

CENTRO UNIVERSITÁRIO DO ESTADO DO PARÁ
ÁREA DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

*Implementação de uma VPN em Linux utilizando o
protocolo IPSec*

Alan Tamer Vasques
Rafael Priante Schuber

Belém - Pará
2002

CENTRO UNIVERSITÁRIO DO ESTADO DO PARÁ
ÁREA DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

*Implementação de uma VPN em Linux utilizando o
protocolo IPSec*

Alan Tamer Vasques

Rafael Priante Schuber

Trabalho de Conclusão de Curso apresentado à Coordenação do curso de Bacharelado em Ciência da Computação do Centro Universitário do Estado do Pará - CESUPA, como parte dos requisitos à obtenção do título de bacharel em Ciência da Computação, orientado pelo professor MSc. Marcelo Hely da Silva Oliveira.

Belém - Pará

2002

CENTRO UNIVERSITÁRIO DO ESTADO DO PARÁ
ÁREA DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

*Implementação de uma VPN em Linux utilizando o
protocolo IPSec*

Alan Tamer Vasques
Rafael Priante Schuber

BANCA EXAMINADORA:

NOTA: _____

DATA: ____ / ____ / _____

Dedicamos este trabalho aos nossos pais,
Claudio Fernandez Vasques (*In memoriam*),
Maria das Graças Tamer Vasques, Vilson
João Schuber e Annunciada Maria Teresa
Priante Schuber, que nos ensinaram e
incentivaram a buscar novos conhecimentos,
de forma que sejamos sempre pessoas
honestas e dedicadas àquilo que fazemos.

Agradecemos ao professor MSc. Marcelo Hely da Silva Oliveira, por sua orientação, que foi muito útil para o desenvolvimento deste trabalho.

À professora MSc. Darcy de Nazaré Flexa Di Paolo, por sua importante revisão metodológica.

Aos professores e coordenadores do curso, que em muito contribuíram para nossa formação acadêmica.

Aos amigos acadêmicos, que ao longo de 4 anos, mostraram união, companheirismo e, principalmente, amizade uns com os outros.

Ao amigo Bruno Figueiredo Pantoja e à Marko Engenharia, que gentilmente cederam alguns equipamentos que foram utilizados na implementação deste trabalho.

Às nossas namoradas, Tayana Augusta de Carvalho Neves e Patrícia Reis Valente, pela compreensão e incentivo nos momentos necessários.

A todos, que de alguma forma, contribuíram para a concretização deste objetivo.

"Se você conhece o inimigo e conhece a si mesmo, não precisa temer o resultado de cem batalhas. Se você se conhece, mas não conhece o inimigo, para cada vitória obtida sofrerá também uma derrota. Se você não conhece nem o inimigo nem a si mesmo, perderá todas as batalhas".

Sun Tzu

RESUMO

A necessidade da troca de informações sigilosas de forma segura e com baixos custos tornou-se um problema para a maioria das empresas que possuem seus dados estruturados através de redes de computadores. O avanço e a criação de tecnologias que buscam solucionar estas questões têm sido um dos maiores desafios na área da computação. Algoritmos criptográficos, protocolos de segurança, meios de comunicação seguros, são alguns dos itens primordiais para que esta informação possa trafegar em ambientes livres de interferências externas. A VPN – Rede Privada Virtual – é uma das soluções mais viáveis presentes no atual mercado da informática. Neste trabalho, serão mostrados os principais itens desta tecnologia, implementado-a no sistema operacional Linux, utilizando o protocolo IPSec, a fim de posicioná-la como uma alternativa segura e economicamente atrativa para organizações privadas e estatais.

Palavras-Chaves: Redes de Computadores, Segurança em Redes, Criptografia, VPN, IPSec, Linux.

ABSTRACT

The necessity of secret information exchange in a safe way and with low costs has become a problem for most of the companies that possess their data structured throughout computer networks. The advance and creation of technologies that try to solve these matters have been one of the major challenges in the field of Computer Science. Cryptographic algorithms, security protocols, safe communication means, are some of the fundamental items to allow this information to traffic in environments free of external interferences. The VPN – Virtual Private Network – is one of the most viable solutions found in the computer market nowadays. At this work, the main features of this technology will be shown, implementing it in the Linux operational system, using the IPSec protocol, in order to place it as a safe and economically attractive alternative for private and governmental organizations.

Key-Words: Computer Networks, Network Security, Cryptography, VPN, IPSec, Linux.

LISTA DE ILUSTRAÇÕES

Figura 1: Exemplo de uma rede de computadores.....	4
Figura 2: Rede ponto-a-ponto.	5
Figura 3: Rede cliente-servidor.	6
Figura 4: Exemplo de uma LAN.	7
Figura 5: Exemplo de uma WAN.	7
Figura 6: Endereçamento IP.	10
Figura 7: Fluxo normal de uma informação.	12
Figura 8: Fluxo interrompido.	13
Figura 9: Fluxo interceptado.	13
Figura 10: Fluxo modificado.	14
Figura 11: Fluxo fabricado.	14
Figura 12: Exemplo de um Firewall.	15
Figura 13: <i>Gateway</i> VPN dentro de um <i>firewall</i>	17
Figura 14: <i>Gateway</i> VPN paralelo ao <i>firewall</i>	18
Figura 15: <i>Gateway</i> VPN atrás de um <i>firewall</i>	18
Figura 16: Processo de cifragem e decifragem.	19
Figura 17: Processo de criptografia simétrica.	20
Figura 18: Processo de criptografia assimétrica.	21
Figura 19: Processo de assinatura digital.	22
Figura 20: Túnel virtual entre 2 redes.	25
Figura 21: Funcionamento do transporte subjacente	26
Figura 22: Topologia <i>host-host</i>	29
Figura 23: Topologia <i>host-rede</i>	29
Figura 24: Topologia rede-rede.	30
Figura 25: Modo transporte.	32
Figura 26: Modo túnel.	33
Quadro 1: Exemplo de um SPD.	37
Quadro 2: Exemplo de um SAD.	37
Figura 27: Cenário da implementação.	40

LISTA DE ABREVIATURAS E SIGLAS

3DES – *Triple Data Encryption Standard*

AC – *Autoridade Certificadora*

AES – *Advanced Encryption Standard*

AH – *Authentication Header*

ARPA – *Advanced Research Project Agency*

CAST – *Carlisle Adams and Stafford Tavares*

DES – *Data Encryption Standard*

DNS – *Domain Name System*

DoS – *Denial of Service*

ESP – *Encapsulating Security Payload*

FTP – *File Transfer Protocol*

GRE – *Generic Routing Encapsulation*

GRUB – *Grand Unified Bootloader*

ICMP – *Internet Control Message Protocol*

IETF – *Internet Engineering Task Force*

IKE – *Internet Key Exchange Protocol*

IPSec – *Internet Protocol Security*

IPX / SPX – *Interwork Packet Exchange / Sequenced Packet Exchange*

ISAKMP – *Internet Security Association and Key Management Protocol*

ITU-T – *International Telecommunication Union-Telecommunication Standardization Sector*

