vingança, ou mesmo, por dinheiro, no caso de estarem sendo usados por um concorrente ou um terceiro que tenha interesse em prejudicar a empresa em questão. Estes são responsáveis por 80% dos ataques.

- Técnicos: normalmente são os que criam programas que causam danos, sendo extremamente bons no que fazem e espalham rapidamente suas novas técnicas através da Internet. Podem ser pessoas problemáticas com dificuldade de integração na comunidade e trabalham pelo prazer da destruição.

- Profissionais: esses são os mais perigosos, muitas vezes criam novas ferramentas ou até utilizam-se das que já existem. O diferencial é que recebem pelo que fazem. Trabalham para grupos mafiosos, terroristas ou espionagens industriais, normalmente são mais velhos (25 anos ou mais), muito inteligentes e difíceis de serem pegos.

ANTIHACEKRS (2000) observou que, independente do tipo de *hacker*, as motivações para seus ataques são bastante variadas. É possível categorizá-las nas seguintes definições:

- Espionagem Industrial: ocorre quando uma empresa contrata um *hacker* para que este invada o sistema da concorrência, e descubra seus planos, roube seus programas ou até mesmo suas políticas de parcerias e de investimento.

- Proveito Próprio: o *hacker* pode invadir um sistema para roubar dinheiro, transferir bens, cancelar dívidas ou até mesmo ganhar concursos, ou seja, qualquer ação em que ele seja diretamente beneficiado.

- Inexperiência: há também o caso de uma invasão ocorrer por ignorância. Por exemplo, um funcionário que acessa sua conta da empresa através do seu micro em casa. Dependendo da política de segurança da empresa, isto pode ser considerado uma invasão, mesmo que o usuário não tenha conhecimento do problema que pode causar.

- Vingança: Um ex-funcionário, tendo conhecimento do sistema, pode causar vários problemas, se o gerente de segurança da empresa não bloquear seu acesso imediatamente após sua saída da empresa. Ou, um parceiro de pesquisas pode acessar "mais do que deve" após a quebra de um contrato, trazendo complicações e prejuízos à empresa.

- *Status* ou Necessidade de Aceitação: uma invasão difícil pode fazer com que o invasor ganhe um certo *status* junto aos seus colegas. Isso pode acarretar uma competição, ou uma verdadeira "gincana" nas empresas. Dentro de grupos, é constante a necessidade de mostrar sua superioridade. Este é um fato natural, seja entre humanos, animais selvagens ou *hackers*.

- Curiosidade e Aprendizado: muitos *hackers* alegam invadir sistemas apenas para aprender como eles funcionam. Alguns fazem questão de testar o esquema de segurança, buscando brechas e aprendendo sobre novos mecanismos. Este tipo de ataque raramente causa um dano maior ou compromete os serviços atacados.

- Busca de Aventuras: o ataque a sistemas importantes, onde o esquema de segurança é muito avançado, pode fazer com que o *hacker* se sinta motivado pelo desafio e pelo perigo de ser pego, assim como alpinistas sobem montanhas, mesmo sabendo do risco de caírem.

- Maldade: algumas pessoas sentem prazer na destruição. Invadem e destroem, pelo puro prazer de causar o mal. Raramente são pegos e se vangloriam dos seus atos.

Seja o *hacker* quem for e faça ele o que fizer, é importante que ele seja neutralizado, pelo menos temporariamente, até que seu esquema de segurança seja revisto e atualizado. Essa atualização precisa ser constante, pois os *hackers* estão sempre em busca de falhas de segurança e, muitas vezes, não fazem nada além de invadir sistemas. É extremamente necessário que haja alguém dedicado a este assunto, pelo menos o mesmo tempo gasto por dia pelos *hackers* nas tentativas de invasão (ibidem).

3. ATAQUES

3.1. Técnicas De Ataques

Segundo ZWICKY (2000), os riscos que envolvem os procedimentos de ataques podem ser divididos em duas categorias. A primeira envolve conexões permitidas entre um cliente e um servidor, incluindo:

- Ataques de canal de comando.
- Ataques direcionados a dados.
- Ataques de terceiros.
- Falsa autenticação de clientes.

E a segunda, envolve ataques que trabalham a necessidade de se fazer conexões, incluindo:

- Seqüestro.
- *Packet Sniffing*.
- Injeção e modificação de dados.
- *Replay*.
- Negação de serviços.

UNISINOS (1998) complementa com as seguintes técnicas:

- *SYN Flood*.
- *Ping of Death*.
- IP *Spoofing*
- Ataques contra o protocolo NETBIOS.
- Ataques contra o X-Windows.

- Ataques utilizando o RIP (*Routing Information Protocol*).

3.1.1. Ataques de Canal de Comando

Esta técnica ataca diretamente um serviço em particular através do envio de comandos da mesma forma que o servidor geralmente os recebe. Ocorre de duas formas: Pelo envio de comandos válidos que acabam por provocar danos, e pelo envio de comandos inválidos que exploram *bugs* na manipulação de dados de entrada (ZWICKY et al. 2000).

É culpa do administrador de sistemas permitir a execução de comandos válidos que possam causar danos ao sistema. E, é culpa do programador que implementou o serviço, permitir que comandos inválidos possam provocar acesso não permitido ao sistema (ibidem).

Um exemplo desta técnica trata-se do *Worm* de Morris, conhecido como o primeiro ataque por parte de um *hacker*. Neste caso, Morris atacou o servidor de envio de mensagens Sendmail usando um comando válido de depuração que não foi devidamente bloqueado por vários *hosts* e que dava acesso completo ao sistema onde estavam hospedados (ibidem).

3.1.2. Ataques Direcionados a Dados

É o que envolve as informações transmitidas por um protocolo, ao invés daquela implementada por um serviço. Exemplos desta forma de ataque são os vírus transmitidos por correio eletrônico, ou ainda, o roubo das informações que trafegam na rede, como números de cartão de crédito (ZWICKY et al. 2000).

3.1.3. Ataques de Terceiros

Caso se permita conexão TCP entrante em qualquer porta acima de 1024, na tentativa de dar suporte a algum protocolo (como o FTP que necessita, alem da porta 21, uma porta para cada comando que o cliente faz) está se abrindo um grande número de oportunidades para que terceiros façam conexões de entrada (ZWICKY et al. 2000).

3.1.4. Falsa Autenticação de Clientes

Um grande risco para conexões entrantes é a falsa autenticação, que é a subversão da autenticação que se exige de quem deseja fazer uso do sistema. Por exemplo, criptografar uma senha e enviá-la pela rede por si só não funciona, pois, um *hacker* pode estar coletando as informações que trafegam com um *sniffer* e depois usa a senha (em sua forma criptografada mesmo) para obter acesso ao sistema. Tal ato é conhecido como *playback* (ZWICKY et al. 2000).

3.1.5. Seqüestro

Ataques de seqüestro permitem que um *hacker* tome conta de uma sessão de terminal em andamento, de um usuário que foi autenticado e autorizado pelo sistema. Ataques de seqüestro geralmente ocorrem em um computador remoto. Entretanto, às vezes, é possível seqüestrar uma conexão a partir de um computador na rota entre o cliente e o servidor (ZWICKY et al. 2000).

A única forma de proteção neste caso é permitir conexões à partir de computadores em que o servidor "confie". Isto pode ser conseguido através de configuração do filtro de pacotes ou por alterações de configuração no *software* que é executado no servidor (ibidem).

Os seqüestros de conexões que ocorrem no computador remoto, acontecem quando usuários deixam conexões em aberto e se ausentam do local. A possibilidade de este ataque ocorrer, depende da quantidade de usuários que podem se identificar no sistema e principalmente, da responsabilidade de cada um deles. Seqüestros que ocorrem em computadores intermediários são altamente técnicos e, geralmente, deve haver muitos motivos que levem um *hacker* a fazê-lo, devido a sua relativa complexidade. (ZWICKY et al. 2000)

O risco de seqüestro de conexões pode ser diminuído aplicando-se políticas sérias de controle de *time-outs* em caso de inatividade, como também por auditorias, na tentativa de se alertar ao administrador de sistema caso um seqüestro ocorra (ibidem).

3.1.6. Packet Sniffing

O invasor de sistemas não precisa necessariamente seqüestrar uma conexão para obter a informação que se pretende proteger. Pelo simples fato de observar os pacotes passando em qualquer direção entre o servidor e o cliente, o invasor pode capturar informações importantes (ZWICKY et al. 2000).

Proteger o *login* e a senha é uma tarefa fácil, através de senhas não reutilizáveis, ou seja, que são alteradas para cada conexão que se faz, inutilizando a senha que serviu para a conexão (ibidem).

Já a tarefa de se proteger os dados que trafegam em uma conexão é mais complicada. Estes dados precisarão ser criptografados antes de serem transmitidos de uma forma que somente o verdadeiro computador de destino possa decodifica-lo (ZWICKY et al. 2000).

3.1.7. Injeção e Modificação de Dados

Um invasor que controle um equipamento roteador entre o cliente e servidor, pode interceptar um pacote e alterá-lo, ao invés de apenas lê-lo. Proteger-se totalmente contra a injeção ou a modificação dos dados que trafegam por um roteador, requer alguma forma de proteção da mensagem na íntegra, adicionado-se valores de *checksum* que são computados na origem e não podem ser recalculados pelo atacante (ZWICKY et al. 2000).

3.1.8. Replay

Um invasor, que não possa tomar controle de uma conexão ainda pode ser capaz de danificá-la salvando uma cópia da informação que passa, e reenviando-a novamente. Há dois tipos de ataque por *replay*: um onde deve ser capaz de identificar certas partes da informação (por exemplo, senhas) e outro onde simplesmente se reenvia o pacote inteiro. A criptografia ajuda a evitar este tipo de ataque (ZWICKY et al. 2000).

3.1.9. Negação de Serviço

A Negação de serviço ocorre quando um atacante não está tentando ter acesso a nenhuma informação, apenas impede outras pessoas consigam acessá-la (ZWICKY et al. 2000).

Os ataques chamados de negação de serviço (ou *denial of service*) são aqueles que tem por objetivo deixar um recurso da rede indisponível (UNISINOS, 1998).

Do ponto de vista do atacante, é muito eficiente poder efetuar um ataque que não pode ser traçado (já que se forja o endereço de origem) e que requer um mínimo de esforço (explorar *bugs* no sistema alvo). Estes ataques, entretanto, tornam-se previsíveis, e os endereços forjados podem ser filtrados através do *firewall* (ZWICKY et al. 2000).

Outras formas deste ataque são praticamente impossíveis de se impedir. Um grupo de pessoas pode escolher o mesmo alvo e caracterizar um ataque distribuído, impedindo novamente que os usuários legítimos façam uso do recurso em questão ibidem).

Em geral, esta forma de ataque não causa danos as informações, nem ao *hardware* atacado, salvo se, em conseqüência de seu desligamento aconteça algum dano lógico (devido ao processo normal de desligamento que muitas vezes não poder ser efetuado, causando perda de alguns arquivos) (UNISINOS, 1998).

Mesmo sendo verdadeiro o fato de que os ataques de negação de serviços não podem ser totalmente prevenidos (ZWICKY et al. 2000), pode-se dificultar que um atacante (ou um grupo de atacantes) consiga realizar seu intento. Isto acontece em duas etapas:

- Os *hosts* não devem ficar indisponíveis quando comandos inválidos ocorrerem. Muitas vezes, *hosts* mal administrados deixam de responder ao usuário se um comando inválido repetir-se várias vezes.
- Os *hosts* devem limitar os recursos alocados para cada entidade que solicita conexão. Isto inclui:
    - O número de conexões abertas por servidor (*Web server*, SMTP *server*, etc...);
    - O tempo máximo de persistência de uma conexão;
    - A quantidade de processamento alocada para atender uma requisição de conexão;
    - A quantidade de memória alocada para atender uma requisição de conexão;
    - A quantidade de espaço em disco rígido alocada para atender uma requisição de conexão.

3.1.10. SYN Flood

Para UNISINOS, (1998) é um dos ataques mais populares de negação de serviço. Esse ataque visa impedir o funcionamento de um *host* ou um serviço em específico. O computador cliente (ver Figura 1) envia um pacote para o servidor com um *flag* especial, chamado de *flag* de SYN e este indica que o cliente deseja estabelecer uma conexão. Em seguida, o servidor responde um pacote contendo os *flags* SYN e ACK, indicando que ele aceitou o pedido de conexão e está aguardando uma confirmação do cliente para marcar a conexão como estabelecida. O ataque consiste em se enviar várias solicitações de conexão com o endereço de origem forjado. Como resultado, os usuários legítimos ficam impedidos de fazer uso dos serviços prestados pelo *host* alvo do ataque.

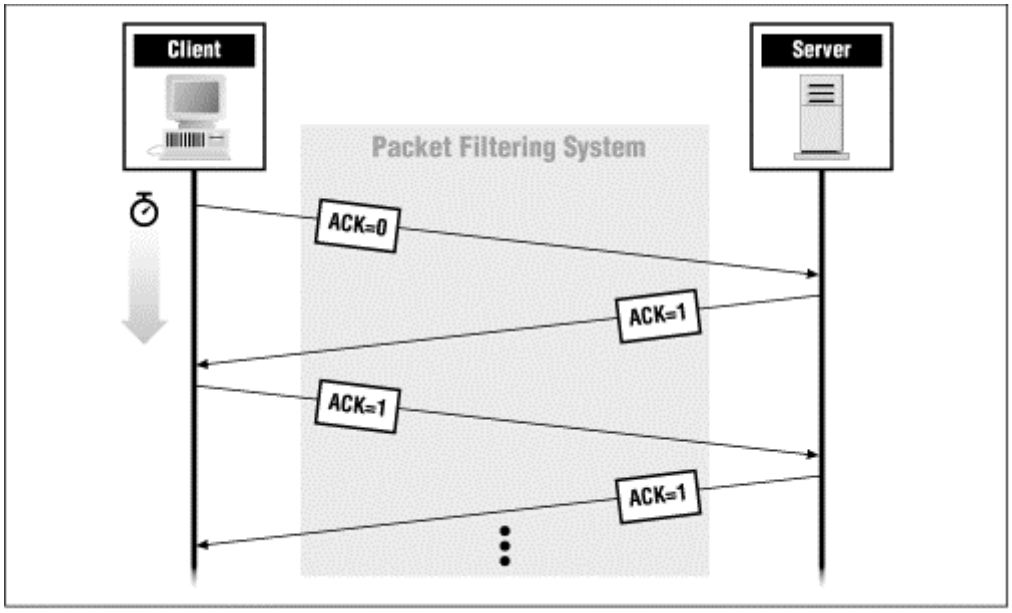

Figura 1 Negociação do bit ACK.
Fonte: (ZWICKY et al. 2000)

3.1.11. Ping of Death

Consiste em enviar um pacote IP com tamanho maior que o permitido (65535 bytes) para o *host* que se deseja atacar. O pacote então é enviado na forma de fragmentos ocasionando vários problemas no servidor, desde mensagens no console até o travamento sistema operacional. Alvos: todos sistemas operacionais que não implementaram correções para o problema de fragmentação de pacotes IP (UNISINOS, 1998).

3.1.12. IP Spoofing

É o nome dado para as falsificações de endereços IP, fazendo com que um pacote seja emitido com o endereço de origem diferente do computador que o enviou. Com o uso desta técnica é possível para um atacante assumir a identidade de qualquer outro computador ligado à Internet. O maior perigo deste ataque se dá nos serviços baseados no protocolo UDP (*User Datagram Protocol*). Através do *spoofing* de IP é possível para um atacante tirar proveito dos *hosts* confiáveis armazenados no arquivo .rhosts existentes nos sistemas operacionais Unix e derivados, possibilitando acessos via rlogin por exemplo, onde a senha não é exigida. Alvos: principalmente os sistemas operacionais derivados do BSD (*Berkley System Distribution*), por serem os que implementam servidos de rlogin e rsh (UNISINOS, 1998).

3.1.13. Ataques Contra o Protocolo NetBIOS

As plataformas Windows e OS/2 têm a característica de disponibilizar todos os seus serviços através do protocolo NETBIOS. Em redes onde o NETBIOS é disponibilizado sobre TCP/IP, é possível para um atacante verificar quais são os diretórios e as impressoras compartilhadas por cada computador presente nestas redes. Normalmente é comum a prática dos usuários de deixar o acesso compartilhado às pastas sem proteção ou autenticação por senhas. Fica relativamente fácil portanto, para um *hacker* conseguir acessar arquivos presentes nestas pastas compartilhadas (UNISINOS, 1998).

3.1.14. Ataques Contra o X-Window

O sistema X-Windows é o padrão de interface gráfica utilizada em *workstations* com sistema operacional derivado do Unix. Este padrão foi criado para ser utilizado em redes locais. Desta forma é possível executar um programa em um computador e ver seu resultado em outro, localizado em qualquer ponto da Internet. É possível também, executar programas em diversas máquinas e visualizar todos como se fossem aplicações locais, cada um em sua janela (UNISINOS, 1998).

Toda essa flexibilidade, entretanto, tem problemas de segurança: caso disponha de autorização do servidor X-Windows, é possível para um atacante exercer controle total sobre a tela do cliente, como o movimento do *mouse* ou as funções do teclado (ibidem).

Esta autorização muitas vezes não é difícil de ser obtida pois o X-Windows possui dois métodos de autenticação, um no nível de usuário e outro no nível de máquina. Este último permite que um ataque por IP *spoofing* dê acesso a este recurso (o ambiente gráfico) do sistema operacional (UNISINOS, 1998).

Os servidores X-Windows utilizam portas com números bem definidos, sendo que o primeiro servidor de cada computador utiliza a porta 6000, o segundo 6001 e assim sucessivamente. Portanto, basta bloquear no *host* o acesso externo a estas portas (ibidem).

3.1.15. Ataques Utilizando o RIP

O RIP é um protocolo de configuração dinâmica de rotas. Os roteadores utilizam este protocolo para transmitir informações para outros roteadores (normalmente em *broadcast*). Em geral, cada roteador, anuncia de tempos em tempos para os roteadores ao seu redor, que ele é capaz de rotear pacotes para determinada rede, de tal forma que em pouco tempo, a partir destas mensagens, todos tenham uma tabela única e otimizada de roteamento. (UNISINOS, 1998)

O problema principal do protocolo RIP é o de que as informações não são checadas. Um atacante pode, portanto, enviar informações falsas e maliciosas de roteamento para outros roteadores, de tal forma que possa “fingir” que é outra máquina, pretensamente confiável, ou até mesmo para gravar todos pacotes enviados para uma determinada máquina (ibidem).

Um ataque típico é o de repassar, para os roteadores da rede, a informação de que os pacotes enviados para uma determinada rede devem ser roteados através dele, atacante, que posteriormente os envia para o destino correto. Durante a passagem do pacote, o atacante é capaz de examinar seu conteúdo em busca de senhas ou qualquer outra informação que julgar interessante ou, até mesmo, modificar seu conteúdo (UNISINOS, 1998).

Os pacotes RIP são enviados para a porta 513 UDP. Portanto bloquear o acesso a essas portas pode solucionar este problema.(ibidem)

3.2. Implementações de Ataques

CURY (2000), descreve como ocorrem os ataques mais comuns:

- Hanson: explora os soquetes criados pelo *software* cliente de IRC (*Internet Relay Chat*) fazendo-o travar. Geralmente estes *softwares* (como o Mirc, para Windows) aceitam um volume de dados maior do que conseguem manipular, ocasionando uma necessidade de armazenamento em *buffer* maior do que o limite disponível. O resultado é o travamento do sistema operacional. Alvo: Usuários de cliente de IRC.

- Beer: é o envio de pacotes ICMP (*Internet Control Message Protocol*) com endereços de origem aleatórios e falsos, causando, no computador de destino, a necessidade de conferir a origem de cada um destes pacotes, elevando o uso do processador à níveis que o façam parar de responder para o usuário local. Alvos: Sistemas operacionais Windows.

- Teardrop: este algoritmo explora um bug no módulo de fragmentação de endereços IP. Pode causar indisponibilidade no computador alvo. Alvos: Linux, com *kernel* inferior à versão 2.0.32 e Windows.

- Win nuke: causa negação de serviço, derrubando a conexão do protocolo IP. Isto é feito explorando a porta TCP 139, que atende à requisições NetBIOS. Vários problemas ocorrem. O mais comum é a falha geral de proteção do sistema operacional, o GPF (*General Protection Fault*). Alvos: Sistemas operacionais Windows.

- Land: este algoritmo causa a negação de serviço através do envio de pacotes que têm endereço de origem falso gerado aleatoriamente, causando no alvo a necessidade de conferir a origem de cada um deles. O computador fica sem recursos para atender ao usuário. Alvos: Windows NT 4 e Workgroups.

- Coke: dependendo do estado da configuração de *logging* do *host*, este algoritmo pode causar aumento da ocupação do espaço em disco arbitrariamente. Alvo: *Host* Windows NT com serviço WINS.

- Open tear: consiste no envio de uma grande quantidade de pacotes UDP para o alvo. Geralmente isto causa a negação de serviço, fazendo com que o *host* não atenda aos usuários. Alvos: Windows e Linux.

4. ENGENHARIA SOCIAL

LIMA (2000) define Engenharia Social da seguinte forma "É o método de se obter dados importantes de pessoas incautas através da velha e conhecida lábia."

Já FONTES (2001), profere que o termo Engenharia Social é relativamente novo, porém, o assunto não, definindo-o assim: “É aquela conversa que encanta quem está ouvindo e faz com que esse ouvinte fique com total confiança em quem está falando.”

Estas duas citações têm muito a ver com a segurança da informação. Quando dizemos que a proteção da informação deve ser tratada de forma profissional, significa que ela deve ser estruturada, deve ter regras e normas explícitas e ser válida para todos. E isto deve ser de conhecimento de todos. Tudo isto acaba inválido se um assessor da presidência da empresa for barrado no acesso físico de uma área restrita e no outro dia o vigilante que estava cumprindo a norma for mandado embora por este fato. Esta situação demonstra que a organização não deseja ter uma segurança efetiva, mas sim, uma segurança “faz de conta” (FONTES, 2000).

Na opinião de ANTIHACKERS (2000), o método mais simples, mais usado e, infelizmente, mais eficiente de se descobrir uma senha é perguntando ao detentor dela. Basta que alguém convença um funcionário mal informado que ele acaba contando. Pode não ser a senha, mas ele vai contar o tipo de sistema, o tipo de computador, e o que mais ele souber ou lembrar. Tudo depende de quão eficiente é o "Engenheiro Social", e quais conhecimentos sobre a empresa o alvo possui.

Prestar atenção no tipo de informação que sai da empresa, nos papéis jogados no lixo e na entrada das pessoas estranhas são formas comuns de prática da engenharia social. Outra estratégia é encontrar um organograma da empresa. A partir daí, o intruso vai saber com quem estará falando, podendo se fazer passar, inclusive, por uma pessoa de maior hierarquia, como o diretor de informática, por exemplo (ibidem).

O *hacker* se passa por outras pessoas, enganando os funcionários da empresa. Para poder fazer um "teatro" convincente, ele utiliza informações (nomes de usuários ou, do administrador) coletadas previamente. Com isto o invasor consegue obter informações privilegiadas (p.e. senhas), ou então induzir funcionários a executar ações que enfraqueçam a segurança (p.e. executar um *trojan* ou uma reinicialização de senha) (NETSEC INTERNET SECURITY, 2000).

4.1. Exemplos de Ataque de Engenharia Social

Um usuário recebe um *e-mail* do *hacker* (se passando pelo administrador) solicitando que a sua senha seja alterada para "x". Se o usuário proceder conforme solicitado, o *hacker* saberá qual sua senha, e poderá passar a utilizar as credenciais deste usuário (NETSEC INTERNET SECURITY, 2000).

O administrador da rede recebe um e-mail ou telefonema de um *hacker* (alegando ser um usuário), solicitando a reinicialização da sua senha. Isto permite ao *hacker* logar com a senha deste usuário (ibidem).

4.2. Ações Contra Ataques De Engenharia Social

LIMA (2000) recomenda que nas organizações exista uma política de segurança centralizada e bem divulgada para que todos saibam a quem recorrer em casos de dúvida, além de orientação e esclarecimentos aos usuários. Não há necessidade de desconfiar de todas as solicitações de informação, basta estar sempre alerta para pedidos de dados sigilosos, e nunca divulgar senha nenhuma em qualquer circunstância, pois a mesma perde o sentido se não se mantiver secreta.

5. FIREWALLS

Os *firewalls,* para UNISINOS (1998), protegem a rede interna de trabalho contra os perigos da Internet (Figura 2). Eles servem para múltiplos propósitos:

- Restringir a entrada de pessoas no sistema.
- Prevenir contra ataques que possam atingir defesas.
- Restringir ou limitar acessos dos usuários internos à Internet.

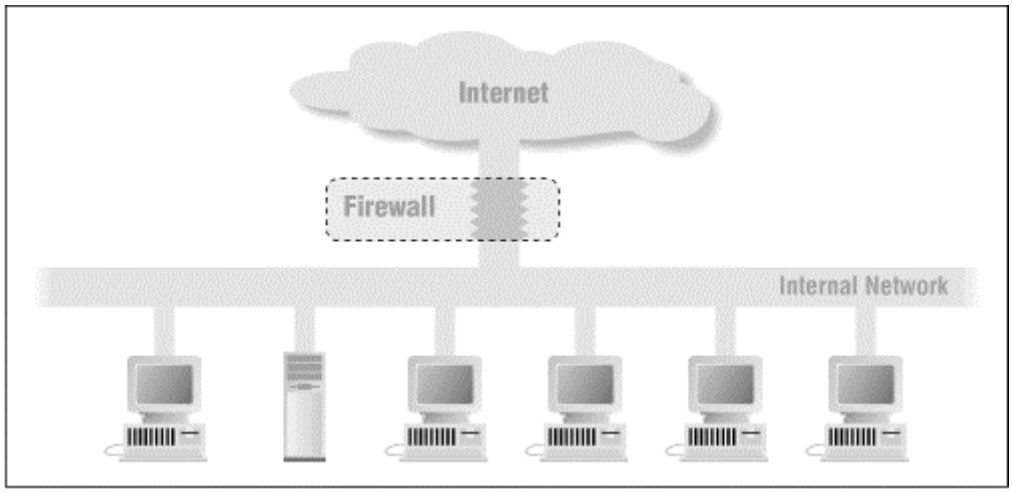

Figura 2 Posicionamento do *firewall* na rede de computadores.
Fonte: (ZWICKY et al. 2000)

Para se compreender a tecnologia utilizada nas implementações de *firewalls*, é preciso ter conhecimento dos objetos com os quais o *firewall* lida: pacotes e protocolos. Para transferir informações através de uma rede, estas informações devem ser quebradas em pequenas partes, transferidas uma a uma separadamente. Quebrar as informações em pedaços permite a muitos sistemas dividirem a rede, cada um enviando partes de suas informações na sua vez. Em uma rede IP, estes pedaços de informação são chamados de pacotes. (ZWICKY et al. 2000)

5.1. Características de um Pacote

Para entender a filtragem de pacotes, deve-ser entender como eles são classificados na pilha TCP/IP, sendo elas:

- Camada de aplicação (exemplos: FTP, Telnet, HTTP)
- Camada de transporte (TCP e UDP)
- Camada Internet (IP)
- Camada de acesso à rede (exemplos: Ethernet, FDDI, ATM)

Os pacotes são construídos de uma forma que as camadas para cada protocolo usado para uma conexão em particular sejam quebrados em pacotes menores, ficando encapsulados de uma forma concêntrica (ZWICKY et al. 2000).

Em cada camada, um pacote tem duas partes: o cabeçalho e o corpo. O cabeçalho contém informações relevantes para aquela camada, enquanto o corpo contém os dados daquela camada, que consiste de um pacote completo para a próxima camada superior. Cada camada trata as informações que recebe da camada acima e aplica seu cabeçalho ao dado. Este processo de se preservar os dados enquanto se anexa novos cabeçalhos é chamado de encapsulamento (Figura 3) (ibidem).

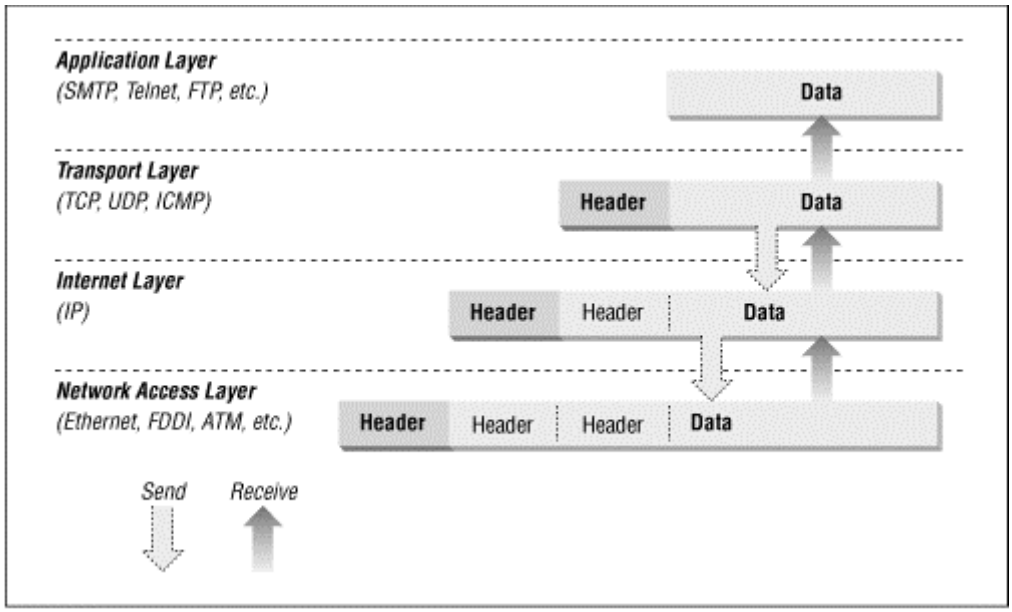

Figura 3 Camadas de um pacote IP.

Fonte: (ZWICKY et al. 2000)

5.1.1. Exemplo TCP / IP / Ethernet

Considerando um pacote TCP/IP em uma rede Ethernet , (ZWICKY et al. 2000) examina o conteúdo dos dados e cabeçalhos em cada camada existente no exemplo em questão, sendo elas:

- Camada Ethernet.
- Camada IP.
- Camada TCP.
- Camada de dados.

5.1.1.1. A Camada Ethernet

Na camada Ethernet, segundo ZWICKY et al. (2000), o pacote consiste de duas partes: o cabeçalho Ethernet e o corpo Ethernet. Em geral, não é possível filtrar pacotes com base nas informações contidas neste pacote. Em algumas situações, pode-se estar interessado entretanto, nas informações de endereçamento deste protocolo, conhecido como MAC. Basicamente este cabeçalho informa:

- O tipo do pacote: neste exemplo, o pacote é IP, não é portanto um pacote AppleTalk, ou um pacote IPX, nem um pacote DECnet.
- O endereço Ethernet do computador que pôs o pacote para transmissão na rede.
- O endereço Ethernet do computador que deve receber o pacote.

5.1.1.2. A Camada IP

ZWICKY et al. (2000) diz que, nesta camada, o pacote IP é também composto de duas partes: o cabeçalho IP (Figura 4) e o corpo IP. Do ponto de vista de um filtro de pacotes , o cabeçalho IP contém informações úteis:

- O endereço IP de origem.
- O endereço IP de destino.
- O tipo de protocolo IP (TCP ou UDP).
- O campo de opções.

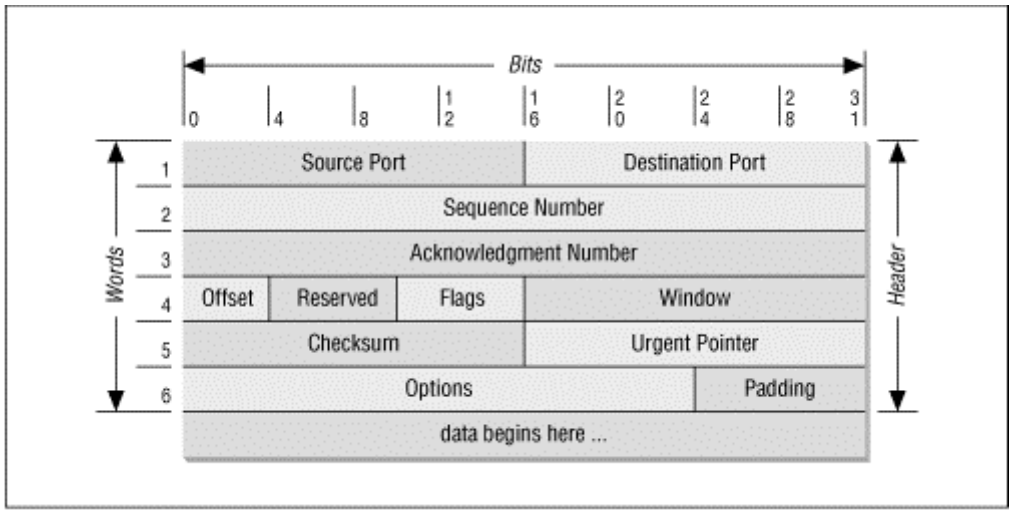

Figura 4 Definição do cabeçalho e corpo de um pacote TCP/IP.
Fonte: (ZWICKY et al. 2000)

A maioria das redes especificam um tamanho máximo para o pacote, que é muito maior que o limite imposto pelo IP. Para sanar este conflito, o IP divide os pacotes grandes demais para trafegar na rede em séries de pacotes, chamadas fragmentos. A fragmentação (Figura 5) não muda as estruturas da camada IP (os cabeçalhos IP são duplicados em cada fragmento), mas isto pode significar que o corpo contém apenas uma parte do pacote na próxima camada (ZWICKY et al. 2000).

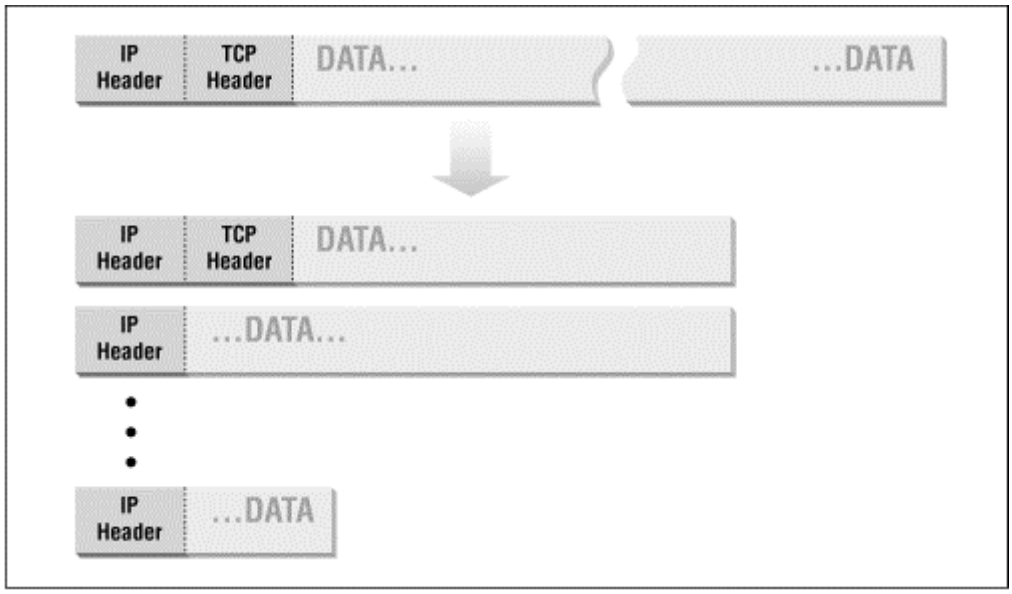

Figura 5 A Camada TCP.
Fonte: (ZWICKY et al. 2000)

Nesta camada, ZWICKY et al. (2000) relata que o pacote novamente consiste de duas partes: o cabeçalho TCP e o corpo TCP. Do ponto de vista do *firewall*, o cabeçalho TCP contém três informações interessantes:

- A porta TCP de origem.
- A porta TCP de destino.
- O campo de *flags*.