KLIPS – *Kernel IPSec Support*

L2F – *Layer Two Forwarding*

L2TP – *Layer Two Tunneling Protocol*

LAN – *Local Area Network*

LILO – *Linux Loader*

MAN – *Metropolitan Area Network*

MD-5 – *Message Digest 5*

MIT – *Massachusetts Institute of Technology*

MPPE – *Microsoft Point-to-Point Encryption*

MS-CHAP – *Microsoft-Challenge Handshake Authentication Protocol*

NASA – *National Aeronautics and Space Administration*

NAT – *Network Address Translation*

NetBEUI – *NetBIOS Extended User Interface*

NIST – *National Institute of Standards and Technologies*

NSF – *National Science Foundation*

PFS – *Perfect Forward Secrecy*

PPP – *Point-to-Point Protocol*

PPTP – *Point-to-Point Tunneling Protocol*

RNP – *Rede Nacional de Pesquisa*

RPM – *Red Hat Packet Manager*

RSA – *Rivest Shamir Adleman*

SWAN – *Secure Wide Area Network*

SA – *Security Association*

SAD – *Security Association Database*

SHA-1 – *Secure Hash Algorithm 1*

SHA-2 – *Secure Hash Algorithm 2*

SMTP – *Simple Mail Transfer Protocol*

SPD – *Security Policy Database*

SPI – *Security Parameters Index*

SRI – *Stanford Research Institute*

TCP/IP – *Transmission Control Protocol / Internet Protocol*

UCLA – *University of California Los Angeles*

UCSB – *University of California Santa Barbara*

UDP – *User Datagram Protocol*

VPN – *Virtual Private Network*

WAN – *Wide Area Network*

SUMÁRIO

1. INTRODUÇÃO	1
1.1. OBJETIVOS.....	2
1.1.1. Objetivo Geral	2
1.1.2. Objetivos Específicos	2
1.2. MATERIAIS E MÉTODOS	2
1.3. ORGANIZAÇÃO DO TEXTO	3
2. REDES DE COMPUTADORES.....	4
2.1. TIPOS DE REDE	5
2.1.1. Ponto-a-ponto	5
2.1.2. Cliente-servidor.....	5
2.2. CLASSIFICAÇÕES	6
2.2.1. LAN.....	6
2.2.2. WAN.....	7
2.3. PROTOCOLOS	8
2.3.1. TCP/IP	8
2.3.1.1. <i>Histórico da Internet</i>	<i>8</i>
2.3.1.2. <i>A Arquitetura TCP/IP de quatro camadas.....</i>	<i>9</i>
2.3.1.3. <i>O protocolo IP.....</i>	<i>9</i>
3. ASPECTOS DE SEGURANÇA.....	11
3.1. AMEAÇAS	11
3.2. ATAQUES	12
3.2.1. Interrupção.....	12
3.2.2. Interceptação	13
3.2.3. Modificação.....	14
3.2.4. Fabricação.....	14
3.3. MÉTODOS DE DEFESAS	15
3.3.1. Firewall.....	15
3.3.1.1. <i>Firewall e VPN.....</i>	<i>16</i>
3.3.2. Criptografia.....	19
3.3.2.1. <i>Criptografia simétrica.....</i>	<i>19</i>
3.3.2.2. <i>Criptografia assimétrica</i>	<i>20</i>
3.3.2.3. <i>Função hash.....</i>	<i>21</i>
3.3.2.4. <i>Assinatura digital.....</i>	<i>22</i>
3.3.3. Certificado digital	23

4. CONCEITOS DE VPN.....	24
4.1. ELEMENTOS DE UMA VPN	24
4.1.1. Tunelamento	24
4.1.2. Autenticação das extremidades.....	25
4.1.3. Transporte Subjacente.....	25
4.2. VANTAGENS E DESVANTAGENS	26
4.3. COMPARAÇÃO COM OUTRAS TECNOLOGIAS.....	27
4.3.1. VPN x Linhas Dedicadas	27
4.3.2. VPN x Servidor de Acesso Remoto	28
4.4. TOPOLOGIAS.....	28
4.4.1. Host-host.....	28
4.4.2. Host-Rede	29
4.4.3. Rede-Rede.....	29
4.5. PPTP	30
4.6. L2TP	31
4.7. IPSec	31
4.7.1. Componentes.....	32
4.7.1.1. Modo Transporte.....	32
4.7.1.2. Modo Túnel.....	33
4.7.1.3. SA (Security Association).....	33
4.7.1.4. SPD (Security Policy Database)	34
4.7.1.5. Gerenciamento de Chaves	34
4.7.1.6. Manipulação de Pacote de Dados	35
4.7.2. Arquitetura IPSec	36
4.7.3. Vantagens e Desvantagens.....	38
5. IMPLEMENTAÇÃO DE UMA VPN EM LINUX	39
5.1. CENÁRIO	40
5.2. FreeS/WAN.....	41
5.2.1. KLIPS	42
5.2.2. Pluto.....	42
5.3. IMPLEMENTAÇÃO	42
5.3.1. Instalação do FreeS/WAN.....	44
5.3.2. Configuração do FreeS/WAN.....	49
5.3.2.1. Encriptação Oportunista	53
5.4. CONSIDERAÇÕES SOBRE A IMPLEMENTAÇÃO.....	54
6. CONCLUSÃO	55
REFERÊNCIAS	58

1. INTRODUÇÃO

Segundo Torres (2001, p.5), “As redes de computadores surgiram da necessidade da troca de informações, onde é possível ter acesso a um dado que está fisicamente localizado distante de você”.

Com o avanço da informática, é comum que uma empresa possua uma ou várias filiais com suas próprias redes de computadores. Entretanto, na maior parte dos casos, estas não possuem qualquer ligação uma com as outras, principalmente pelo fator financeiro e, em alguns casos, devido a grande distância existente entre elas. Apesar disso, a troca de informações sigilosas entre elas é essencial e, conseqüentemente, diversos métodos são desenvolvidos com o intuito de interligar tais redes de forma segura.

A *Virtual Private Network* (VPN) ou Rede Privada Virtual é uma das formas de se unir diferentes redes de uma empresa, onde se utiliza para isso um meio público para o tráfego dos dados entre elas, que geralmente é a Internet. Sua principal característica é criar “túneis virtuais” de comunicação entre essas redes, de forma que os dados trafeguem criptografados por estes túneis, aumentando a segurança na transmissão e recepção dos dados (CYCLADES, 2000).

De acordo com Cyclades (2000, p.167) “A principal motivação para a implementação de VPNs é financeira [...]”, ou seja, isto tem despertado interesse cada vez maior de empresas com diversas filiais, que necessitam de uma forma econômica de comunicação, a adotarem esta tecnologia.

Além disso, uma VPN possui a capacidade de expandir a quantidade de computadores que podem ter acesso a esta rede, sem investir em infra-estrutura extra e permite suporte a usuários móveis, sem a utilização de bancos de *modems* ou servidores de acesso remoto, ajudando a aumentar a flexibilidade e diminuir os gastos com equipamentos extras. (SCRIMGER et al, 2002).

1.1. OBJETIVOS

1.1.1. **Objetivo Geral**

O objetivo geral deste trabalho é implementar uma VPN em Linux, utilizando o *Internet Protocol Security* (IPSec), visando analisar a funcionalidade e a segurança desta tecnologia na interligação de redes de computadores.

1.1.2. **Objetivos Específicos**

Para atingir o objetivo geral estabelecido, este trabalho teve como objetivos específicos:

- Analisar a tecnologia VPN, assim como o IPSec e os seus respectivos conceitos;
- Configurar uma VPN em Linux utilizando o protocolo IPSec;
- Verificar a funcionalidade da VPN configurada, no sentido de avaliar aspectos de segurança na interligação de duas redes;
- Aprimorar os conhecimentos nestas tecnologias.

1.2. MATERIAIS E MÉTODOS

Para a realização desta monografia, foi realizada uma ampla pesquisa bibliográfica, além de diversas consultas em *sites* especializados no assunto, a fim de dar subsídios suficientes à implementação da tecnologia em estudo.

Na fase de implementação, além do sistema operacional *Red Hat*¹, o aplicativo *FreeS/WAN*² foi necessário para tal tarefa, servindo para a criação do túnel VPN. Este aplicativo é gratuito e implementa os protocolos do IPSec, conforme será abordado no Capítulo 5.

¹ (RED..., 2002).

² (FREESWAN, 2002).

1.3. ORGANIZAÇÃO DO TEXTO

Este capítulo trata da introdução do trabalho, onde são evidenciados os objetivos geral e específicos e a metodologia utilizada.

O capítulo 2 introduzirá os conceitos fundamentais de redes de computadores, como tipos de redes, classificações e protocolos, destacando o *Transmission Control Protocol / Internet Protocol* (TCP/IP).

O capítulo 3 conterà informações relevantes a respeito da segurança em VPNs, onde serão abordados os principais tipos de ataques e defesas, as ameaças existentes, os métodos de criptografia, assinaturas e certificados digitais, entre outras.

O capítulo 4 abordará os conceitos de uma VPN, onde serão observadas as suas principais características, as vantagens e desvantagens de sua implementação, além dos principais protocolos, onde será dado ênfase ao IPsec.

O capítulo 5 mostrará como a VPN foi implementada em caráter experimental, explicando cada uma de suas fases, detalhando aspectos da configuração desta tecnologia.

Por fim, o capítulo 6 tece as conclusões deste trabalho e apresenta encaminhamentos para trabalhos futuros.

2. REDES DE COMPUTADORES

Para Tanenbaum (1997), rede de computadores é um conjunto de computadores autônomos interconectados, trocando informações entre si, através de um fio de cobre, fibras ópticas, rádio, microondas, satélites de comunicação, entre outros, conforme a Figura 1, composta por cinco computadores, uma impressora e dois *hubs*³.

Um dos objetivos da criação das redes foi compartilhar recursos aos usuários, como aplicações, equipamentos e dados, independente da localização física deste recurso ou do próprio usuário.

A confiabilidade (várias fontes alternativas de fornecimento do mesmo dado), a escalabilidade (possibilidade de aumentar os recursos à medida que for necessário) e a economia de custos (troca de computadores de grande porte – *mainframes* – por computadores pessoais) foram itens fundamentais para o desenvolvimento das redes.

Um exemplo de rede mundialmente difundida é a Internet, que possui milhares de computadores interconectados trocando as mais diversas informações, tais como e-mail, arquivos, páginas pessoais e corporativas.



Figura 1: Exemplo de uma rede de computadores.

Fonte: 3COM CORPORATION, 2000.

³ Dispositivo responsável por concentrar as conexões dos equipamentos de uma rede.

2.1. TIPOS DE REDE

Levando-se em consideração o compartilhamento dos dados, existem dois tipos básicos de rede: ponto-a-ponto e cliente-servidor.

2.1.1. *Ponto-a-ponto*

Estas pequenas redes, também conhecidas como descentralizadas, geralmente com até 15 computadores, ora utilizam recursos de outras máquinas, ora fornecem os seus próprios recursos para as demais, como arquivos e periféricos. Um exemplo de rede ponto-a-ponto está na Figura 2, composta por 3 computadores, compartilhando arquivos entre si.

Elas são de fácil instalação e configuração, porém uma grande desvantagem está relacionada à segurança da rede, devido não ser possível criar uma eficiente política de segurança com este tipo de rede.

Por não haver centralização nas informações, outra desvantagem é quando dois ou mais computadores possuem arquivos com os mesmos nomes podendo causar eventuais problemas de organização no ambiente de trabalho.



Figura 2: Rede ponto-a-ponto.

Fonte: 3COM CORPORATION, 2000.

2.1.2. *Cliente-servidor*

Este tipo de rede, também conhecido como centralizada, tem como característica possuir pelo menos um computador dedicado a uma ou várias tarefas, sendo este chamado de servidor, de acordo com a Figura 3. Os exemplos mais

comuns de servidores são: arquivos, impressão, aplicações e comunicações (correio eletrônico e páginas *web*).

Devido a existência de máquinas dedicadas a determinadas tarefas, o desempenho da rede será mais eficiente, sendo a sua configuração e manutenção feitas de forma centralizada, reduzindo os custos administrativos, porém, a instalação necessita de especialistas.

Uma das grandes vantagens do tipo cliente-servidor está relacionado à segurança das informações, pois ficam centralizadas e os acessos a estas são controlados.

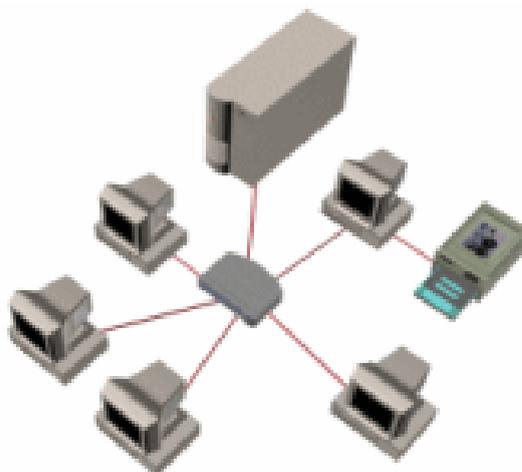


Figura 3: Rede cliente-servidor.

Fonte: 3COM CORPORATION, 2000.

2.2. CLASSIFICAÇÕES

As redes possuem diversos tamanhos, entretanto as classificações mais comuns são: *Local Area Network* (LAN), *Wide Area Network* (WAN) e *Metropolitan Area Network* (MAN), porém, serão abordadas apenas as duas primeiras, por serem mais utilizadas.

2.2.1. LAN

São redes locais que conectam servidores, estações, terminais e outros dispositivos em um ou mais edifícios, capazes de transmitir dados em alta

velocidade, com baixa taxa de erros de transmissão e cobre uma área relativamente pequena (até 2,5 Km em prédios separados). A Figura 4 mostra uma rede local composta por dois computadores, uma impressora e um *hub*.

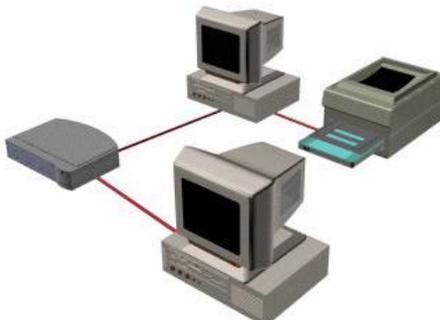


Figura 4: Exemplo de uma LAN.

Fonte: 3COM CORPORATION, 2000.

2.2.2. WAN

São redes remotas formadas pela ligação de sistemas de computadores localizados em regiões fisicamente distantes, segundo a Figura 5. As WANs geralmente utilizam linhas de transmissão de dados oferecidas por empresas de telecomunicações.

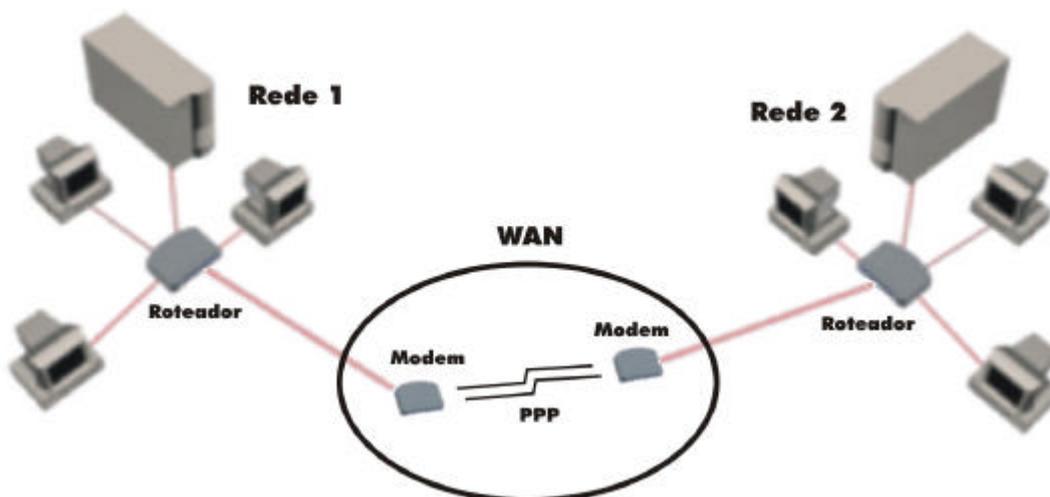


Figura 5: Exemplo de uma WAN.