O corpo TCP contém a informação propriamente dita, por exemplo, o caractere transmitido em uma sessão Telnet, ou o arquivo transferido em uma conexão FTP.

6. SERVIÇOS DA INTERNET

Saber quais serviços serão prestados pela infra-estrutura da corporação é um passo muito importante durante a fase de projeto dos equipamentos computacionais que vão compor uma rede. Portanto todo modelo que difere nos serviços a prestar, difere na configuração dos equipamentos de segurança. Entretanto para ZWICKY et al. (2000), a maioria das redes atende sempre a cinco serviços básicos:

- *World Wide Web access* (HTTP);
- Correio eletrônico (SMTP);
- Transferência de arquivos (FTP);
- Acesso de terminal remoto (TELNET e/ou SSH);
- Conversão de nomes de domínio em endereços IP (DNS).

Todos estes serviços são regulamentados e podem ser providos de forma segura de diversas formas, inclusive estando protegido por filtros de pacotes ou por sistemas *proxy* (Figura 6).

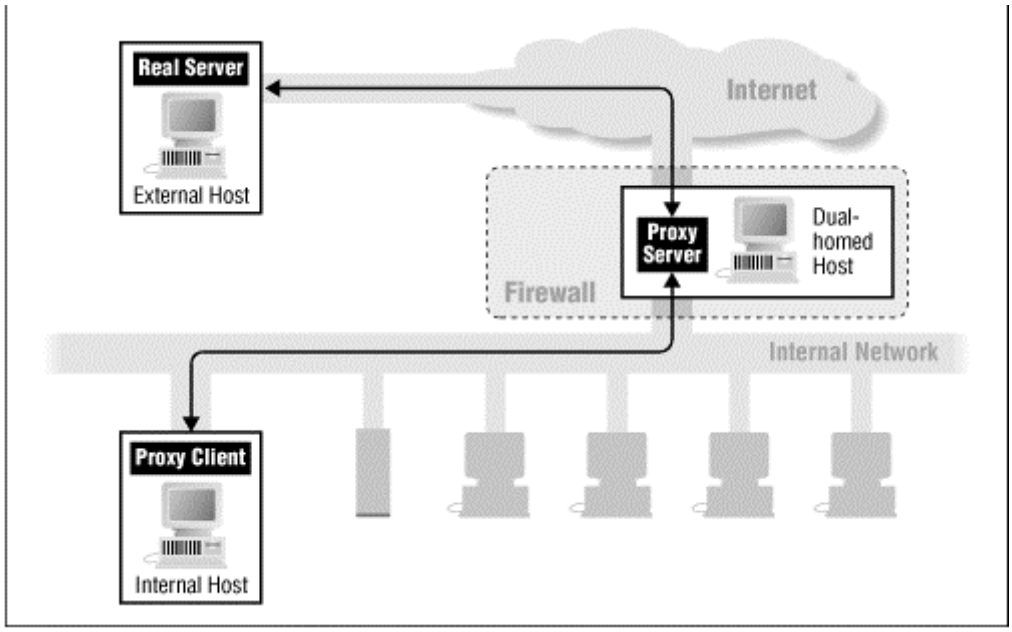

Figura 6 Usar *proxy* para redirecionar pedidos do cliente.
Fonte: (ZWICKY et al. 2000)

6.1. A World Wide Web

Atualmente, o serviço de *world wide web* é um dos mais populares da Internet. Para que se entenda claramente do que é composta esta grande fatia da Rede pode-se dividi-la nas seguintes estruturas:

- A *Web*;
- O protocolo http (*hypertext transfer protocol*);
- A linguagem HTML(*hypertext markup language*);
- Os navegadores.

6.1.1. A Web

Consiste na coleção de servidores na Internet que atendem a requisições no protocolo de comunicação HTTP. É baseado em conceitos desenvolvidos na *European Particle Physics Laboratory* (CERN) em Genebra, na Suíça, por Tim Berners-Lee e outros pesquisadores. Hoje, muitas organizações e indivíduos estão desenvolvendo este serviço da rede e um número muito maior de empresas e pessoas estão fazendo uso desta tecnologia. A *Internet Engineering Task Force* (IETF) é responsável por manter o HTTP padronizado e o *World Wide Web Consortium* (W3C) por desenvolver seus futuros sucessores (ZWICKY et al., 2000).

6.1.1.1. Considerações sobre segurança

Quando se mantém um *web server*, se permite que qualquer pessoa alcance este computador e envie comandos para ele. Os riscos de segurança se encontram quando o *software* servidor deixa de somente manipular HTML para chamar acesso a programas ou módulos externos que ampliam as suas capacidades. Estes programas são relativamente fáceis de se escrever porém difíceis de se implementar porque não se consegue prever todas as passagens de comandos que poderá receber e o comportamento que terá em sua execução.

6.1.2. O protocolo HTTP

É um dos protocolos que fazem parte da camada de aplicação do TCP/IP. Provê aos usuários, acesso a arquivos que fazem a *Web*. Estes podem ter diferentes formatos (texto, gráficos, áudio, vídeo etc.) mas primariamente ligados através de *hyperlinks* e programados na notação *HyperText Markup Language* (ZWICKY et al. 2000).

6.1.3. A linguagem HTML

Descrição padronizada para criação de páginas. Basicamente, provê capacidades de formatação de documentos, inclusão de gráficos e marcação de referência à outros documentos através dos *hyperlinks* (ZWICKY et al. 2000).

6.2. Correio eletrônico

O correio eletrônico provê troca de mensagens eletrônicas umas com as outras sem a necessidade de requerer respostas imediatas ou em tempo real. É também o serviço mais antigo e também, um dos mais populares. Relativamente de baixo risco (mas isto não significa que seja risco-zero), trivialmente se forja mensagens (assim como no serviço postal) e isto facilita dois tipos de ataque:

- Ataques contra a reputação;
- Ataques de manipulação social.

SMTP é o protocolo padrão na Internet para o transporte de mensagens entre computadores. Os servidores de SMTP, como outros programas, diferenciam-se entre orientação a recursos e orientação à segurança.

Enquanto o SMTP transporta mensagens entre servidores, os clientes que recebem mensagens não usam este protocolo porque a leitura de mensagens diretamente no servidor se tornou não usual. Na Internet, os protocolos mais comuns para o descarregamento das mensagens no *software* cliente são o *Post Office Protocol* (POP) e o *Internet Message Access Protocol* (IMAP) (ZWICKY et al. 2000).

6.2.1. Considerações sobre segurança

Servir correio eletrônico toma espaço em disco e tempo de processamento no *host*, ficando aberto para ataques de negação de serviço e muitas vezes deixa-se um canal aberto para a entrada de cavalos-de-tróia. Também, é comum neste sistema o envio em massa de mensagens não solicitadas e as cartas “correntes”, bem como a confiança exagerada que as pessoas têm no serviço, fazendo seu uso para envio de informações confidenciais sem o mínimo de preocupação com os intermediadores entre o remetente e o destinatário (ZWICKY et al. 2000).

POP e IMAP têm as mesmas implicações de segurança, eles comumente transferem as informações de autenticação do usuário sem nenhum recurso de criptografia, permitindo a *hackers* lerem tanto suas credenciais quanto as mensagens que estão para ser transmitidas (ibidem).

6.3. Transferência de arquivos

File Transfer Protocol (FTP) é o serviço padronizado na Internet para transferência de arquivos. A maioria dos navegadores baseados no protocolo HTTP também implementam este protocolo como uma extensão dos seus recursos.

6.3.1. Considerações sobre segurança

A primeira preocupação em servir o protocolo FTP é que ele pode atuar como ponto de entrada para cavalos-de-tróia e outros códigos maliciosos que exploram falhas na implementação deste serviço. Outras vezes o motivo da preocupação é sobre o licenciamento do conteúdo que é enviado para o servidor. Entre eles softwares comerciais, jogos copiados ilegalmente e conteúdo pornográfico. Apesar de não apresentarem um risco técnico de segurança, este conteúdo implica em vários outros problemas relevantes, como distribuição ilegal de *software* e infrações fiscais.

Algumas recomendações para sistemas que precisam servir este serviço são:

- Não permitir *upload* de usuários não identificados pelo sistema, mais precisamente *upload* como usuário anônimo.
- Educar os usuários, inclusive com penalidades descritas no documento de políticas de segurança, sobre o conteúdo que os mesmos disponibilizam.
- Monitorar periodicamente o conteúdo dos diretórios dos usuários.
- Implementar quotas no sistema de arquivos de modo a limitar a quantidade de espaço que cada usuário pode ocupar.

6.4. Terminais Remotos

Há muitas situações em que é necessário ou ao menos, desejável, executar comandos em computadores diferentes daquele que se está fazendo uso. Muitas vezes o motivo que justifica esta necessidade é o fato de se estar fisicamente muito distante do computador a controlar. Como solução, implementou-se serviços capazes de prover esta tarefa, tanto em computadores com sistema operacional orientado ao modo texto quanto nos que utilizam interfaces gráficas.

Questões relevantes para ZWICKY et al. (2000) sobre o fornecimento deste tipo de serviço são:

- Há controles apropriados para identificar quem pode acessar o computador remotamente? De que forma são autenticados os usuários?
- É possível algum intruso tomar conta de uma conexão em andamento?
- É possível alguma ferramenta de análise de rede capturar informações sigilosas, particularmente, as credenciais do usuário?

Telnet está, cada vez mais, deixando de ser o serviço padronizado na Internet para tarefas de administração remota. Porém, este serviço já foi considerado seguro por requerer credenciais do usuário. Comparado com os serviços de execução remota de comandos *rsh e rlogin*, isto realmente faz sentido. Entretanto, estas credenciais podem ser tão facilmente capturadas através da Rede que seu uso se torna muito arriscado (ZWICKY et al. 2000).

Para solucionar as sérias falhas do Telnet, criou-se o tão bem aceito Secure Shell (SSH), que provê uma coleção de utilitários capazes de fornecer serviços criptografados de execução remota de comandos, como também, transferência segura de arquivos. Implementações deste protocolo estão disponíveis através de vários fornecedores e estão se tornando o novo padrão no que diz respeito à administração remota (ibidem).

6.4.1. Considerações sobre segurança

Na utilização do *Secure Shell* de forma a evitar o uso indevido deste serviço, recomenda-se o seguinte:

- Explorar ao máximo as capacidades inovadoras para autenticação do usuário. Isto significa exigir mais que apenas o par *login* e senha. Como recurso embutido, as implementações SSH podem ser configuradas para exigir uma chave criptográfica pública do usuário que tenha sido gerada por uma chave privada anteriormente instalada no servidor. Este procedimento faz com que o usuário não só forneça algo que só ele tenha conhecimento (senha) mas também algo que somente ele possua (o par de chaves criptográficas).

- Quando financeiramente viável, exigir identificação através de recursos menos burláveis ainda, como autenticação por *SmartCards*, que são cartões, (parecidos com o de crédito) que armazenam a chave pública do usuário e são praticamente impossíveis de se forjar.

6.5. Conversão de nomes-de-domínio em endereço IP

A tradução de nomes que os usuários usam em endereços numéricos que os protocolos necessitam é implementada pelo *Domain Name System*. No início da Internet era possível para cada servidor manter uma tabela de *hosts* presentes naquela rede. Mais tarde, com o número de computadores aumentando drasticamente, percebeu-se que esta tabela não poderia mais ser mantida. O DNS é uma solução de base de dados distribuída que atende requisições para tradução nome do domínio -> IP, e quando não sabe responder, consulta uma base presente um nível acima de sua hierarquia.

O resultado prático é que quem precisa servir na Internet, precisa do serviço DNS.

6.5.1. Considerações sobre segurança

Um dos riscos de se servir DNS é prover mais informações do que o necessário. Por exemplo: este protocolo permite embutir nas respostas informações sobre o *hardware* e o sistema operacional onde está sendo executado.

O DNS pode ficar mais seguro quando se toma algumas medidas, como:

- Obrigar a utilização de números IP (ao invés do *hostname*) para autenticação em serviços que precisam alto nível de segurança;

- Autenticar sempre os usuários e não a origem deles. Isto significa exigir autenticação do usuário mesmo quando a origem dele consta como segura na tabela de *hosts* do sistema operacional, porque esta origem pode estar sendo forjada.

7. POLÍTICAS DE SEGURANÇA

"A política de segurança, do ponto de vista da administração de sistemas, deve ser vista como a política de segurança utilizada por uma nação para receber estrangeiros." (ZWICKY et al. 2000).

"A Política de Segurança é um conjunto de regras, que têm por objetivo disciplinar o uso da rede de trabalho. Na política de segurança, deve se especificar quem tem acesso à rede, quais os direitos de cada usuário ou grupo, quais serviços são disponibilizados e para quais grupos. É importante também que a Política de Segurança tenha definido punições para cada tipo de infração cometida." (UNISINOS, 1998).

A palavra “política” sugere, para muitos, que se trata de documentos inacessíveis, formulados por grandes comitês e que são impostas a todos e, posteriormente, ignoradas com o passar do tempo. Porém, as políticas discutidas na administração de sistemas, são táticas, que envolvem a construção de um *firewall*, os detalhes de que serviços são necessários e quais são proibidos, entre outros. Entretanto, não importa a qualidade do conjunto de táticas, se a estratégia de aplicação destas for mal sucedida (ZWICKY et al. 2000).

Para RIBEIRO (1998), o objetivo da política de segurança resume-se em manter sob controle o armazenamento da informação, que muitas vezes, é o bem mais valioso de uma empresa devendo seguir estes 4 paradigmas básicos:

- Integridade: A condição na qual a informação ou os recursos da informação são protegidos contra modificações não autorizadas.
- Confidencialidade: Propriedade de certas informações que não podem ser disponibilizadas ou divulgadas sem autorização prévia do seu dono.
- Disponibilidade: Característica da informação que se relaciona diretamente a possibilidade de acesso por parte daqueles que a necessitam para o desempenho de suas atividades.
- Legalidade: Estado legal da informação, em conformidade com os preceitos da legislação em vigor, no que se refere à aplicação de medidas punitivas.

De outra forma, ZWICKY et al. (2000) considera 4 questões para a formulação da política de segurança:

- Capacidade financeira: Quanto custa a segurança?
- Funcionalidade: Pode-se utilizar o sistema de forma plena?
- Compatibilidade cultural: A política de segurança proposta está em conflito com a forma como as pessoas normalmente interagem com os que estão do lado de fora da empresa?
- Legalidade: A política de segurança está de acordo com os requerimentos legais exigidos?

7.1. Conteúdo de Um Documento de Políticas de Segurança

Um documento de políticas de segurança é uma forma de comunicação entre administradores e usuários. Deve explicar a eles o que precisarem saber quando houver dúvidas sobre segurança (ZWICKY et al. 2000).

7.1.1. Explicações

É importante que o documento seja explícito e compreensível sobre todas as decisões a serem tomadas. A maioria das pessoas não irá seguir as regras a não ser que compreendam a sua importância.

7.1.2. Responsabilidade de Todos

Um documento de políticas de segurança esclarece expectativas e responsabilidades entre usuários e administradores; permite a todos saberem o que esperar de cada um.

7.1.3. Linguagem Clara

A maioria das pessoas estão acostumadas com explicações casuais. Para o administrador, pode ser desconfortável elaborar um documento casual, porque pode parecer ao mesmo tempo, muito pessoal. Porém, é mais importante fazer este documento amigável e compreensível do que fazê-lo preciso e com formato oficial.

7.1.4. Autoridade de Execução

Escrever o documento de políticas de segurança não é tão importante quanto segui-lo. Significa que quando a política não é seguida, algo deve acontecer para consertar a situação e alguém precisa ser responsável por fazer as correções.

Alguns exemplos do que deve especificar a política de segurança:

- Gerentes de certas áreas têm autoridade de revogar acesso a um usuário.
- Gerentes são responsáveis por efetuar correções em casos de transgressão.
- Penalidades que estão reservadas aos transgressores.

7.1.5. Exceções e Revisões

Nenhuma política de segurança é perfeita. Não se pode cobrir qualquer evento futuro. Entretanto, pode-se especificar as possíveis providências e as possíveis penalidades em casos que não constem no documento, sempre sujeito a revisões e reavaliações futuras.

Muitas vezes, regras que são impostas a seis funcionários não se adaptam a sessenta. Tudo na informática muda muito mais rápido do que em qualquer outra área, desde o *hardware* até a área de atuação de mercado da empresa. E isto implica na necessidade de revisões constantes no documento de políticas de segurança.

7.1.6. Itens Que Não Devem Estar Presentes em um Documento de Políticas de Segurança

Algumas características sobre assuntos que não devem constar em um documento de políticas de segurança:

Detalhes Técnicos

O documento deve descrever o que se está tentando proteger, e por quê. Não deve necessariamente, descrever detalhadamente de quê forma. É muito mais útil se ter um documento de uma página descrevendo o quê e por quê do que ter dez páginas descrevendo como.

Comparações Externas

Todos os documentos de políticas de segurança são diferentes, pois, empresas diferentes têm conceitos diferentes, usuários diferentes e capacidades diferentes. Não se pode portanto, esperar bons resultados utilizando uma política já existente e alterando apenas os nomes escritos nela.

7.1.7. Principais Ameaças da Política de Segurança da Informação

Para RIBEIRO (1998), as principais ameaças que devem ser tratadas na Política de Segurança da Informação são:

- Integridade:
    - Ameaças de Ambiente (fogo, enchente, tempestade...).
    - Erros humanos.
    - Fraudes.
    - Erro de processamento.
- Indisponibilidade:
    - Falhas em sistemas ou nos diversos ambientes computacionais.
- Divulgação da Informação:
    - Divulgação de informações premeditada.
    - Divulgação de informações acidental.
- Alterações não Autorizadas.
    - Alteração premeditada.
    - Alteração acidental.

III - DESENVOLVIMENTO

1. INTRODUÇÃO

Para que se possa entender a importância e a necessidade de se manter seguros os sistemas que hospedam conteúdo na Internet, são apresentadas aqui algumas informações a respeito do uso da Internet para fins maliciosos.

A Tabela 2 informa em valores numéricos a quantidade de ataques ocorridos no Brasil que foram reportados ao NIC BR Security Office, entidade mantida pelo Comitê Gestor da Internet no Brasil, que responde aos ataques com procedimentos legais de investigação.

Tabela 2 Totais Mensais e Anual Classificados por Tipo de Ataque.

Ano 2000															
Mês	Total	axfr (%)		af (%)		dos (%)		invasão (%)		aw (%)		scan (%)		fraude (%)	
Jan	424	28	6,60	108	25,47	11	2,59	3	0,71	57	13,44	217	51,18	0	0,00
Fev	509	61	11,98	78	15,32	8	1,57	11	2,16	80	15,72	270	53,05	1	0,20
Mar	541	55	10,17	117	21,63	14	2,59	8	1,48	38	7,02	309	57,12	0	0,00
Abr	351	19	5,41	93	26,50	17	4,84	5	1,42	22	6,27	194	55,27	1	0,28
Mai	480	12	2,50	145	30,21	15	3,12	11	2,29	28	5,83	267	55,62	2	0,42
Jun	641	7	1,09	215	33,54	8	1,25	17	2,65	20	3,12	374	58,35	0	0,00
Jul	585	2	0,34	80	13,68	13	2,22	17	2,91	22	3,76	450	76,92	1	0,17
Ago	432	2	0,46	112	25,93	6	1,39	13	3,01	16	3,70	280	64,81	3	0,69
Set	337	3	0,89	90	26,71	6	1,78	15	4,45	12	3,56	209	62,02	2	0,59
Out	468	0	0,00	139	29,70	4	0,85	7	1,50	17	3,63	301	64,32	0	0,00
Nov	573	2	0,35	167	29,14	34	5,93	13	2,27	57	9,95	295	51,48	5	0,87
Dez	656	9	1,37	196	29,88	23	3,51	7	1,07	46	7,01	372	56,71	3	0,46
Total	5997	200	3,34	1540	25,68	159	2,65	127	2,12	415	6,92	3538	59,00	18	0,30

Fonte: NIC BR Security Office

Legenda:

- axfr: Tentativas de obter mapas de DNS.
- af: Ataque ao usuário final.
- dos: *Denial of Service* (negação de serviço).
- aw: Ataque ao servidor web.

Na Figura 7, uma representação gráfica dos valores acumulados:

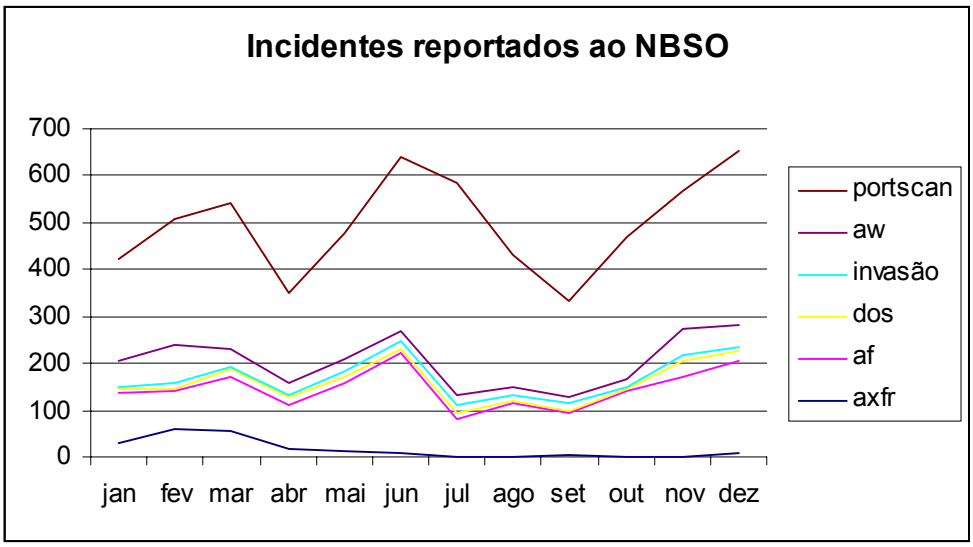

Figura 7 Incidentes - Valores acumulados

Fonte: NBSO (2001)

2. ESTRATÉGIAS DE SEGURANÇA

É importante entender algumas das estratégias básicas empregadas na construção de sistemas de segurança. Os tópicos mais relevantes são:

- Lei do privilégio mínimo;
- Defesa em profundidade;
- Ponto de aferição;
- Elo mais fraco;
- Segurança em caso de falha;
- Política de negar conexões como padrão;
- Política de aceitar conexões como padrão;
- Participação universal;
- Simplicidade;
- Segurança através de obscuridade.

2.1. Lei do privilégio mínimo

Um dos principais princípios de segurança é o fornecimento do privilégio mínimo. Neste sentido, qualquer objeto (usuário, administrador, programador, etc.) deve obter apenas privilégios suficientes para execução de suas tarefas. A lei do privilégio mínimo limita a exposição dos sistemas e reduz a gama de danos que podem ser causadas por ataques contra a falha na implementação dos serviços que se provê.

No contexto da Internet, cada usuário deve obter apenas o acesso que realmente necessita ao sistema. Por exemplo, os usuários não necessitam direitos de leitura a todos binários do sistema, os operadores de *backup* não necessitam todos privilégios de um administrador de sistemas. Um administrador de sistemas não necessita credenciais de acesso em sistemas que ele não mantém. Um sistema não necessita de privilégios de escrita em outro sistema.

A maioria dos sistemas operacionais não chega até os administradores com a lei do privilégio mínimo totalmente aplicada, forçando o administrador de sistemas a constantemente explorar novas formas de prover os mesmos recursos envolvendo sempre os menores privilégios, como por exemplo:

- Não conceder privilégios administrativos a um usuário se tudo que ele necessita é poder reiniciar o serviço de impressão. Para tal, inclua-o no grupo de operadores de impressão (criando um novo, onde for caso) e para este grupo conceder permissão de execução do binário que ativa o serviço.

- Não executar programas nem serviços com privilégios genéricos. Muitas vezes é comum haver no sistema um usuário genérico, sob o nome de *nobody* e a ele atribuir todos serviços. Este procedimento expande os privilégios de um atacante que invade o sistema.

- Não aceitar credenciais, incluindo as legítimas, se estas apresentarem como ponto de origem um dos sistemas que age como *firewall*, porque a segurança deles pode sempre estar comprometida.

Muitos dos problemas de segurança na Internet podem ser vistos como falha no cumprimento da lei do privilégio mínimo. Por exemplo, o Sendmail, que é o agente de transporte de mensagens eletrônicas mais utilizado na Rede, é complexo e tem todas as suas tarefas (coletar mensagens entrantes, escrever na caixa de mensagens do usuário, etc…) sob privilégios de um mesmo usuário (até o presente momento, *setuid to root*). Por necessitar de tanto acesso ao sistema este *software* continuará a receber muita atenção dos *hackers.* Isto implica em que programas superprivilegiados devem ser o mais simples possível e deve sempre haver formas de se separar e isolar partes de *software* que não precisam estar agrupadas em binários de execução complexa.

A aplicação da lei do privilégio mínimo é relativamente simples de se aplicar quando se quer reduzir privilégios de execução de *softwares* mas pode exigir cautela quando se tratar de pessoas. Deve haver cuidado para que os usuários realmente consigam fazer o que desejam e aceitem os limites definidos pelo administrador para realizar suas tarefas.

Em suma, a maioria das soluções que um administrador de sistemas aplica a uma rede resume-se a aplicação da lei do privilégio mínimo. Como exemplos tem-se o filtro de pacotes, que é um conjunto de regras cujo objetivo é a minimalização de privilégios que se concede a conexões entre redes, e o sistema de quotas em disco, que impede um usuário de utilizar mais espaço em disco do que o concedido.

2.2. Defesa em profundidade

Outro princípio de segurança é a defesa em profundidade. Um dos significados para este termo é: “não depender de um mecanismo de segurança em específico, por mais forte que ele pareça” . Em vez disso, deve se instalar vários sistemas que possam suportar uns as necessidades dos outros. O administrador de segurança em redes não deve esperar, nem arquitetar sua rede, com foco em apenas em uma única estratégia de segurança.

Apesar de os *firewalls* serem o principal instrumento de proteção, não se deve depositar unicamente nele todas as esperanças de proteção da rede interna. O objetivo deve ser: fazer o sistema muito arriscado ou muito custoso para ser invadido. Para tanto, somente um conjunto de técnicas diferentes pode garantir esta regra.

2.3. Ponto de aferição

Focalizar a atenção para os pontos de aferição do sistema pode tornar a detecção de intrusos uma tarefa mais simples. Exemplos destes pontos de aferição na vida cotidiana são:

- Pedágios;
- Caixas dos supermercados;
- Portão de entrada de uma casa.

Na segurança de redes e sistemas o *firewall* entre a Internet e a intranet é o maior ponto de aferição que o administrador tem com o atacante, então é muito importante saber monitorar este canal e principalmente, saber como agir caso um ataque se inicie.

Todo cuidado com um canal perde o sentido quando outros são deixados sem atenção. Por exemplo, de que adianta monitorar o *firewall* se existem inúmeras linhas *dial-up* aguardando conexões que dão acesso não filtrado aos servidores de informação?

O modelo de segurança orientado a *hosts*, que é aquele quando se colocam vários servidores atendendo um link, cada um com seu *firewall* (que na verdade não passa de um filtro pessoal de pacotes) e é um caso típico de falha na implementação de pontos de aferição.

2.4. Elo mais fraco

Uma estratégia para aumentar a segurança dos sistemas ligados à Internet é a idéia de que a força de uma corrente se mede por seu elo mais fraco. Este pensamento deixa a mensagem de que *hackers* espertos saberão atacar nos pontos mais fracos e vulneráveis do sistema. É necessário que o administrador esteja atento e seja capaz de classificar os pontos fracos e elimina-los quando possível, ou monitora-los com cuidado, quando sua existência é imprescindível.

Mais uma vez, a utilização de modelo de segurança orientado a *hosts* falha na aplicação deste conceito, porque torna todos “elos da corrente” pontos susceptíveis a ataques.

2.5. Segurança em caso de falha

Outro princípio de segurança é o de que os sistemas de informação devem ser a prova de falhas, ou seja, se falharem, a segurança do resto do ambiente não deve ser comprometida, mesmo que isto custe a inacessibilidade do sistema até mesmo por quem possui credenciais para utiliza-lo até que os reparos sejam feitos.

Os sistemas elétricos são projetados para se desligar, ou até mesmo queimar, quando falham; Os elevadores de uma construção têm presilhas para freá-los em caso de falha mecânica. Da mesma forma deve ser o comportamento dos sistemas computacionais que compõem o *firewall* de uma rede. A maioria dos sistemas já é segura em caso de falhas. Por exemplo, se um filtro de pacotes fica sem energia elétrica automaticamente o tráfego deixa de passar por ele, negando fisicamente todo e qualquer acesso à rede.

O fato mais importante para aplicação deste conceito é conhecer a postura geral da rede com relação à segurança, podendo ser agrupadas em duas:

- Postura de negar conexões em situações não regulamentadas;
- Postura de permitir conexões em situações não regulamentadas;

Pode parecer óbvia qual deve ser a postura geral de uma rede com relação a sua segurança, mas os motivos que fundamentam cada uma delas são os que seguem.

2.5.1. Negar conexões não regulamentadas

A política de se negar conexões como sendo a atitude para tudo que não está regulamentado faz sentido quando se usa o ponto de vista da segurança. O administrador da rede fica sabendo tudo que pode colocá-lo em risco e elimina tudo que for desnecessário, mas esta visão geralmente não cabe para o usuário.

Neste ponto de vista, nega-se toda conexão desconhecida como padrão e então se concede acessos de acordo com a necessidade. Isto exige do administrador as seguintes tarefas:

- Examinar quais serviços os usuários querem e necessitam;
- Fazer considerações sobre as implicações de segurança sobre a prestação de tais serviços;
- Permitir apenas os serviços que são do entendimento do administrador, e que podem ser providos com segurança, que têm motivo legítimo para serem usados e que estão de acordo com a política administrativa imposta pelos diretores da organização.

2.5.2. Política de aceitar conexões como padrão

A maioria dos usuários e gerentes não-técnicos preferem a política de aceitar conexões como padrão. Ela assume que tudo que não foi proibido será permitido mas, as práticas inseguras devem ser impedidas, como:

- Compartilhamento de arquivos SMB/CIFS (*Server Message Block* / *Common Internet File System*) e/ou NFS (*Network File System*) através da Internet;
- Proibição aos usuários de instalar *softwares* servidores em seus computadores pessoais;

- Assinatura de documento responsabilizando o usuário por seus atos na *World Wide Web* e demais serviços.

Vários conflitos entre administradores de sistemas e usuários surgem com a implantação desta política, como por exemplo, o compartilhamento de arquivos através da Internet, que é visto como extremamente inseguro pela maioria dos administradores, porém, é a primeira idéia que os usuários em geral têm quando necessitam trocar arquivos dentro e fora da rede de trabalho. A utilização de um servidor de FTP pode neste caso amenizar a situação, já que exige de ambos a necessidade de se autenticar em um sistema operacional designado para atuar como servidor.

2.6. Participação universal

Para ser completamente eficaz, a maioria dos sistemas de segurança necessita do comprometimento de todos, ou ao menos, não deve receber resistência daqueles que o mantém ou o gerencia. Se uma única pessoa puder optar por ficar excluída do sistema de segurança, então toda a eficácia deste sistema estará comprometida.

2.7. Simplicidade

Simplicidade é uma estratégia por dois motivos:

1. Manter os sistemas simples faz deles sistemas mais fáceis de se entender. Não dominar por completo um sistema, faz do administrador alguém incapaz de dizer se ele está seguro ou não;

2. A complexidade fornece um ambiente propício para ocultar falhas ao administrador.

Programas complexos costumam ter mais *bugs* que podem levar a falhas de implementação de *software*. Deve ser portanto, uma prática do administrador de sistemas, manter as tarefas simples de usar, manter e de administrar.

2.8. Segurança através de obscuridade

Este é o princípio de se proteger a rede escondendo-a. No dia a dia este procedimento se compara com o de não deixar objetos à vista dentro do automóvel mas sim guarda-los no porta luvas. Em termos computacionais os significados são:

- Instalar um computador e conecta-lo a Internet sem ninguém saber sobre a existência dele;
- Desenvolver algoritmos de criptografia próprios, ou fazer uso de outros incomuns;
- Instalar um servidor em uma porta diferente do padrão para fins de utilização interna;
- Fornecer informações diferentes para usuários de origem externa sobre os nomes-de-máquina da rede;
- Omitir versão do *software* que está atendendo a um serviço.

Em geral, discute-se muito a eficiência deste método, já que existem formas de detectar a presença de computadores não anunciados na rede através de *network scanning* (varredura de rede) ou de servidores em portas diferentes do padrão através de *port scanning* (varredura de portas de *hosts*). De fato, este método é ineficiente em muitos casos, mas quando usado em conjunto com os demais, certamente contribui para o aumento do nível de segurança da rede.

3. AVALIAÇÃO DE PROGRAMAS DE SEGURANÇA DA INFORMAÇÃO

Para auxiliar no desenvolvimento de um modelo de proteção de redes inovador, foi realizada uma pesquisa baseada em questionário (Anexo 1) com administradores de rede que se propuseram a informar quais conceitos e tecnologias são empregados nas redes que administram.

A Tabela 3 informa o resumo dos ambientes pesquisados.

Tabela 3 Resumo da pesquisa

Convidados	20 administradores de rede da região do Vale do Itajaí, em Santa Catarina, no Brasil.
Participantes efetivos	15.
Ambientes de trabalho visitados pelo autor.	08 ambientes.
Locais de trabalho dos entrevistados.	Provedores de acesso à Internet, redes acadêmicas e empresas comerciais.
Quantidade de questões	6 questões objetivas sendo 3 de escolha simples e 3 de múltipla escolha

3.1. Questão 1

A primeira questão "Qual o sistema operacional predominante em seus *hosts*?" tem o objetivo de fornecer o cenário da região com relação ao uso dos diversos sistemas operacionais existentes.

Tabela 4 Qual o sistema operacional predominante em seus *hosts*?