Fonte: Adaptada de CYCLADES, 2000, p. 65.

2.3. PROTOCOLOS

Para que os dispositivos de redes possam se comunicar é necessário que eles falem a mesma “língua”, ou seja, utilizem o mesmo protocolo de comunicação. Em função disso, vários protocolos foram criados e são utilizados para transmissão de dados em uma rede. Dentre os mais comuns, podemos citar o TCP/IP, o *NetBIOS Extended User Interface* (NetBEUI) e o *Interwork Packet Exchange / Sequenced Packet Exchange* (IPX/SPX).

2.3.1. TCP/IP

2.3.1.1. Histórico da Internet

Apesar da atual popularização da Internet, ela teve início no final da década de 50, nos Estados Unidos, no auge da Guerra Fria, com o objetivo de proteger e transferir informações vitais na eventualidade de um ataque nuclear por parte da antiga União Soviética.

Em 1957, o Departamento de Defesa dos Estados Unidos criou a *Advanced Research Project Agency* (ARPA), que, no início da década de 60, criou a ARPANET, conectando vários centros de pesquisa: *University of California Los Angeles* (UCLA), *Stanford Research Institute* (SRI), *University of California Santa Bárbara* (UCSB) e *Utah University*.

A primeira mensagem trocada entre as redes partiu da UCLA para o SRI em outubro de 1969. Em 1971, já haviam 23 computadores conectados, entre eles o *Massachusetts Institute of Technology* (MIT) e a *National Aeronautics and Space Administration* (NASA). Em 1973, a ARPANET já estava presente em Londres, na *University College* e na Noruega, no *Royal Radar Establishment*.

Em virtude da rápida expansão da ARPANET, o protocolo de comunicação utilizado já não estava mais suprindo as necessidades desta rede. Foi então que, após intensa atividade de pesquisa na década de 70, surgiu o conjunto de protocolos que até hoje é a base da Internet, conhecido como TCP/IP.

Em 1985, a *National Science Foundation* (NSF) interligou os seus supercomputadores, formando a NSFNET, que, no ano seguinte, interligou-se à ARPANET. A união dessas redes passou a ser conhecida oficialmente como Internet.

Em 1988, finalmente a Internet chegou ao Brasil, através das comunidades acadêmicas de São Paulo e do Rio de Janeiro. No ano de 1989, foi criada a Rede Nacional de Pesquisa (RNP), uma instituição com o objetivo de coordenar os serviços de acesso à Internet no Brasil.

2.3.1.2. *A Arquitetura TCP/IP de quatro camadas*

O TCP/IP é um conjunto de protocolos de interconexão de sistemas, executado em ambiente aberto, utilizando uma arquitetura de quatro camadas, definidas como:

- **Camada de Aplicação:** fornece a interface do usuário de rede na forma de aplicativos e serviços de rede. Exemplos de protocolo: *Simple Mail Transfer Protocol* (SMTP), *File Transfer Protocol* (FTP) e Telnet;
- **Camada de Transporte:** responsável por organizar as mensagens recebidas de camadas mais altas nos segmentos, por controlar os erros e por controlar o fluxo de fim-a-fim. Exemplos de protocolo: TCP e *User Datagram Protocol* (UDP);
- **Camada de Rede:** responsável pelo endereçamento dos equipamentos e pela transmissão (ou roteamento) dos dados em redes diferentes. Exemplos de protocolo: IP e *Internet Control Message Protocol* (ICMP);
- **Camada de Interface de rede:** responsável por controlar o fluxo de dados e organizar os bits da camada física.

2.3.1.3. *O protocolo IP*

Este protocolo, que atualmente está na versão 4 (IPv4), é um dos mais importantes da arquitetura TCP/IP e é utilizado para rotear os pacotes, para que

estes possam chegar com rapidez ao seu destino e endereçar cada equipamento de rede, de modo que eles possuam um único número capaz de identificá-los. Este número é composto por 32 bits, representado por quatro campos de números decimais inteiros, que variam de 0 a 255, conforme a Figura 6.

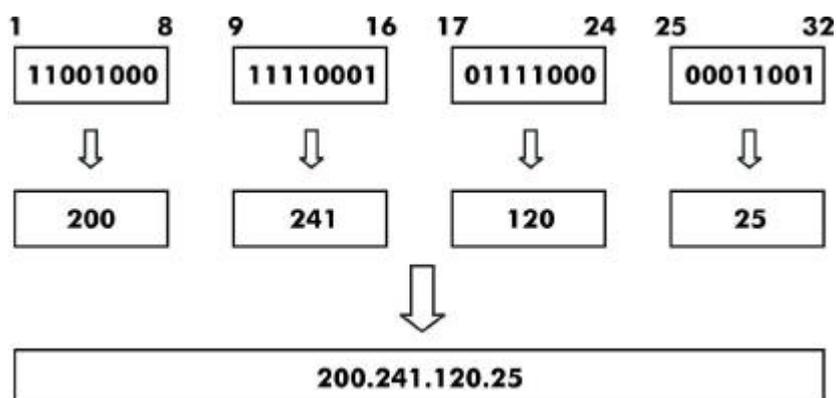


Figura 6: Endereçamento IP.

Fonte: CYCLADES, 2000, p. 68.

Este capítulo apresentou os principais conceitos de redes de computadores, como os tipos de redes existentes, as suas classificações e o protocolo mais utilizado, o TCP/IP.

Estes conceitos servirão como base para um melhor entendimento do funcionamento de uma VPN, pois como o próprio nome já diz, esta tecnologia nada mais é do que uma rede virtual. No capítulo seguinte, serão abordados os principais aspectos relativos à segurança de uma VPN.

3. ASPECTOS DE SEGURANÇA

Cada vez mais as empresas possuem informações sigilosas disponíveis em seus computadores, fazendo com que certos cuidados sejam necessários, a fim de protegê-las, como limitar o acesso físico e lógico aos computadores, através de mecanismos de segurança.

Essa preocupação torna-se ainda maior com a popularização da Internet nas empresas, onde o risco das informações serem acessadas ou alteradas é grande, pois qualquer pessoa conectada à Internet pode vir a tomar posse destas informações, caso elas não estejam bem protegidas.

Segundo Monteiro (2002), as principais funções de segurança são:

- **Autenticidade** – Verifica se a pessoa com quem está se trocando informações sigilosas é realmente quem deveria ser;
- **Confidencialidade** – Limita o acesso a informações, geralmente através do uso de criptografia;
- **Integridade** – Assegura que os dados não serão alterados durante uma transmissão;
- **Controle de acesso** – Limita o acesso e a utilização de recursos apenas a pessoas autorizadas;
- **Disponibilidade** – Mantém os recursos disponíveis, mesmo em caso de ataques;
- **Não-repúdio** – Impede que uma entidade (computador, pessoa, etc.) envolvida em uma transação negue a sua participação no evento.

3.1. AMEAÇAS

Para tornar uma rede mais segura e confiável, deve-se estar atento às principais ameaças que podem comprometer a integridade das informações de uma empresa.

Ao contrário do que se pensa, nem sempre o principal inimigo está fora da rede, como um *hacker*⁴ ou *cracker*⁵, mas sim dentro dela, como um funcionário mal intencionado, que geralmente possui livre acesso aos recursos disponíveis.

As ameaças são classificadas em dois tipos: Passivas, onde o sistema continua a operação sem a percepção de ter um invasor na rede e, geralmente, acontece roubo de informações; e Ativas, onde o invasor prejudica o sistema, atingindo os dados ou degradando os serviços (BATISTA, 2002).

3.2. ATAQUES

Para saber como proteger as informações sigilosas de uma empresa é necessário entender primeiro como os ataques se classificam e quais os tipos mais freqüentes, para que se possa utilizar o método correto de defesa em cada situação.

Baseados na Figura 7, os ataques se classificam em:

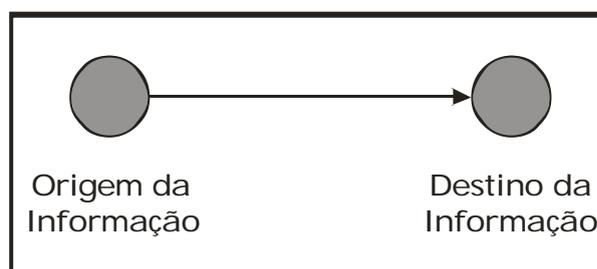


Figura 7: Fluxo normal de uma informação.

Fonte: D'ÁVILLA, 2002, p. 2.

3.2.1. Interrupção

Visa interromper o serviço oferecido, ou seja, ataca-se a disponibilidade das informações, conforme a Figura 8.

⁴ Indivíduo hábil em enganar os mecanismos de segurança de sistemas de computação.

⁵ Mesmo que *hacker*, mas com o intuito de roubar, alterar ou apagar as informações da rede invadida.

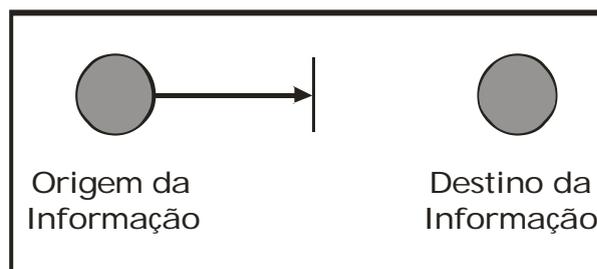


Figura 8: Fluxo interrompido.

Fonte: D'ÁVILLA, 2002, p. 2.

O principal tipo de ataque classificado como interrupção é o *Denial of Service* (DoS), que é o envio de requisições em massa para um determinado computador, de modo que o mesmo não consegue responder todas elas, ficando sobrecarregado, fazendo com que o serviço pare de funcionar.

3.2.2. *Interceptação*

Tem como objetivo capturar o que está sendo transmitido sem que o sistema perceba, ou seja, ataca-se a confidencialidade das informações, de acordo com a Figura 9.

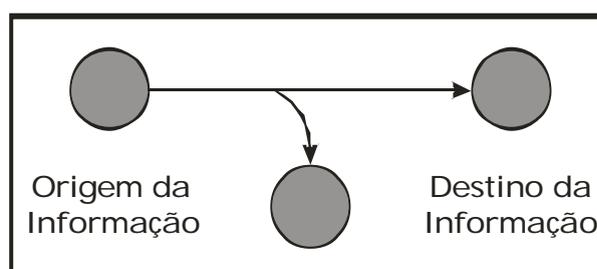


Figura 9: Fluxo interceptado.

Fonte: D'ÁVILLA, 2002, p. 2.

Um dos principais tipos de ataque desta classificação é o *man-in-the-middle*, onde o invasor simula ser o parceiro de ambas as partes envolvidas na conexão, assumindo a identidade de um usuário válido.

3.2.3. *Modificação*

É quando há alteração da informação que está sendo transmitida, ou seja, ataca-se a integridade da mesma, segundo a Figura 10.

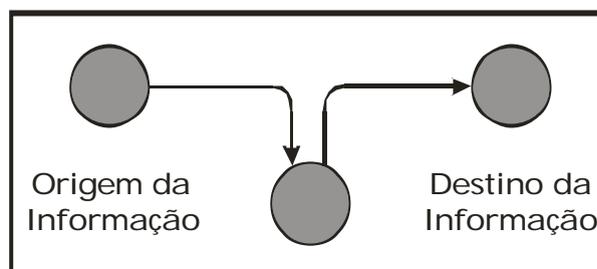


Figura 10: Fluxo modificado.

Fonte: D'ÁVILLA, 2002, p. 2.

Um exemplo de ataque desta classificação é o Replay, onde parte de uma transmissão da rede é copiada e reproduzida posteriormente, simulando um usuário autorizado.

3.2.4. *Fabricação*

O atacante tem como finalidade se passar por um usuário do sistema, a fim de obter informações para transmitir dados na rede, ou seja, ataca-se a autenticidade das informações, conforme a Figura 11.

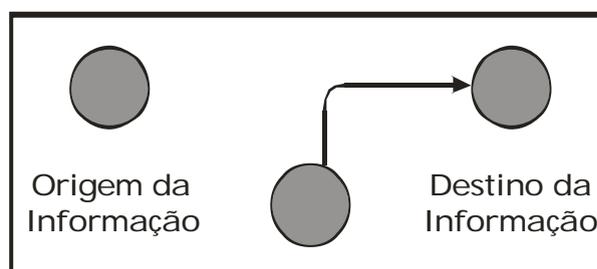


Figura 11: Fluxo fabricado.

Fonte: D'ÁVILLA, 2002, p. 2.

O tipo de ataque mais comum de fabricação é o *IP Spoofing*, que consiste na substituição do endereço IP do computador do invasor, fazendo com que ele se passe por um computador confiável da rede, podendo assim obter privilégios na comunicação.

3.3. MÉTODOS DE DEFESAS

De acordo com Scrimger et al (2002, p. 507) “Há dois aspectos da segurança de uma rede – proteger o acesso aos dados e proteger a transmissão dos dados”. Neste contexto, a autenticação dos usuários e a criptografia das informações destacam-se, servindo como base para a segurança de uma VPN.

Existem outros métodos de defesas disponíveis para os mais variados tipos de ataque. Entretanto, o objetivo deste trabalho limita-se à explicação daqueles que são úteis à implementação de uma VPN, os quais serão explicados a seguir.

3.3.1. Firewall

É uma combinação de hardware e software, cujo objetivo é controlar o trânsito de informações entre as redes privadas e a Internet, como se houvesse uma barreira entre ambas, de modo que os acessos não autorizados sejam impedidos, conforme mostra a Figura 12.

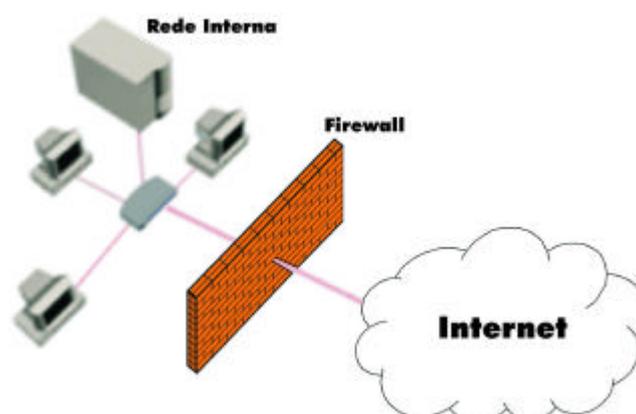


Figura 12: Exemplo de um Firewall.

Fonte: Adaptada de SCRIMGER et al, 2002, p.255.

Os *firewalls* podem ser de três tipos: filtros de pacotes, inspeção de pacotes com informações de estado e aplicativos de *firewalls* e de *proxy*⁶.