GNU/Linux	62%
FreeBSD	25%
Netware	13%

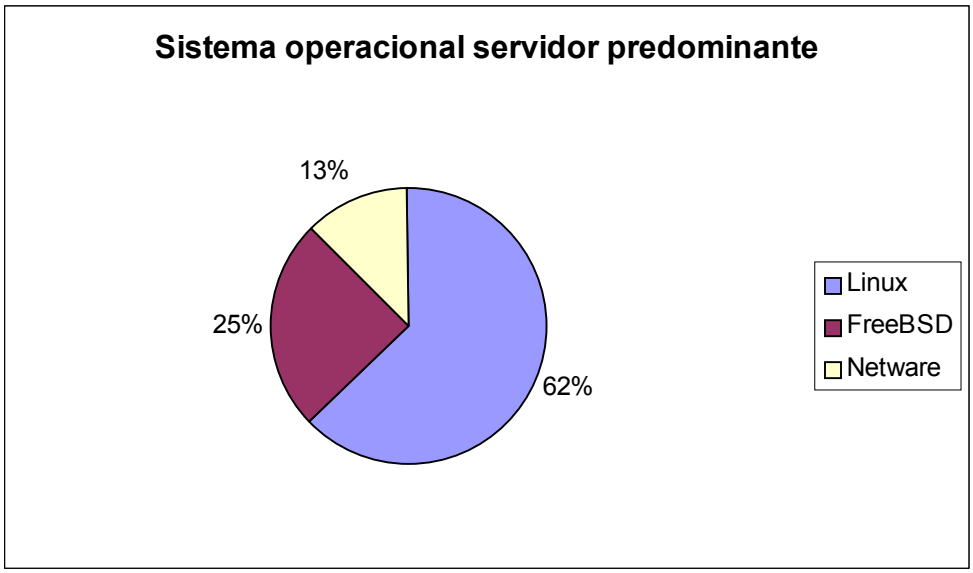

Figura 8 Qual o sistema operacional predominante em seus *hosts*?.

3.1.1. Análise

Fica evidente o predomínio de sistemas operacionais que implementam o padrão POSIX (*Portabe Operating System Interface*) sendo eles o GNU/Linux e o FreeBSD. Isto leva a concluir que o modelo de proteção de redes contará também com estes sistemas operacionais.

3.2. Questão 2

O segundo questionamento trata da maneira como está definida atualmente a política de uso e segurança dos sistemas e recursos computacionais que são fornecidos para os usuários da rede (clientes *dial-up* e dedicados no caso dos provedores de acesso; usuários da LAN para os demais).

Tabela 5 Como você define a política de segurança na rede que administra?

Existe uma política documentada, que estipula limites e penalidades.	13%
Faço recomendações verbais e informais para os usuários da rede.	74%
Não existe nada nesse sentido.	13%

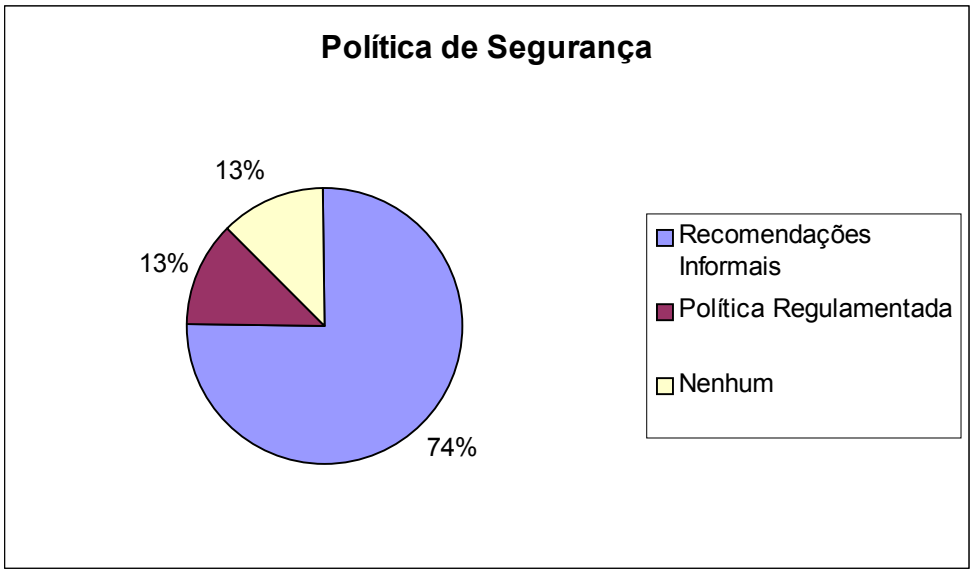

Figura 9 Como você define a política de segurança na rede que administra?

3.2.1. Análise

A importância da execução formal da política de segurança é relatada na Parte I deste trabalho. Entretanto, a maioria dos administradores de rede entrevistados desconhece ou não mantém vigente algum conjunto formal de normas sobre as normas de uso dos serviços prestados por seu setor ou sua empresa.

3.3. Questão 3

Na terceira questão "Quais dos seguintes conceitos de segurança você aplica na rede que administra?" obteve-se estes resultados:

Tabela 6 Quais dos seguintes conceitos de segurança você aplica na rede que administra?

Criptografia em terminais	14 administradores.
Filtragem de pacotes	10 administradores.
Zona desmilitarizada (DMZ)	03 administradores.
Centralização de autenticação (NIS, LDAP, Outros)	01 administrador.
Bridging	Nenhum administrador.

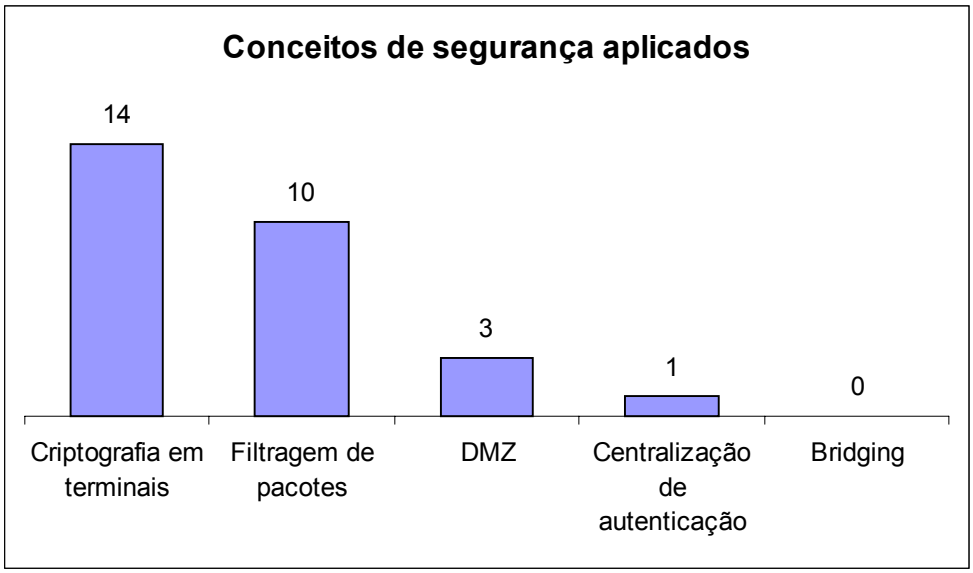

Figura 10 Quais conceitos de segurança você aplica na rede que administra?

3.3.1. Análise

A criptografia de conexões entre terminal e servidor se tornou fundamental. Essas conexões transportam a autenticação dos usuários e proteger-se contra ação de *sniffers* é esforço quase unânime para os participantes da pesquisa.

A filtragem de pacotes, em teoria, deve ser o resultado de um estudo profundo sobre a política de segurança, porque desse modo fica sendo apenas a aplicação prática daquilo que já se formalizou. Porém, como grande parte dos entrevistados não exerce documentação formalizada a filtragem fica a cargo do conhecimento técnico do administrador da rede.

Zonas desmilitarizadas fornecem uma camada adicional de segurança através da criação de uma terceira rede entre a Internet e a LAN. O uso desta técnica implica na adição de um roteador e de alterações na política de direcionamento de tráfego. Estes motivos, em conjunto com o relativo aumento dos custos de projeto e manutenção da rede, justificam o baixo índice de utilização da topologia DMZ.

Os sistemas centralizados de autenticação, como LDAP (*Lightweight Directory Access Protocol*), NIS (*Network Information System*), NDS (*Novell Directory Service*) e *Active Directory* não são muito utilizados. Na pesquisa, isto se deu por dois motivos: O LDAP e o *Active Directory* são protocolos muito recentes, estando em fase de estudo para alguns administradores e ainda não tendo sido avaliados pelo restante. O NIS, que já existe a mais tempo, é inseguro, sendo portanto, deixado de lado.

O *Bridging* é um caso especial: não é usado por nenhum dos participantes da pesquisa. Os motivos são vários e estão descritos na questão 5 desta pesquisa. Entre eles:

- Impossibilidade de conciliar tempo para estudo do *bridge host* e dos pré-requisitos para seu funcionamento;
- Documentação sobre o assunto ainda limitada e pouco divulgada;
- O administrador admite desconhecer ou não saber fazer este método de segurança.

3.4. Questão 4

A quarta pergunta questiona o posicionamento do(s) filtro(s) de pacote(s) com relação à topologia de rede existente no local de trabalho do entrevistado. A tabela abaixo apresenta as respostas.

Tenho um (ou mais de um) *host* fazendo somente filtragem de pacotes.	22%
Mantenho um filtro de pacotes mas agrego nele outras funções. (*Proxy*, NAT, FTP etc...)	34%
Todos serviços que disponibilizo estão em um computador e nele filtro pacotes.	22%
Somente configuro a filtragem de pacotes nos servidores que atendem aos usuários.	11%
Não faço filtragem de pacotes.	11%

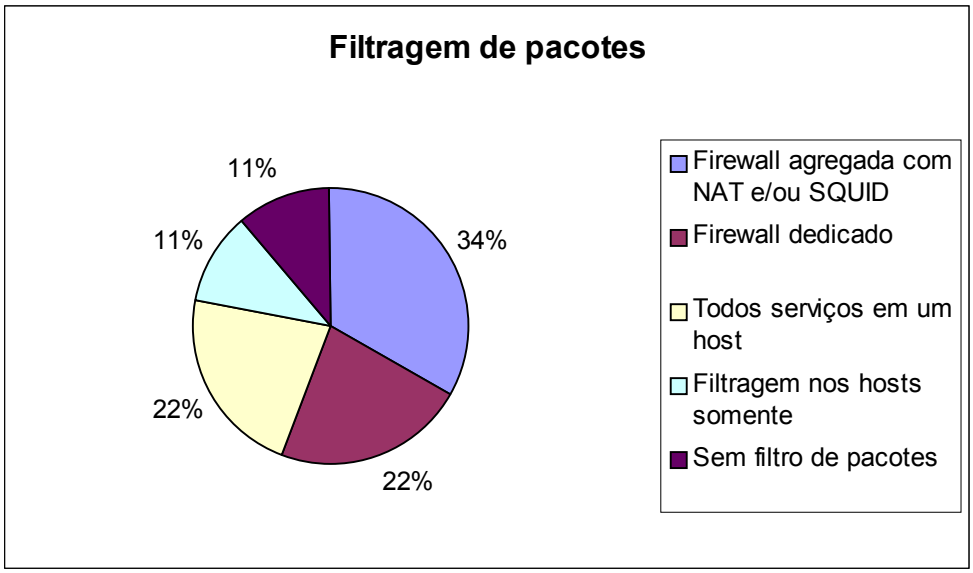

Figura 11 Posicionamento do filtro de pacotes.

3.4.1. Análise

O cenário mais comum que a pesquisa relata no que diz respeito a filtragem de pacotes é o do posicionamento do filtro no mesmo computador que faz a tradução de endereços e o *proxying*. Esta topologia é justa, pois o equipamento pode suportar a carga que lhe é dada. Mesmo assim existe um comprometimento do *firewall* porque este precisa receber endereçamento IP roteável sendo visível por toda a Internet. Esta topologia é comum nas redes acadêmicas e em algumas LANs comerciais.

A análise do questionário conclui que 22% dos administradores mantêm um filtro dedicado. Já que este(s) equipamento(s) recebe(m) endereçamento no protocolo IP subentende-se portanto que estão tão expostos quanto os *firewalls* com serviços agregados.

O motivo para que outros 22% dos participantes concentrassem todos serviços em um só equipamento é claro: custo. As organizações que mantém esta topologia optam por não investir em métodos de segurança além daqueles que não precisam novos investimentos. Esta situação ocorreu somente em empresas comerciais, não ocorrendo portanto com provedores de acesso nem nas redes acadêmicas.

Alguns administradores de rede optam por habilitar a filtragem de pacotes diretamente nos servidores. Esta topologia é comum em provedores de acesso e os benefícios do Modelo Proposto são muito evidentes neste caso, como será visto no próximo capítulo.

O restante (11%) não filtra pacotes. Nestes casos, geralmente o(s) servidor(es) não recebem atenção de administradores de rede, mas sim de profissionais que exercem outras tarefas no departamento de informática e que estão apenas começando a servir conteúdo na Internet.

3.5. Questão 5

Quinta questão: "Se você não mantém um *host* dedicado somente a filtrar pacotes, isto acontece por qual (quais) motivo(s)?" (Tabela 7)

Tabela 7 Motivos para não manter um filtro de pacotes dedicado.

Complexidade *versus* Tempo disponível	34%
Não sei porque ou como fazer	33%
Financeiramente inviável	22%
Não aumentará a segurança no meu caso	11%

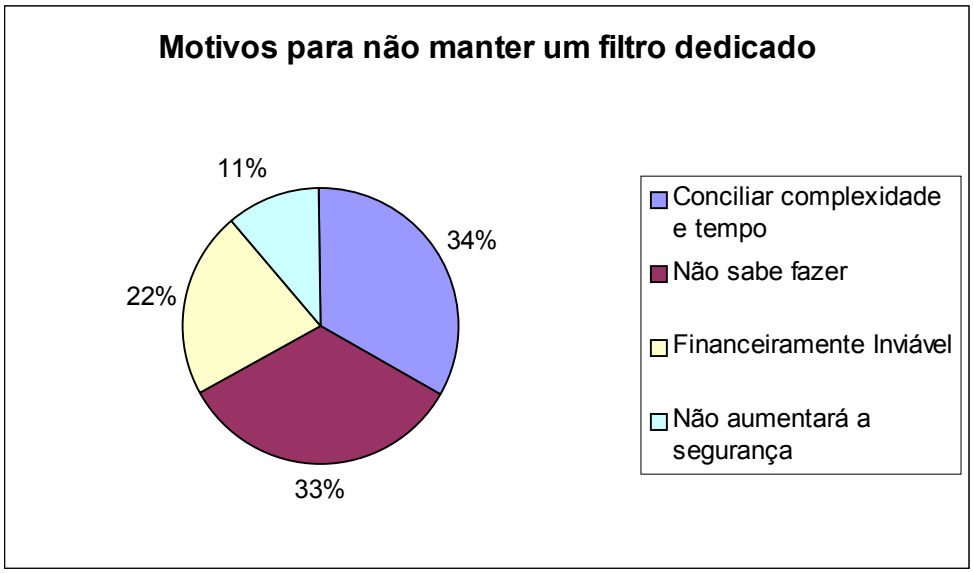

Figura 12 Motivos para não manter um filtro de pacotes dedicado.

3.5.1. Análise

Conciliar tempo para entender o funcionamento de um filtro dedicado é, neste caso, o principal motivo para que um administrador não o faça. Mais do que ser capaz de prepará-lo, este equipamento deve receber monitoramento contínuo porque nele são relatadas todas as anomalias detectadas nos pacotes IP pelos NIDSs (*Network Intrusion Detection System*). Neste ponto, surge a figura do analista de segurança, um profissional que trabalha em conjunto com o administrador de redes, porém focalizado no monitoramento da informação que trafega, na pesquisa por novos sistemas e nas tarefas contínuas de execução do plano de contingência e da política de segurança.

A falta de conhecimento sobre como se prepara e se mantém um *firewall* dedicado também é um dos motivos para os entrevistados não o fazer. Na maioria dos casos as redes começam pequenas, sem necessidades requintadas de segurança e depois a migração e o *downtime* necessário para execução de testes quase impossibilita esta melhoria. No Modelo Proposto sugere-se um *firewall* dedicado cujo *downtime* é muito próximo de zero ou, não maior do que o tempo necessário para uma troca de dois cabos de rede.

3.6. Questão 6

Em relação à sexta questão: “Qual(is) sistema(s) de detecção de intrusos estão em atividade na sua rede?” obteve-se os seguintes resultados:

Tabela 8 Qual(is) sistema(s) de detecção de intrusos estão em atividade na sua rede?

Outros	10 ocorrências.
Tcpdump	8 ocorrências.
Tripwire	4 ocorrências
Portsentry	3 ocorrências
Chkrootkit	2 ocorrências
Netsaint	2 ocorrências.
Logcheck	1 ocorrência.
Snort	Nenhuma ocorrência.
Nenhum	2 ocorrências.