O filtro de pacotes é o tipo mais comum de *firewall* e tem como objetivo permitir ou negar a entrada de um determinado pacote de informações em uma rede, levando em consideração o endereço IP ou a porta de origem e de destino. Possui como vantagens ser mais barato e rápido que os outros tipos de *firewall*, uma vez que ele não se importa com o conteúdo dos pacotes. Entretanto, por fazer apenas uma filtragem superficial, sua principal desvantagem é ser mais inseguro que os demais.

A inspeção de pacotes com informações de estado, além de desempenhar as funções do filtro de pacotes, inspecionam o estado da conexão, ou seja, apenas aquelas conexões previamente estabelecidas e válidas, que cumprem as condições configuradas pelo *firewall*, têm acesso à rede. Uma de suas vantagens é não ter a necessidade de configurar cada computador dentro da rede, reduzindo os encargos administrativos. Todavia, suas configurações são complicadas e não fornecem autenticação de usuário.

Os aplicativos de *firewall* e de *proxy* são os mais complexos, pois varrem todos os dados que passam por eles, descartando os perigosos ou não autorizados e nunca deixando um computador de dentro da rede ficar exposto à redes externas. Possui como vantagens oferecer a maior segurança dos três tipos de *firewalls*, além de autenticar as atividades dos usuários. Devido ser mais complexo, ele também é mais lento e mais caro que os demais.

3.3.1.1. *Firewall e VPN*

Uma VPN permite interligar duas ou mais redes de computadores, provendo um mecanismo seguro de comunicação, conforme será abordado no capítulo seguinte, sendo uma ferramenta muito útil para proteger os dados de uma empresa. Entretanto, ela não deve ser única na rede, pois quanto maior a segurança empregada, maior a dificuldade para acessar os dados. Com isso, o *firewall*, que já está mais presente nas empresas e é uma tecnologia mais consolidada, torna-se

⁶ *Software* utilizado para permitir o acesso de uma rede à Internet, geralmente através de um *firewall*.

uma boa alternativa, fazendo a interação com o *gateway*⁷ VPN. Porém, é necessário um planejamento cuidadoso antes de escolher a melhor opção, que podem ser três: *gateway* VPN dentro, paralelo ou atrás de um *firewall*.

O *gateway* VPN dentro de um *firewall* é a opção que parece mais natural, pois toda a segurança de sua rede é feita em apenas um computador, de acordo com a Figura 13. Todavia, o mesmo deve ser extremamente seguro, uma vez que qualquer configuração imprópria nas regras do *firewall* pode permitir que o tráfego vindo da Internet penetre a sua rede utilizando endereços de VPN, sem ser percebido.

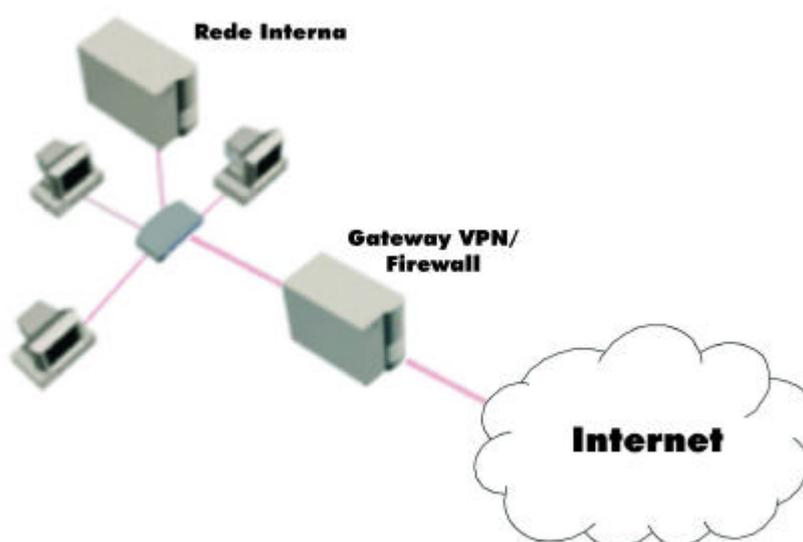


Figura 13: *Gateway* VPN dentro de um *firewall*.

Fonte: Adaptada de KOLENISKOV e HATCH, 2002, p. 45.

O *gateway* VPN paralelo ao *firewall*, por sua vez, separa o tráfego de uma VPN do tráfego da Internet em dois ou mais computadores, segundo a Figura 14. Uma das vantagens em separá-los é que, devido o fluxo das informações transmitidas à VPN não passar pelo *firewall*, nenhuma configuração extra é necessária no mesmo para permitir a passagem de pacotes VPN. Entretanto, isso faz com que o *gateway* VPN torne-se vulnerável a ataques externos, caso os mesmos não estejam bem seguros.

⁷ Computador responsável pela ligação entre duas redes.

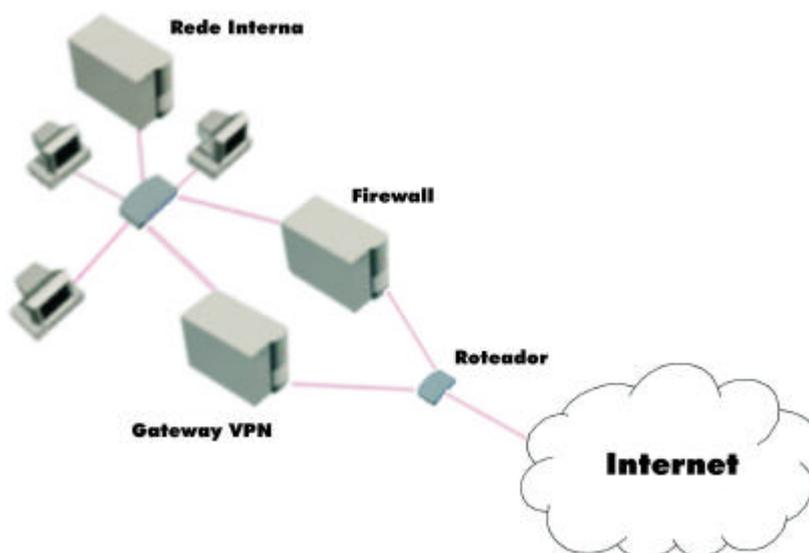


Figura 14: Gateway VPN paralelo ao *firewall*.

Fonte: Adaptada de KOLENISKOV e HATCH, 2002, p.46.

O *gateway* VPN atrás de um *firewall* faz com que todo o tráfego vindo da Internet passe primeiramente pelo *firewall*, conforme a Figura 15, deixando a rede mais segura. Em função disso, o mesmo deve possuir uma rota específica para redirecionar o tráfego da VPN. Esta topologia possui como uma de suas desvantagens a queda de performance na comunicação, em razão de todo o conteúdo de uma VPN passar antes pelo *firewall*.

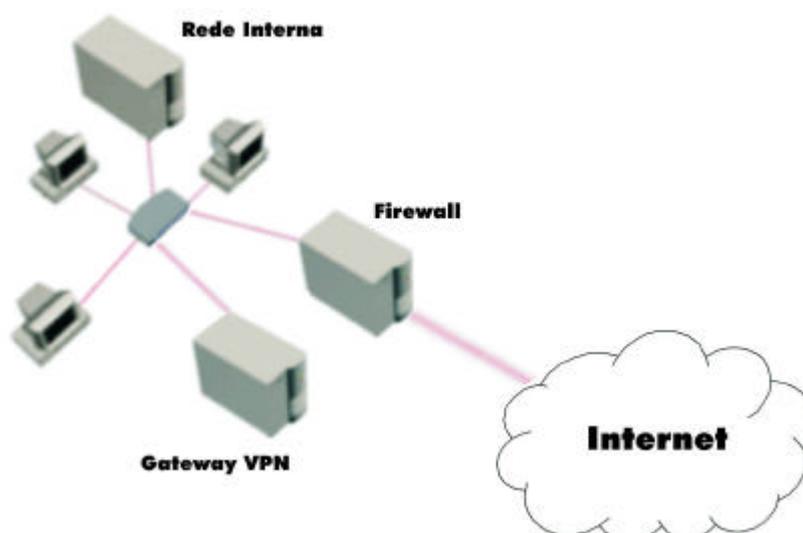


Figura 15: Gateway VPN atrás de um *firewall*.