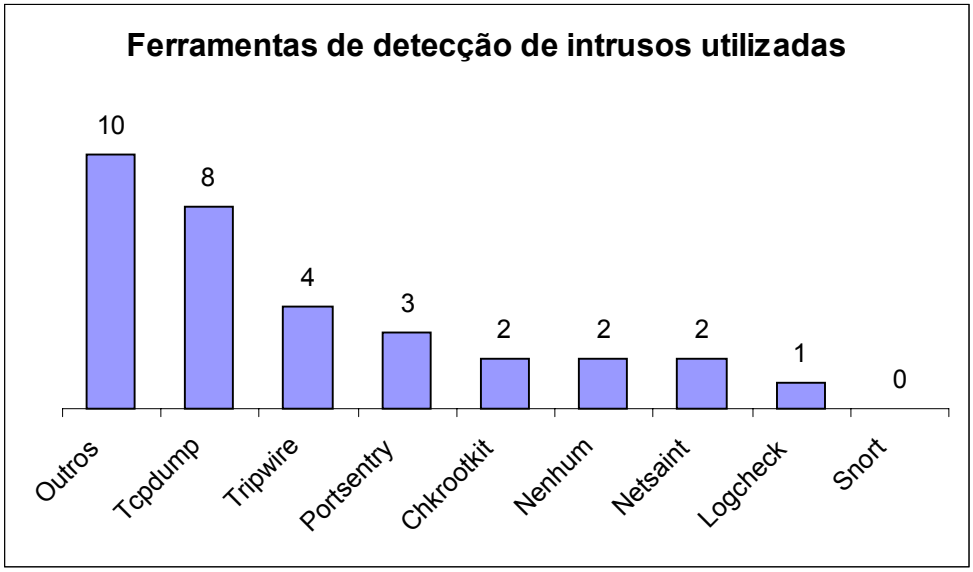

Figura 13 Qual(is) sistema(s) de detecção de intrusos estão em atividade na sua rede?

3.6.1. Análise

O alto índice de uso de outras ferramentas que não as listadas se dá pelo motivo de que existem procedimentos específicos de cada sistema operacional que servem como ferramenta de detecção de intrusos. O GNU/Linux conta com o LIDS (*Linux Intrusion Detection System*). Já o FreeBSD emite relatórios diariamente com alterações relevantes no sistema. O Novell Netware por sua vez pode ter o *border manager* configurado para fins similares.

Porém, entre as ferramentas multiplataforma, têm se o Tcpdump (TCPDUMP, 2001) como a mais usada. O Tcpdump é um utilitário que monitora o tráfego que passa nas interfaces de rede. É uma ferramenta genérica.

Seus pontos fortes são:

- Flexibilidade.
  Por exemplo: `tcpdump -i rl0 dst host 192.168.1.1 and port 23 or port 110 or port 21` para monitorar logins nas portas TELNET, POP ou FTP;
- Distribuído junto com a maioria dos sistemas operacionais POSIX;
- Fácil aprendizado e utilização. A saída pode ser direcionada para arquivo e analisada posteriormente.

E principais pontos fracos:

- Precisa ter seu comportamento ajustado pelo administrador da rede;
- Está limitado a ser um *sniffer*, a interpretação depende do administrador.

O Tripwire (TRIPWIRE, 2001) é um analisador de integridade do sistema de arquivos. Este tipo de *software* utiliza algoritmos criptográficos para criar "fotografias" do sistema de arquivos. Qualquer modificação nos arquivos, como por exemplo a instalação de um *rootkit* é facilmente detectada comparando as "fotografias" de antes e depois do ataque. Para ambientes GNU/Linux está sob licença GNU *General Public License.* No FreeBSD entretanto, caso não se deseja obter licença comercial para o Tripwire pode-se optar pelo AIDE (*Advanced Intrusion Detection Environment*) sob GPL para todos sistemas operacionais.

Portsentry é um *Host-based Intrusion Detection System* criado pela Psionic (PSIONIC, 2001) capaz de identificar varredura de portas. Manter um *software* como este faz com que o administrador de redes saiba o momento e a origem dos ataques que o Portsentry pode identificar.

Pontos fortes do Portsentry:

- Detecta *stealth scans* (GNU/Linux);
- Baseado em regras pré-definidas de comportamento, sem necessidade de intervenção do usuário.

Pontos fracos:

- Por funcionar com regras pré-definidas precisa ser atualizado para se adaptar a novas formas de ataque;

- Não detecta *stealth scans* no FreeBSD e demais sistemas operacionais.

Chkrootkit (MURILO, 2001) é um conjunto de *shell scripts* que testa o sistema contra uma variedade de *rootkits*.

Pontos fortes do chkrootkit:

- Detecta de forma rápida e eficiente os *rootkits* mais conhecidos;

- Sua utilização é fácil e pode ser inserida no agendador do sistema (vixie cron, anacron, etc...)

Pontos fracos:

- Assim como os demais *softwares*, não deve ser o único sistema de detecção instalado;

- O programa somente informa a detecção de *rootkits*. Não os remove nem provê procedimentos para remoção deles.

O netsaint (NETSAINT, 2001) é um IDS híbrido, que monitora *host* e rede utilizando-se de regras obtidas com o produto e que também podem ser programadas para fins específicos do administrador.

Pontos fortes:

- Análise em tempo real do tráfego da rede.

- Relatórios de fácil leitura e alto teor técnico são gerados e podem ser acessados por um *browser*.

Ponto fraco:

- *Setup* inicial complexo e propenso a falhas até que se entenda o funcionamento dos arquivos de configuração.

O Logcheck (PSIONIC, 2001) é, assim como o Porsentry, um dos integrantes da suíte de ferramentas de segurança Abacus Project. Este programa tem a função de analisar logs gerados pelos seguintes *softwares*:

- Suíte Abacus Project;
- TCP Wrappers;
- System Logger (SyslogD);
- *Log* Dæmon;
- TIS *Firewall Toolkit*.

É uma ferramenta de auxílio na detecção de intrusos, já que sua tarefa é fazer a análise e interpretação de arquivos-texto em lugar do administrador.

Ponto forte:

- É altamente integrado aos *softwares* para os quais foi desenhado;

Ponto fraco:

- Por ser altamente integrado, não é importante quando a configuração do *host* não inclui muitos dos aplicativos que ele suporta.

Snort (ROESCH, 2001) é um kit de detecção de intrusos quase completo. Vem pré configurado e a atualização das regras de comportamento são atualizadas a cada 30 minutos no website. Funcionando de forma parecida com os anti-vírus para *desktops* ele pesquisa por "assinaturas" conhecidas de ataque e toma as seguintes ações:

- Relata, em arquivo de *log* todas informações necessárias do ataque, entre elas: classificação, nível de risco, endereço de origem, data e referências para estudo sobre o ocorrido;
- Opcionalmente, através do módulo Guardian, escreve em tempo real uma regra que alimenta o filtro de pacotes impossibilitando a continuação do ataque.

3.7. Questão 7

A sétima questão “Qual(is) ataque(s) você já detectou em sua rede?” trata dos ataques que já foram detectados nas redes pesquisadas.

Tabela 9 Qual(is) ataque(s) você já detectou em sua rede?

Port Scan	13 ocorrências.
Força bruta	10 ocorrências.
Outros	07 ocorrências
Negação de serviço	04 ocorrências
Rootkits	03 ocorrências
Ataque contra o usuário	02 ocorrências.
Nenhum	02 ocorrência.
Mapas de DNS	01 ocorrência.

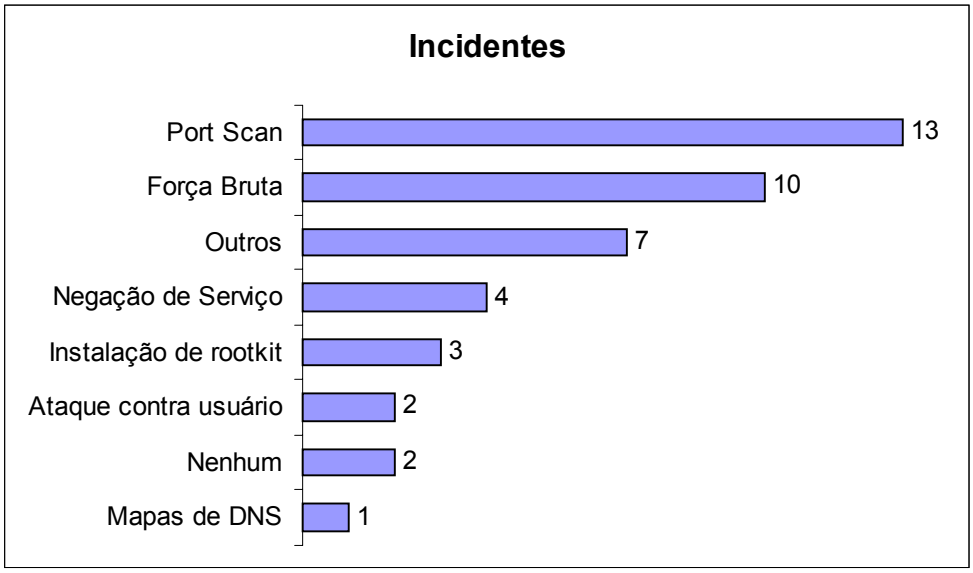

Figura 14 Quais ataques você já detectou em sua rede?

O índice relativamente alto de varreduras de portas (*port scan*) é justificado pelo fato de que ele serve como base para os demais ataques. Através de uma varredura consegue-se obter entre outras, as seguintes informações do alvo:

- Nome e versão do sistema operacional;
- Quais portas TCP e UDP estão em estado de escuta;
- Nome e versão do *software* que está atendendo cada porta em estado de escuta.

Estas informações são essenciais quando o atacante, para obter sucesso, depende de falhas na implementação dos *softwares* que estão em execução no *host*.

O ataque por força bruta é aquele onde o atacante sabe o nome de *login* da vítima e tenta seqüências de senhas – com base em dicionários ou geradas seqüencialmente – contra servidores que as solicitam, como FTP e POP. O resultado é que após horas de execução, estes programas podem acabar por formular a senha correta informando-a para seu utilizador.

A maioria dos servidores destes protocolos, ao detectar a repetição de tentativas de *login* são capazes de alertar o administrador sobre o ataque em execução, mas dependem da presença dele no sistema para efetivamente bloquear o endereço de origem do atacante.

Os ataques de negação de serviço resultam em conseqüências desagradáveis para o prestador de acesso a Internet. Geralmente seus clientes (usuários da LAN e *dial-up*) ficam com a impressão de que não há conexão nem podem medir por quanto tempo a situação persistirá.

4. MODELO PROPOSTO PARA PROTEÇÃO DE REDES

O esboço de rede aqui sugerido (Figura 15) é inspirado em um protótipo que tem mostrado resultados eficazes para VEEN (2001) e propõe a utilização de apenas um endereço IP válido para toda a rede Esta situação é muito comum para as empresas que não podem (e não precisam) obter uma classe de endereços ou fração dela. Em adição, é possível adequar este modelo a redes acadêmicas e a provedores de acesso, segmentando a rede de acordo com as necessidades.

O modelo faz uso dos seguintes recursos computacionais:

- Roteador.
- *Bridge Host.*
- *NAT Host.*
- *LOG Host.*
- As redes de servidores e clientes.

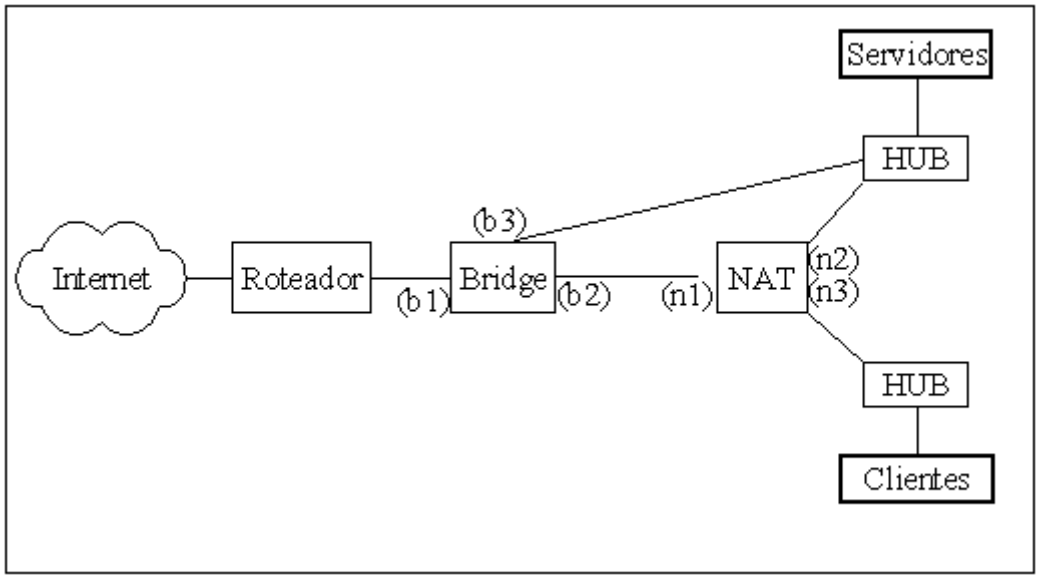

Figura 15 Visão geral do modelo proposto.

4.1. Roteador

Muitos roteadores têm implementado verdadeiros sistemas operacionais que possibilitam ao administrador de redes filtrar vários tipos de acesso antes destes entrarem em sua rede. Outros, entretanto, não dispõem de tais ferramentas, restringindo-se somente ao roteamento de pacotes sem analisar em nenhum aspecto seu conteúdo. Aqui, assume-se que o roteador faça apenas seu trabalho principal, ou seja, o roteamento. A filtragem e o tratamento de pacotes se dará no *Bridge Host*, visto a seguir.

4.2. Bridge *Host*

Este computador possui 3 interfaces de rede. A primeira (b1) e a segunda (b2) agem como *bridge.* A função primária de um *bridge* é conectar dois segmentos de rede criando a impressão de uma única e grande rede. Porém, neste caso, o *bridge* tem a função especial de filtrar todo o tráfego que passa por ele através de regras que direcionam os pacotes, bloqueando os indesejados. Tipicamente o que se vê hoje é o computador que age como *proxy* fazendo este trabalho. A desvantagem nisso é que o *firewall* é visível (acessível) na Internet. No caso aqui proposto as interfaces b1 e b2 **não recebem endereçamento IP**. Este procedimento assegura que nenhum computador - tanto na rede interna como na externa - saberá de sua existência. Portanto ele não pode ser acessado via Internet nem ser atacado no nível do protocolo IP.

A terceira interface de rede (b3) existe para uma finalidade muito importante: ela é a responsável por enviar relatos (logs) ao *LOG Host* sobre as atividades suspeitas e as tentativas de invasão para que sejam reportadas (servido como evidências) para eventual processo jurídico, bem como receber novas regras de filtragem de emergência sobre acessos indevidos detectados pelo IDS (Sistema de detecção de intrusos) que estão em cada segmento de rede. Esta interface necessita um endereço IP, porém o trabalho de *bridging* entre b1 e b2 ocorre de tal forma que elas são incapazes de trocar informações com b3.

4.3. NAT *Host* / Bastion *Host*

A tecnologia NAT (*Network Address Translator*) permite que computadores configurados com endereços não roteáveis possam enviar e receber pacotes pela Internet. Os motivos para o uso deste recurso são estes:

- Escassez de endereços válidos: Muitas vezes o custo para se alocar um endereço válido para cada computador em uma rede pode inviabilizar o acesso a Internet.

- Segurança: Considerando que maioria dos computadores faz uso da Internet para acessar recursos e não para servir conteúdo é mais seguro fornecer a estes computadores um endereço que não pode receber pedidos de conexão originados de fora da rede local.

O Modelo então requer um computador que seja capaz de assumir pedidos originados na rede interna, processa-los e devolver os resultados ao computador que fez a solicitação.

Este *host* necessita de 3 interfaces de rede. A primeira (n1) entrega todo conteúdo que recebe de n2 e n3 para a interface b2 presente no *Bridge Host*. A segunda (n2) é o *gateway* para todos os servidores da corporação. A terceira (n3), por fim, é o *gateway* de todos computadores que não servem conteúdo, ou seja, os clientes. Esta separação entre rede de servidores e rede de clientes fornece o benefício de que se uma vulnerabilidade for encontrada e explorada por um *hacker* nos serviços (DNS, SMTP, HTTP etc..) de um dos servidores, os computadores da rede de clientes ainda estarão protegidos de tal ataque.

O *bastion host* é o primeiro computador (se não o único) com presença real na Internet. Este pode ser comparado com o *hall* de um edifício comercial. As pessoas que vêem de fora não devem ter acesso às escadarias nem aos elevadores da construção, mas podem transitar livremente pelo *hall* e perguntar o que quiserem. Da mesma forma, o *bastion host* é o sistema exposto a estes elementos potencialmente hostis. É nele que todos usuários - inclusive os que agem de má fé – devem se conectar para acessar todos os outros sistemas e serviços.

O *bastion host* está altamente exposto porque sua existência é conhecida na Internet. Portanto, o mantenedor do sistema deve concentrar esforços de segurança nele, principalmente durante sua construção e fase inicial de operação. Mesmo em se falando em um único *bastion host*, é importante saber que pode haver mais de um, dependendo da configuração do *firewall*. O número depende dos requerimentos de cada *site* em particular, mas o princípio é sempre o mesmo.

Bastion hosts são usados de várias formas em várias topologias. A maioria, entretanto, é orientada à filtragem de pacotes, ao *proxying* ou a ambas. Em uma abordagem híbrida seus princípios gerais devem ser:

- Orientação à simplicidade;
- Preparação para o comprometimento do sistema.

4.3.1. Orientação à simplicidade

Quanto mais simples for um *bastion host*, mais facilmente pode-se mantê-lo seguro. Qualquer serviço que o *bastion host* ofereça pode conter *bugs* de *software* ou erros de configuração que potencialmente acarretam problemas de segurança. Portanto, deve-se ter em mente a importância em manter a simplicidade e oferecer o menor número de serviços possível neste sistema – e com os mais baixos privilégios – mantendo assim, a rede sem criar nela um gargalo.

4.3.2. Preparação para o comprometimento do sistema

Por mais que se empreenda todos os esforços para garantir a segurança do *bastion host*, pontos de quebra (ou, pontos de entrada) podem ocorrer. Subestimar o quão seguro é este sistema deve ser regra para o administrador de sistemas. Somente preparando-se para o pior, e planejando-se para este fato será possível revertê-lo. Sempre manter a questão “E se o *bastion host* for comprometido?”...

Caso o comprometimento do *bastion host* venha a ocorrer, deve-se evitar que esta quebra leve a um desastre do sistema de proteção por completo, instruindo os servidores de conteúdo à não aceitar nenhuma conexão com origem no *bastion host* exceto aquela que justifica a existência do servidor em questão. Por exemplo: negar, no filtro do servidor HTTP qualquer conexão originada no *bastion host* com destino diferente da porta 80/tcp.

4.4. O Log Host

O *Log Host* se trata de um computador, isolado fisicamente dos demais e, preferencialmente, conectado através de uma comunicação serial ou qualquer outra inalcançável por qualquer intruso. Seu papel é receber e registrar eventos que estão ocorrendo pela rede, bem como coletar provas que possam incriminar judicialmente o intruso que atacar o perímetro monitorado.

4.5. As redes de servidores e clientes

Estando protegida tanto por um filtro de pacotes invisível quanto por um *gateway* desempenhando tradução de endereços, as redes de servidores e clientes estão relativamente protegidas. As medidas tomadas neste modelo elevam ao máximo o nível de segurança entre os computadores que formam estas redes. Porém, isto não significa que a barreira seja intransponível.

Por mais que se inspecione todos os pacotes, quando o serviço parecer legítimo ele deve ser atendido. Deste ponto em diante, é possível ao atacante agir de duas formas:

- Causar negação de serviço;
- Explorar falhas de implementação dos *softwares* que realizam serviços.

4.5.1. Negação de serviço

Sendo os pacotes legítimos, eles devem ser atendidos. Porém se o *hacker* fizer muitos pedidos de conexão antes de receber confirmação (enviar SYNs antes de receber ACKs), configura-se um ataque de negação de serviço. Isto não causa propriamente uma invasão mas o serviço disponibilizado naquele computador fica inacessível para todos os outros interessados em usá-lo. Geralmente um computador só não consegue causar tal efeito, mas uma rede distribuída de atacantes consegue este intento, principalmente escolhendo como alvos grandes empresas comerciais.

Não existe solução que garanta com sucesso a proteção de uma rede contra ataques de negação de serviços. O que se pode fazer é utilizar um sistema de detecção de intrusos ativo que monitora o sistema e encontrando alguma atividade suspeita, instruindo o filtro de pacotes a não mais aceitar qualquer conexão com o endereço de origem do atacante. Neste caso, para evitar a sobrecarga de informação (*flood*) deve-se enviar uma única notificação ao administrador sobre o ocorrido e rejeitar os demais pacotes, evitando assim, entupir o *Log Host* com mensagens repetidas sobre o estado da filtragem.

4.5.2. Exploração de falhas na implementação do software servidor

Quando todos os componentes de segurança estão ativos e no máximo de sua performance, incluindo o sistema que controla e evita ataques de negação de serviço ainda pode haver falhas nos *softwares* que servem nos computadores desta rede.

Estas falhas são brechas deixadas pelos programadores da implementação de um serviço (p.e. o Sendmail é uma implementação do protocolo SMTP) e descobertas por *hackers* experientes.

A lista de discussão Bugtraq (BUGTRAQ 2001) freqüentemente emite boletins informando aos assinantes relatando descobertas de falha na implementação dos vários *softwares* utilizados nos servidores espalhados por toda Internet.

5. RECOMENDAÇÕES PARA O MODELO DE PROTEÇÃO PROPOSTO

Para o entendimento sobre a importância de cada elemento do Modelo Proposto, é apresentado aqui um relatório descrevendo os procedimentos necessários para se obter as funcionalidades desejadas.

5.1. O Sistema Operacional do Bridge Host

Para aplicação prática deste estudo, foi escolhido o FreeBSD, que é um sistema operacional BSD UNIX profissional, para computadores com processadores baseados no modelo i386, DEC-Alpha ou PC-98. É mantido pela Universidade da Califórnia, Berkley, e seus contribuidores, bem como recebe suporte de grandes empresas como a Daemon News Inc., BSD Central Inc. e Yahoo Inc. (FREEBSD, 2001).

Derivado do último *release* do BSD UNIX (versão 4.4), o FreeBSD herda a implementação do protocolo TCP/IP que deu origem a ArpaNet e posteriormente, Internet.

Entre os motivos para esta escolha, estão:

- Possui filtro-de-pacotes (*firewall*) incorporado ao *kernel* do sistema;
- O acesso a seu código fonte, além de ser didático, permite ajustes de performance e segurança;
- Tem como princípios a estabilidade, segurança e uniformidade de uso;
- Possui, historicamente, a melhor implementação do protocolo TCP/IP entre todos sistemas operacionais;
- Não há custos financeiros com licenciamento;
- Possui um dos históricos de menor número de falhas que possibilitam acesso indevido ao sistema, ficando geralmente por anos sem nenhum relato neste sentido. (BUGTRAQ, 2001)

Como a maioria dos sistemas UNIX, o FreeBSD permite:

- Compartilhamento de arquivos via NFS;

- Distribuição de informações de rede via NIS;
- *Logins* remotos via SSH;
- Monitoramento remoto, incluindo carga do processador, memória, interfaces de rede, estado de processos via SNMP;
- Código para configuração de *bridge* mais estável do que os outros sistemas operacionais.

5.2. Verificador de integridade

AIDE (*Advanced Intrusion Detection Environment)* é um verificador de integridade de sistemas construído nos moldes do *software* comercial Tripwire (LEHTI & VIROLAINEN, 2001).

O AIDE constrói um banco de dados a partir de expressões regulares existentes em seu arquivo de configuração, armazenando atributos como: permissões, *inodes* (nós no sistema de arquivos), UID (código de identificação do usuário), GID (código de identificação do grupo), tamanho do arquivo, data de modificação, data de acesso, número de *links*. Cria também um *checksum* ou *hash* de cada arquivo usando um ou uma combinação dos seguintes algoritmos: sha1, md5, rmd160, tiger (crc32, haval e gost podem ser compilados se o suporte a mhash estiver disponível).

Tipicamente, o administrador de sistemas irá criar um banco de dados AIDE no novo *host* antes deste ser colocado para funcionamento em rede. Este primeiro banco de dados é um retrato do estado normal dos arquivos protegidos e servirá como base para todas as alterações futuras no sistema. Este repositório deve conter informações sobre os binários-chave do sistema, bem como bibliotecas, *headers* e demais arquivos que devem ser preservados. Não se inclui porém, arquivos que sofrem alterações freqüentemente, como os aquivos de *log*, os diretórios de usuários etc...

Após uma invasão, geralmente o administrador examina o sistema usando ferramentas como o ls, ps, netstat e who. É comum entretanto, o invasor alterar estes binários com versões modificadas para esconder arquivos e processos que estão em execução na tentativa de ocultar seu *rootkit* do administrador. Mesmo um administrador que tenha anotado previamente as datas e tamanhos de seus binários a comparação pode não funcionar porque é possível manipular todos atributos de um arquivo e, alguns *rootkits* fazem este trabalho sem que o invasor tenha que se preocupar.

Apesar de ser possível manipular os arquivos do sistema, é muito difícil manipular um algoritmo de *checksum* como o md5 e, exponencialmente mais difícil é manipular cada um dos algoritmos suportados pelo AIDE. Executando-o após uma invasão, o administrador pode facilmente identificar as alterações feitas em seu sistema com alto grau de confiança e precisão.

5.3. O detector de intrusos

Optou-se por utilizar o Snort como detector de intrusos. Alguns motivos que levaram a esta escolha são:

- Simplicidade de uso;
- Possibilidade de se manter as assinaturas de ataques em sincronia quase simultânea com as regras atuais do *website* do autor da ferramenta;
- Alto índice de resultado nos testes informais iniciais.

6. POLÍTICA DE SEGURANÇA

Com base nas informações fornecidas por ZWICKY et al. (2001) formulou-se linhas mestras para a elaboração de um documento de política de segurança. O objetivo é encontrar o melhor modelo considerando:

- Custo da segurança;

Em ambientes conectados à Internet, não se pode assumir segurança absoluta, mesmo que se tenha a disposição recursos financeiros ilimitados.

- Funcionalidade;

As pessoas não gostam de trabalhar ou estudar em ambientes hostis, então, por um lado se perde em segurança e por outro se perde em funcionalidade. Sempre.

- Compatibilidade cultural;

Regulamentar o comportamento do usuário de acordo com o que está dentro de seus costumes é fator decisivo no sucesso da implantação da política de segurança.

- Aspectos legais.

É importante que haja conformidade entre as penalidades impostas no documento de política de segurança e a lei vigente no País.

6.1. Conteúdo

A política de segurança deve ser vista como um canal de comunicação entre usuários e administradores da rede. Precisa explicar a importância da segurança para motivar o usuário a praticá-la.

É importante incluir no documento a mensagem de que a responsabilidade por atos prejudiciais é de todos. É hostil e injusto distribuir um documento que especifica apenas as obrigações dos usuários.

A maioria das pessoas não convive com textos jurídicos nem são especialistas em computação. É preferível portanto, utilizar uma linguagem casual para formular o documento de política de segurança mesmo que ele não ganhe toda aparência oficial que se deseja.

Mais do que escrever o documento, é necessário usa-lo como regra todos os dias. Isto significa que se a política não é seguida, algo deve ser feito para consertar a situação. O profissional de segurança da informação deve ser responsável por fazer tais correções acontecerem e esta afirmação também deve estar contida no documento, assim como outras:

- Gerentes de certos serviços tem autoridade para revogar acesso de usuários subordinados a ele;
- Gerentes terão como responsabilidade avaliar transgressões ocorridas em seus setores;
- O administrador da rede, em conjunto com a diretoria pode cessar recursos que não se enquadram nos padrões da empresa.

A política deve especificar quem decide e dar indícios de quais penalidades estão previstas para cada caso descrito. Porém não deve apontar o que acontecerá em seguida com muita exatidão, já que não se tratam de sentenças de lei, mas sim de políticas.

Nenhuma política atinge a perfeição. Não é possível prever cada caso e documenta-los. Entretanto, é preciso especificar as exceções que podem ocorrer em cada processo.

Prever que a política sofrerá revisões é necessário. Nunca se pode dar por encerrada a formulação do documento. Com o passar do tempo novas necessidades precisam ser documentadas e o lançamento destas, é importante para a continuidade efetiva da política.

No momento em que se precisa aprofundar o detalhamento técnico que descreve os sistemas do qual a rede é formada, algumas questões são relevantes para o processo de formulação do documento de política de segurança. Entre elas:

- A quem é permitido ter conta no sistema?
- Existem contas temporárias para visitantes?
- Como tratar fornecedores, parceiros e clientes?

- As contas podem ser compartilhadas entre mais de um usuário?
- Secretárias e auxiliares podem receber permissão para ler correspondência eletrônica de seus superiores?
- De que forma disponibilizar informações de projeto para os parceiros da organização?
- Membros das famílias dos funcionários recebem algum privilégio no sistema?
- Como tratar os empréstimos informais de credenciais de acesso ao sistema?
- Em que circunstâncias um funcionário perde sua credencial de acesso? E quando a recebe de volta?
- Os funcionários podem servir conteúdo em seus computadores?
- Quais procedimentos os usuários devem tomar quando necessitarem ligar um computador pessoal (ou externo) na rede?
- Informações financeiras da corporação precisam tratamento especial? (criptografia, *backups* extra etc…)
- Como deve ser a formulação das senhas dos usuários e com que freqüência devem ser renovadas?
- Quais são os limites de uso da Internet? Quais as penalidades para os infratores?
- Que precauções devem ser seguidas em ordem de se evitar infecções por vírus na rede?
- Quais procedimentos devem os usuários tomar para manter seus computadores domésticos tão seguros quanto os da empresa?
- Devem existir recursos/privilégios especiais para funcionários viajantes?
- Quais pré-requisitos deve-se atender antes de se projetar *sites* de comércio eletrônico dentro da empresa?
- Quais informações são confidenciais? Como serão protegidas? Podem ser transmitidas por meios não criptografados?

Informações que não são importantes para o usuário não devem ser inclusas no documento de política de segurança. É importante enfatizar o que está se tentando proteger e porque estes procedimentos são feitos, entretanto, não se deve detalhar no nível técnico estas instruções. Em resumo, é de maior utilidade ter a política toda documentada em uma página de texto descrevendo o quê e o porquê das medidas de segurança a ter um documento formal e altamente técnico ocupando 10 folhas de papel para detalhar procedimentos que os usuários não conseguem compreender.

6.2. Planejamento

O item planejamento diz respeito às ações que devem ser tomadas antes de colocar os sistemas computacionais em funcionamento. Nem sempre isto é possível, pois na maioria das vezes a preocupação com segurança surge após os incidentes da rede já em funcionamento. Em todos os casos, deve-se avaliar a segurança como um processo cíclico revendo cada ponto crítico de tempos em tempos.

- Identificar o que é necessário proteger;
- Definir quais são as prioridades de proteção do ambiente;
- Especificar normas sobre como proceder diante de cada emergência;
- Educar os usuários da rede interna.

6.3. Usuários e senhas

Algumas práticas com relação aos usuários e senhas são muito eficientes na prevenção de incidentes. Uma prática é instruir os usuários a usar senhas formadas por combinações relativamente complexas que inviabilizem o uso de *password crackers*. Outra é estudar o funcionamento dos programas que atacam com força bruta as contas do sistema. Assim o próprio administrador faz o trabalho de tentar "quebrar" senhas detectando combinações fracas antes do atacante.

Outro ponto importante é o elo de ligação entre os setores de informática e recursos humanos. O administrador da rede deve ser sempre a primeira pessoa a saber da demissão de qualquer usuário/cliente da rede, para invalidar tentativas de destruição das informações que ele possa ter acesso.

- Certificar-se que cada usuário possui uma conta individual no sistema;
- Certificar-se que todas as contas possuem senha;
- Certificar-se que o sistema não aceita senhas mal formuladas;
- Executar *password crackers* contra o próprio sistema à procura de senhas fracas;
- Não transmitir senhas por meios de fácil captura, como telefone e *e-mail*;
- Certificar-se que as permissões do arquivo de contas de usuários não possa ser lido por ninguém além do administrador e do sistema de autenticação;
- Considerar a possibilidade de expirar as senhas dos usuários em intervalos curtos e regulares.

6.4. Contas no sistema

Em um aspecto mais interno do sistema operacional, é importante haver um tratamento especial para as contas do administrador e dos *softwares* servidores. Não se deve atribuir por exemplo a mesma conta para mais de um serviço.

- Inibir a possibilidade de *login* da conta de administrador a partir de terminais remotos;
- Remover periodicamente contas que ficaram inativas;
- Criar contas separadas para cada *software* servidor existente no *host*, evitando atribuir serviços à conta padrão *nobody:nogroup* comum em ambientes que seguem o padrão POSIX.

6.5. Configurações do sistema operacional.

Arquivos do sistema operacional que são vitais para seu funcionamento. São os principais alvos de ataque, pois neles estão todas as regras de comportamento do computador. Deve ser dada atenção máxima a este ambiente. Algumas recomendações:

- Efetuar *backups* dos arquivos de sistema e arquivos de usuário regularmente;

- Executar testes de restauração das mídias de armazenamento dos *backups*.
- Examinar o sistema de arquivos regularmente à procura de arquivos que estejam com o *bit* de *setuid*/*setgid* ligados;
- Eliminar possibilidade de gravação nos dispositivos de terminais e pseudoterminais;
- Remover *shells* desnecessárias;
- Remover utilitários que não sejam necessários mas são instalados com o sistema operacional.

6.6. Logs

Nos arquivos de *log* estão todas as atividades que o administrador precisa saber sobre o sistema. Entender o funcionamento e conteúdo de cada um deles ajuda o administrador a detectar anomalias. Recomendações

- Executar os utilitários de verificação de atividades do sistema operacional, como o "last" e "who /var/log/wtmp", regularmente;
- Estudar e entender a configuração e a saída gerada pelo servidor de logs, para saber quais atividades ele reporta e onde encontrá-las quando necessário.

6.7. Ameaças locais

A maioria das falhas relatadas por BUGTRAQ (2001) só podem ser exploradas quando se obtém acesso local ao computador alvo. Algumas recomendações são:

- Não instalar *softwares* que não forneçam código-fonte;
- Não incluir o diretório atual (".") na variável de ambiente $PATH;
- Incluir e examinar a variável de ambiente $PATH nos scripts de administração de sistema;
- Certificar-se que scripts que possuam *bit* setuid/setgid não provoquem saída para o *shell*.

6.8. Ameaças nos serviços de rede

Falhas nos serviços de rede são as que dão acesso ao sistema sem a necessidade de presença ou conta local no *host*. Apesar de serem difíceis de se explorar são as que mais servem como porta de entrada para intrusos. Recomendações:

- Desativar serviços rlogin, rsh, rexec, rcp e demais serviços "r", substituindo-os pelo *secure shell*;
- Desabilitar recursos de rede que não são necessários. Entre eles NFS, UUCP, Finger, TFTP e TELNET;
- Remover programas de teste que acompanham *softwares* servidores, como o test-cgi, printenv, phf do serviço HTTPD.

6.9. Respondendo a incidentes de segurança

O primeiro passo para responder a um incidente de segurança é decidir qual é a natureza da resposta, se houver alguma, que deve ser feita imediatamente. Questões como "O atacante obteve sucesso em seu ataque?" ou "O ataque ainda está em progresso?" são extremamente relevantes, pois se o atacante obteve sucesso então o administrador de redes se encontra realmente em uma emergência porque é necessário antes de tudo, descobrir quais os danos que ocorreram para depois saber qual serviço/recurso que deu entrada para o intruso. Se o ataque ainda está em progresso pode-se tomar decisões como desligar os equipamentos que fazem conexão com a Internet.