Fonte: Adaptada de KOLENISKOV e HATCH, 2002, p.46.

3.3.2. Criptografia

A palavra criptografia vem do grego (*Kryptos* = escondido, oculto e *Grafia* = Escrita) e pode ser definida como a arte ou ciência de garantir a segurança de mensagens, de forma que apenas pessoas autorizadas a leiam. Ela garante confidencialidade, autenticidade, integridade e não-repúdio (SCHNEIER, 1996).

O processo de criptografia pode ser descrito da seguinte forma: um emissor gera a mensagem original, chamada texto plano e, utilizando uma chave e um algoritmo de cifragem, gera um texto cifrado, ou seja, incompreensível para quem não tem autorização de lê-lo. Ao chegar ao receptor, este texto passa pelo processo inverso, chamado decifragem, resultando no texto plano original, observado na Figura 16.

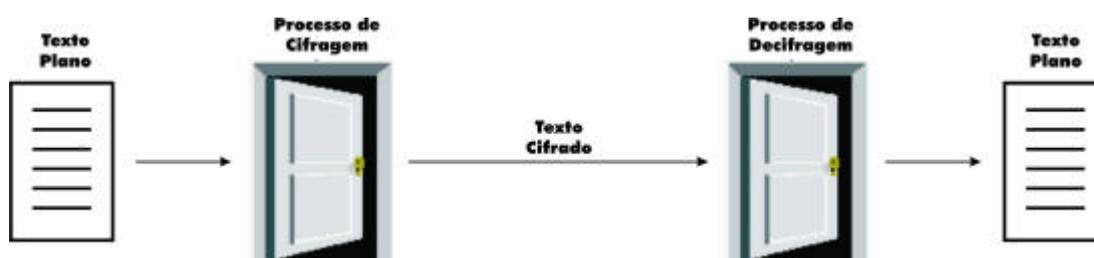


Figura 16: Processo de cifragem e decifragem.

Fonte: Adaptada de BROCARD, 2001, p. 20.

Uma chave é uma combinação de bits e, quanto maior for esta combinação, maior será a segurança obtida. Dependendo do tipo de chave utilizada, a criptografia classifica-se em: Criptografia Simétrica e Criptografia Assimétrica.

3.3.2.1. Criptografia simétrica

A criptografia simétrica, como o nome sugere, baseia-se na simetria das chaves, ou seja, a mesma chave utilizada para criptografar será usada para descriptografar a mensagem, de acordo com a Figura 17. Essa chave, chamada privada, é previamente trocada entre o emissor e o receptor por um canal de comunicação seguro.

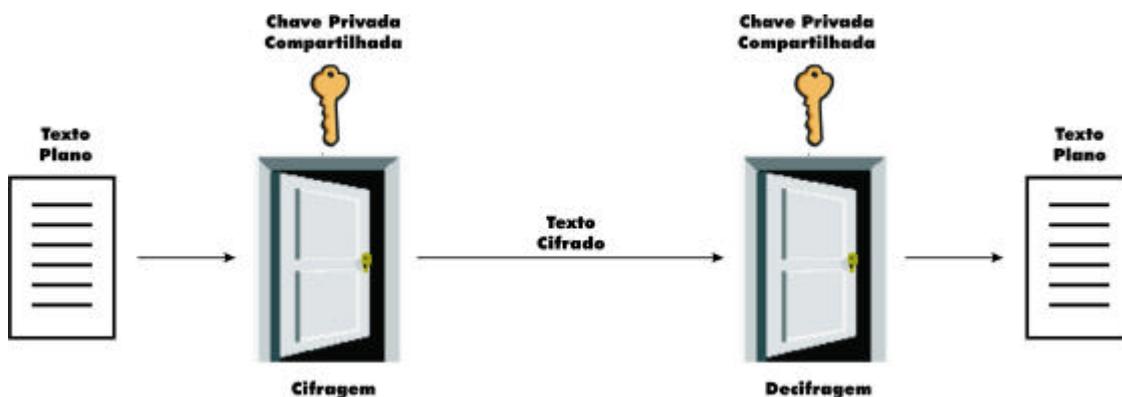


Figura 17: Processo de criptografia simétrica.

Fonte: Adaptada de BROCARDO, 2001, p. 20.

As desvantagens deste processo devem-se ao fato de que como apenas uma chave é utilizada para cada par de pessoas, a segurança em cima dela deve ser rígida e, se o número de pessoas que queiram comunicar-se de forma segura for muito grande, serão necessárias inúmeras chaves, o que dificultará ainda mais a gerência das mesmas.

Pode-se citar os seguintes algoritmos simétricos e os respectivos tamanhos das chaves geradas por cada um deles:

- **Data Encryption Standard (DES)** – 56 bits.
- **Triple Data Encryption Standard (3DES)** – 112 bits.
- **Advanced Encryption Standard (AES)** – 128, 192 ou 256 bits.
- **Blowfish** – até 448 bits.
- **Carlisle Adams and Stafford Tavares (CAST)** – 128 ou 256 bits.
- **Twofish** – 128, 192 ou 256 bits.
- **Serpent** – 128, 192 ou 256 bits.

3.3.2.2. Criptografia assimétrica

A criptografia assimétrica, por sua vez, envolve o uso de duas chaves distintas, uma privada e outra pública. Pode-se utilizar qualquer uma das chaves para cifrar a mensagem. Entretanto, somente a chave inversa deve ser utilizada para decifrá-la. Por exemplo, se um emissor utiliza a chave pública do receptor para cifrar

a mensagem, o receptor deve utilizar a sua chave privada para decifrá-la, conforme a Figura 18. Dessa forma, garante-se autenticidade e confidencialidade.

Contudo, se o emissor utilizar a chave privada para cifrar a mensagem, qualquer pessoa poderá decifrá-la, uma vez que a chave pública é de conhecimento de todos, o que garantirá apenas a autenticidade da mensagem.

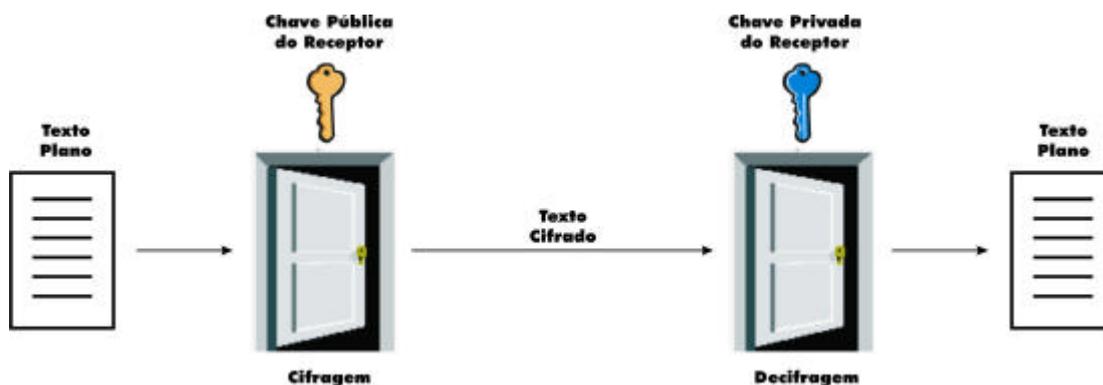


Figura 18: Processo de criptografia assimétrica.