Uma vez determinado que se está realmente em situação de emergência e que é necessário responder, é importante iniciar a documentação de tudo que está ocorrendo. Mesmo não sendo um momento apropriado, escrever um relatório simples em papel, no formato de *log* ajuda a evitar novos problemas como este e é uma oportunidade de se obter uma compreensão sobre o fato.

Uma vez de posse do material necessário para a documentação, é hora de decidir pelo desligamento do sistema. Para avaliar esta necessidade, deve-se considerar conseqüências como:

- Perda de dados que podem ser necessários para os usuários legítimos da rede;
- Perda de dados que evidenciem o ataque;

- Impossibilidade de analisar os equipamentos porque eles estarão desligados/desconectados;

A próxima prioridade é reparar os danos. Manter a tranqüilidade nesta situação pode ser essencial para resolução de problemas, pois o administrador terá que se autenticar no sistema com privilégios máximos e novos erros por causa da tensão podem comprometer ainda mais o sistema. A presença de um colega de trabalho ou administrador de rede de outra organização pode ajudar muito.

Dependendo da natureza da organização pode ser necessário relatar o incidente ao corpo jurídico, auditores, relações públicas e departamento de segurança interno se:

- for necessário acusar judicialmente o atacante;
- houver suspeita que há colaboração de internos no incidente;
- houver suspeita de que houve acesso físico por parte do atacante.

O Anexo III descreve as atribuições do Comitê Gestor (COMITÊ GESTOR, 2001) da Internet no Brasil, para respostas legais a incidentes de segurança de redes.

CONCLUSÕES E RECOMENDAÇÕES

Os temas relacionados com infra-estrutura, procedimentos e recursos de segurança devem ser vistos com prioridade e estar em constante reavaliação dentro das corporações. Como conseqüência do cenário ao mesmo tempo amigável e hostil que a Internet oferece, algumas análises precisam ser focadas na pauta do profissional que atua com a segurança da informação, tais como:

- A estratégia de segurança adotada está alinhada às necessidades de negócio da instituição?
- A alta administração recebe atenção devida no que diz respeito à segurança das informações que armazenam em seus computadores? A cultura de segurança está disseminada entre eles?
- Estrutura, funcionalidade e orçamento dedicados à segurança estão compatíveis com a estratégia de negócios da organização? Existe medição sobre o retorno dos investimentos com segurança?
- Qual é a abrangência e profundidade com que se trata segurança nos ativos eletrônicos da corporação? Toma-se medidas que vão além da implantação de ferramentas e produtos para proteção?
- Qual o impacto que as falhas de segurança podem provocar na relação de confiança e fidelização com os clientes, parceiros e colaboradores?

O objetivo de reavaliar segurança nas organizações, imposta pela nova realidade global, não deve tirar o foco de negócios. Por outro lado, não é desejável que se tome decisões a partir de análises superficiais ou de direcionamentos extremistas e pouco flexíveis, fundamentados simplesmente por medo e insegurança.

Elaborar a política de segurança é sem dúvida um trabalho longo e maçante, exatamente o oposto do tipo de trabalho que a maioria dos técnicos apreciam. Porém, desenvolve-la e mante-la é vital para a segurança da instituição. É importante que ela não apresente textos técnicos e/ou políticos demais, para que o usuário a compreenda e que sumarize de forma simplificada todas as ações que ele precisa tomar, para cumprir sua parcela de comprometimento com a segurança da organização como um todo.

Manter *firewalls* de forma eficaz e em concordância com a política de segurança é também um trabalho de excelência. O conjunto de sistemas computacionais que filtra conteúdo deve ser um espelho daquilo que a diretoria da corporação espera do cumprimento da política de uso da Internet, bem como deve minimizar as possibilidades de ataques, invasões e vazamento de informação, mantendo a integridade e reputação da instituição.

Acredita-se, que o modelo de proteção apresentado neste trabalho, é de grande utilidade para os participantes da pesquisa. A técnica é de relativo baixo custo e *downtime* (tempo de parada necessário para ativar o novo sistema) minimizado, adicionando uma nova e importante camada de segurança para estas redes, facilitando tanto a detecção de intrusos como a invalidação de suas ações.

BIBLIOGRAFIA

ANTIHACKERS. ***Hackers* e suas práticas**. Disponível em <http://www.anti-*hackers*.com.br> Acesso em 20 out. 2000.

BERNSTEIN, T.; BHIMANI, A; SHULTZ, E. **Segurança na internet**. Rio de janeiro: Campus. 1997.

BUGTRAQ **Discussion list. List mantained by Security Focus**. Disponível em <bugtraq@securityfocus.com> Acesso em 1 set. 2001.

COMITÊ GESTOR. **Sobre o comitê gestor.** Disponível em <http://www.cg.org.br/sobre-cg/apresentacao.htm>. Acesso em 28 out. 2001.

FONTES, E. **A velha engenharia social**. Disponível em <http://www.jseg.net/segurancadainformacao72.htm>. Acesso em 4 set. 2001.

FORQUESATO, M.A. **segurança.pdf**. segurança de sistemas e internet firewall. 2001. Adobe acrobat reader 5.0.

FREEBSD. **FreeBSD Website**. Disponível em <http://www.freebsd.org>. Acesso em 28 ago. 2001.

LEHTI, R; VIROLAINEN, P. **Advanced intrusion detection environment**. Disponível em <http://www.cs.tut.fi/~rammer/aide.html> Acesso em 19 out. 2001.

LIMA, **A Engenharia social**. Disponível em <http://www.aqui.com.br/virus/20000824/virus.htm>. Acesso em 5 nov. 2000.

MEDEIROS, C.D.R. **Segurança da informação**. 2001. 75 f. Trabalho de conclusão de curso (Graduação em informática) - Departamento de informática, Universidade da região de Joinville, Joinville.

MURILO, N. apres.ppt. **Ferramentas para detecção de ataques em FreeBSD**. 2001. Microsoft power point 2000.

NBSO. **Nic BR Security Office.** Disponível em <http://www.nic.br>. Acesso em 16 dez 2001.

NETSAINT. **Netsaint documentation.** Disponível em <http://www.netsaint.org/qanda/faq.php> Acesso em 27 out 2001.

NETSEC INTERNET SECURITY. **Engenharia social**. Disponível em <http://www.netsec.com.br/tecnologia/engenharia_social.htm>. Acesso em 25 ago. 2000.

PANGEIA. **Lista de itens de segurança**. Disponível em <http://www.pangeia.com.br/listaseg.htm> Acesso em 22 out 2001.

PSIONIC. **The Abacus Project**. Disponível em <http://www.psionic.com/abacus/> Acesso em 20 out. 2001.

RIBEIRO, A.M. **Segurança é sobrevivência**. Disponível em <http://www.tba.com.br/pages/alexigor/virtual.htm> Acesso em 5 nov 2000.

ROESCH, M. **Snort user´s manual**. Disponível em <http://www.snort.org/docs/writing_rules/> Acesso em 21 out. 2001.

TCPDUMP. **Tcpdump man page.** Disponível em <http://www.freebsd.org/cgi/man.cgi?query=tcpdump&apropos=0&sektion=0&manpath=FreeBSD+4.4-RELEASE&format=html> Acesso em 28 out. 2001.

TRIPWIRE. **Tripwire Open Source.** Disponível em <http://www.tripwire.org/qanda/faq.php> Acesso em 26 out. 2001.

UNISINOS. **firewalls.pdf**. Instalação e uso de firewalls. 1998. Adobe acrobat reader 5.0.

VEEN, J. S. V. D. **A network setup with FreeBSD and OpenBSD**. Disponível em <http://www.daemonnews.org/200109/network.html> Acesso em 10 set. 2001.

ZWICKY, E.;COOPER, S.;CHAPMAN, D.B. **Building internet firewalls**. Sebastpol: O´Reilly, 2000. 869 p. ISBN 1-56592-871-7

GLOSSÁRIO

Broadcast	Transmissão simultânea para vários pontos de uma rede.
Cracking	Ato de invadir um sistema de computadores. Persistência e repetição de truques conhecidos que exploram fraquezas comuns na segurança do sistema alvo.
Criptografar	Cifrar dados de acordo com um algoritmo. Para decifar o texto criptografado necessário conhecer a chave, gerada no momento da codificação.
Hacker	Pessoa ocupada em invadir computadores ou redes.
Host	Computador hospedeiro de um ou mais softwares servidores.
Kerberos	Um esquema de autentificação desenvolvido no MIT usado para previnir a monitoração de logins e senhas.
NFS	Sistema de arquivos em rede utilizado por sistemas operacionais UNIX. Similar ao compartilhamento das redes Microsoft.
NIS	Sistema de passagem de informações. Com ele pode-se transferir, por exemplo, o banco de dados de usuários de um host à outro.
Password cracker	Todos servidores baseados em sistema Unix possuem um arquivo com as senhas dos usuários criptografadas. É virtualmente impossível decriptar estas senhas, mas é possível usar “dicionários” (lista de palavras), encriptá-las e comparar o resultado com as senhas contidas no arquivo.
Phreaking	O termo vem de "phone freak". A arte e ciência de usar a rede telefônica para, por exemplo, fazer ligações sem pagar.
Protocolo	Formato de mensagens para a transmissão de informações entre computadores.
Proxying	Arquitetura de construção lógica de rede onde o proxy server replica os pedidos originados por seus clientes e o faz em nome dele, devolvendo o resultado da conexão para quem o solicitou.
Secure Shell	Protocolo usado para gerenciamento remoto em substituição ao Telnet. Toda comunicação ocorre de forma criptografada impedido a ação de sniffers.
Setuid to root	Ação que fornece privilégios máximos para um programa em execução.
Shell	A camada mais externa de um programa que fornece uma interface para os usuários lançarem comandos. O Unix possui múltiplos shells incluíndo Bash, C Shell e Korn. Também é conhecido como interpretador de comandos.
SNMP	Protocolo utilizado para que se possa obter informações de funcionamento de um host ou sistema computacional, como carga do processador, memória e interfaces de rede.
Sniffer	Algoritmo que monitora os pacotes de dados que circulam pela rede. Mais claramente, toda informação que circula pela rede, circula através de pacotes. Um sniffer checa os pacotes em busca de informações referentes a logins e senhas.
Spoofing	Passar-se por outra pessoa. É um ataque contra a autenticação. Dois tipos de ataque por spoofing bastante conhecidos são: IP Spoofing e DNS Spoofing. Kevin Mitnick ficou famoso por utilizar este tipo de ataque.
Usuário	De um modo geral é a pessoa que acessa a algum tipo de serviço e usufrui

	dele.
Web site	Conjunto de documentos da World Wide Web hospedados em um servidor acessado por intermédio de um navegador.
WINS	Sistema de resolução de nomes-de-máquina em endereço IP e vice-versa.

ANEXOS

1. QUESTIONÁRIO

Avaliação de programas de segurança da informação.

1)Qual o sistema operacional que predomina em seu(s) hosts?

[ ] - AIX

[ ] - FreeBSD

[ ] - HP-UX

[ ] - Linux

[ ] - MacOS

[ ] - NetBSD

[ ] - Netware

[ ] - OpenBSD

[ ] - Solaris

[ ] - Windows NT/2000

[ ] - Outro

2) Sobre segurança da informação na sua rede:

(Marcar todas que se aplicam para seu caso)

[ ] - Existe uma política documentada, que estipula limites e penalidades.

[ ] - Você faz recomendações verbais e informalmais para os usuários da rede.

[ ] - Não existe nada nesse sentido.

3) Quais dos conceitos de segurança você aplica na rede que administra?

(Marcar todas que se aplicam para seu caso)

[ ] - Bridging (*Firewall* dedicado sem camada IP configurada)

[ ] - Centralizacao de autenticação: [ ]NIS [ ]LDAP [ ]Outro

[ ] - Criptografia em terminais remotos (Secure Shell, SSLTelnet...)

[ ] - DMZ (Isolamento fisico e lógico da rede de seridores)

[ ] - Filtragem de pacotes (Firewalling)

4) Sobre a filtragem de pacotes...

(Marcar 1 opção)

[ ] - Tenho um (ou mais de um) host fazendo somente filtragem de pacotes.

[ ] - Mantenho um filtro de pacotes mas agrego nele outras funções. (proxy,

nat, ftp etc...)

[ ] - Configuro a filtragem de pacotes em cada servidor que mantenho.

[ ] - Todos serviços que disponibilizamos estão em um computador e nele filtro

pacotes.

[ ] - Não faço filtragem de pacotes

5) Se você não mantém um host dedidado somente a filtrar pacotes, isto acontece

por que motivo:

(Marcar todas que se aplicam para seu caso)

[ ] - Não aumentará segurança no meu caso.

[ ] - Não consigo conciliar uma tarefa desta complexidade com meu tempo disponível.

[ ] - Não é financeiramente inviável.

[ ] - Não sei como fazer.

6) Quais dos sistemas de detecção de intrusos você mantém efetivamente em sua rede:

(Marcar todas que se aplicam para seu caso)

[ ] - AAFID.

[ ] - Aide.

[ ] - Bro.

[ ] - Chkrootkit.

[ ] - Logcheck

[ ] - Netsaint.

[ ] - Ngrep

[ ] - Portsentry.

[ ] - Snort.

[ ] - Tcpdump

[ ] - Tripwire.

[ ] - Outros.

[ ] - Nenhum.

7) Dentre os ataques listados abaixo, quais você já detectou em sua rede?

(Marcar todas que se aplicam para seu caso)

[ ] - Alteração do conteúdo web servido.

[ ] - Ataque contra usuários da sua rede.

[ ] - Negação de serviço.

[ ] - Força bruta contra alguma conta no sistema (POP, FTP etc...)

[ ] - Instalação de rootkit.

[ ] - Varredura de portas (*port scan*)

[ ] - Tentativas de obter mapas de DNS.

[ ] - Outros.

2. MANUAL DE CONFIGURAÇÃO

Bridge Man *Page*

Os sistemas POSIX documentam as principais características de seus componentes em um repositório eletrônico conhecido como *man pages*. A tradução da seção de *bridging*, de acordo com o *release* 4.4 do FreeBSD é apresentada a seguir.

Nome: Bridge – suporte a *bridging*

Sinopse: **options BRIDGE**

Descrição: FreeBSD suporta *bridging* em interfaces do tipo Ethernet.

A operação em tempo real do *bridge* é controlada por variáveis sysctl(8). A **net.linkether.bridge** pode ser ajustada para **1** para habilitar o *bridging*, ou 0 para desabilita-o.

A variável **net.link.ether.bridge_ipfw** pode ser ajustada para **1** para ativar filtragem ipfw(8) nos pacotes que passam pelo *bridge*. Note que as regras do ipfw(8) somente se aplicam a pacotes IP. Pacotes não-IP ficam sujeitos à regra padrão do filtro (número 65535) que precisa ser **allow** se pacotes ARP e outros não-IP precisam passar pelo *bridge*.

A configuração do *bridge* é controlada pela variável **net.link.ether.bridge_cfg**. Ela consiste uma lista separada por vírgulas de pares interface:*cluster*, onde todas as interfaces com o mesmo *cluster* serão ligadas no mesmo sistema de *bridge*.

Uma outra variável reinicializa o *bridge*; isto é necessário se as configurações do *bridge* incluírem interfaces carregáveis. Depois de carregar o controlador de uma nova interface, ajustar a variável **net.link.ether.bridge_refresh** para **1** irá causar uma reinicialização do *bridge*.

Exemplos:

O seguinte comando irá causar o *bridging* entre as interfaces ep0 e fxp0, e entre as interfaces fxp1 e de0.

```
sysctl -w net.link.ether.bridge_cfg ep0:0,fxp0:0,fxp1:1,de0:1
```

Bugs:

Cuidados devem ser tomados para não contruir *loops* na topologia do *bridge*. O *kernel* suporta apenas uma forma primitiva de detecção de *loops*, desabilitando interfaces quando um deles é detectado. Nenhum suporte para um dæmon executando o algoritmo *spanning tree* é atualmente provido.

Com o *bridging* ativado, as interfaces entram em modo promíscuo, causando um aumento de carga no sistema para a tarefa de receber e filtrar tráfego indesejado.

Funcionalidades estendidas para habilitar *bridging* seletivo em cluster de interfaces ainda estão sendo trabalhadas.

Interfaces que não podem ser colocadas em modo promíscuo ou que não suportam o envio de pacotes com endereços Ethernet arbitrários não são compatíveis com o *bridging*.

Ver também:

ip(4), ng_bridge(4), ipfw(8), sysctl(8)

Histórico:

Bridge bridging foi introduzida no FreeBSD 2.2.8 por Luigi Rizzo luigi@iet.unipi.pt.

3. ATRIBUIÇÕES DO COMITÊ GESTOR

O Comitê Gestor da Internet no Brasil foi criado a partir da necessidade de se coordenar e integrar todas as iniciativas de serviços Internet no País bem como assegurar a qualidade e eficiência dos serviços ofertados. Entre suas atribuições principais está a de manter um grupo de segurança de redes, o Nic BR Security Office.

Este grupo é formado por dois sub-grupos:

- *Backbone,* coordenado por Ricardo Maceira (Embratel) que discute segurança nos troncos que formam a espinha dorsal da Internet;

- Provedores, coordenado por Nelson Murilo (Pangeia) e Rubens Kuhl Jr. (UOL), cujo foco são os aspectos de segurança com os fornecedores de acesso à Internet. Este último, está relacionado diretamente com a Polícia Federal e têm com uma de suas incumbências investigar crimes que envolvem a Internet. Então, quando o incidente necessitar averiguação criminal recomenda-se relatar o acontecimento para <nbso@nic.br>.