Fonte: Adaptada de BROCARD, 2001, p. 22.

Como exemplo de algoritmo assimétrico, pode-se citar o *Rivest Shamir Adleman* (RSA), composto por chaves de 512, 768, 1024 ou 2048 bits. Este algoritmo é a base da maioria das aplicações de criptografia assimétrica utilizadas atualmente, pois seus mecanismos dificultam a obtenção da chave utilizada.

3.3.2.3. Função hash

Dada uma mensagem original, a função *hash* tem como objetivo produzir um número, conhecido como resumo, que representa de forma única esta mensagem. Uma propriedade desta função diz que o caminho inverso deverá ser computacionalmente inviável, ou seja, não poderá ser possível obter uma mensagem original através de um resumo, o que garante a integridade da mesma.

Os algoritmos que implementam a função *hash* têm como objetivo fazer com que o resumo sofra uma grande modificação caso algum caractere do conteúdo da mensagem seja alterado. Dentre os principais, podemos destacar o *Message Digest 5* (MD-5) e o *Secure Hash Algorithm 1* (SHA-1). A principal diferença entre os dois é que o primeiro é mais rápido, pois retorna um resumo de 128 bits, enquanto que o segundo retorna um número de 160 bits e, por isso, é mais seguro. Em 01 de

Janeiro de 2003, o *Secure Hash Algorithm 2* (SHA-2) será o mais novo algoritmo *hash* a ser padronizado pelo *National Institute of Standards and Technology* (NIST), tendo como principal característica o retorno de um resumo de 256, 384 ou 512 bits.

3.3.2.4. Assinatura digital

Uma assinatura digital é um código binário baseado em 2 aspectos: o documento em si e alguma informação que o ligue a uma certa pessoa ou conjunto de pessoas. Essa ligação é denominada autenticação.

Para Albuquerque (2001, p. 225) “Autenticar é confirmar a identidade de uma entidade com o objetivo de se garantir que a entidade é quem ela diz ser”.

As técnicas mais comuns de autenticação utilizam informações sigilosas (como senhas), meios físicos (como cartões magnéticos), verificação de informações biométricas (como impressões digitais) ou através da combinação delas. Um exemplo desta combinação está na utilização de um cartão de crédito, onde se utiliza, além do cartão, uma senha para efetuar a transação.

A assinatura digital faz uso da criptografia assimétrica, pois utiliza um par de chaves, uma privada e outra pública. A chave privada serve para assinar o documento, enquanto que a pública serve para verificar a assinatura.

O processo de geração da assinatura utiliza a função *hash* para a obtenção do resumo do documento, que, em seguida, cifra-o com a chave privada do emissor e envia-o ao receptor. Este utilizará a chave pública do emissor para decifrar a mensagem e a função *hash* para recalculer o resumo do documento, comparando-o com o resumo recebido, segundo a Figura 19. Isto garante a integridade do documento.

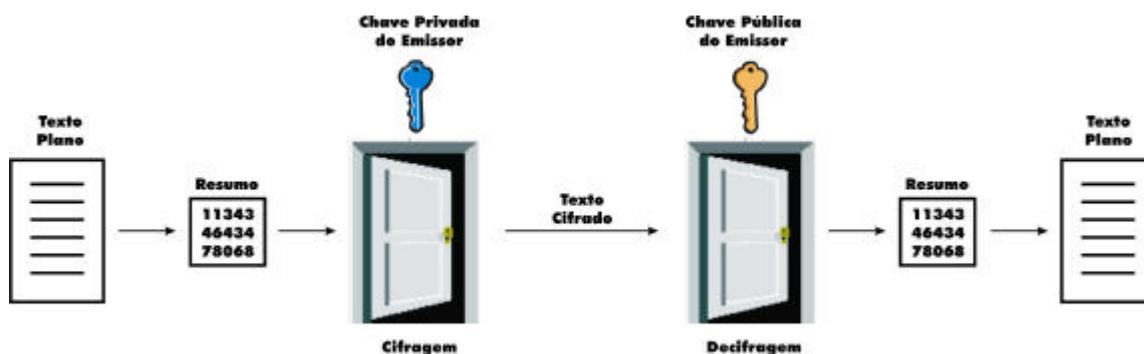


Figura 19: Processo de assinatura digital.

3.3.3. Certificado digital

O certificado digital é um arquivo assinado eletronicamente por uma entidade confiável, chamada Autoridade Certificadora (AC). Um certificado tem o objetivo de associar a chave pública a uma pessoa ou entidade, servindo, assim, como um mecanismo para a divulgação da chave pública.

Qualquer entidade que conheça a chave pública da AC pode examinar o conteúdo e confirmar a autenticidade de um certificado emitido por esta autoridade, uma vez que a AC assina os certificados com a sua chave privada.

Dentre os dados de um certificado digital, as seguintes informações estão presentes: chave pública do usuário, número de série do certificado, nome da AC que emitiu o certificado, a assinatura digital da AC, entre outras.

A recomendação mais aceita e utilizada para a produção de certificados digitais é a X.509v3, formulada pela *International Telecommunication Union – Telecommunication Standardization Sector* (ITU-T).

Este capítulo teve a finalidade de introduzir conceitos básicos de segurança em VPNs, citando as principais ameaças, os ataques mais comuns, além de mostrar os mecanismos de defesas mais utilizados, como os *firewalls*, criptografia, assinaturas e certificados digitais. É de suma importância o entendimento destes conceitos, pois são eles que garantem a segurança de uma VPN.

O próximo capítulo abordará os principais conceitos de uma VPN, explicando cada uma de suas características, citando alguns protocolos utilizados, dando ênfase ao IPsec.

4. CONCEITOS DE VPN

Conforme mencionado anteriormente, VPN significa, em português, Redes Privadas Virtuais. Desmembrando este termo, podemos destacar que **Redes** é a infra-estrutura pela qual os computadores se comunicam; **Privadas**, devido estas redes utilizarem recursos de criptografia para garantir a segurança das informações trafegadas pelo meio de comunicação e; **Virtuais**, por elas estarem fisicamente separadas.

Em outras palavras, VPNs são redes de computadores que estão separadas fisicamente e, que através de um meio público de comunicação – geralmente a Internet – comunicam-se de forma segura, através da utilização de criptografia.

Alguns dos dispositivos que implementam uma VPN são: roteadores⁸, equipamentos específicos e softwares instalados em *gateways*.

As VPNs possuem seus próprios protocolos de comunicação que atuam em conjunto com o TCP/IP, fazendo com que o túnel virtual seja estabelecido e os dados trafeguem criptografados. Dentre eles, podemos destacar o *Point-to-Point Tunneling Protocol* (PPTP), o *Layer Two Tunneling Protocol* (L2TP) e o IPsec, que serão abordados posteriormente neste capítulo, além de protocolos proprietários.

4.1. ELEMENTOS DE UMA VPN

Uma VPN tem como principais elementos: a criação de um túnel virtual encriptado (tunelamento), a autenticação das extremidades e o transporte subjacente.

4.1.1. *Tunelamento*

As informações são trafegadas de forma encriptada, dando a idéia da criação de um túnel virtual, como mostra a Figura 20, onde os dados que estiverem

⁸ Equipamento de encaminhamento de mensagens, utilizado para interligar redes locais ou redes remotas.

trafegando pelo mesmo permanecem ininteligíveis para quem não fizer parte dele. Isto garante que, se a informação for capturada, será muito difícil entendê-la, a menos que se descubra a chave utilizada.

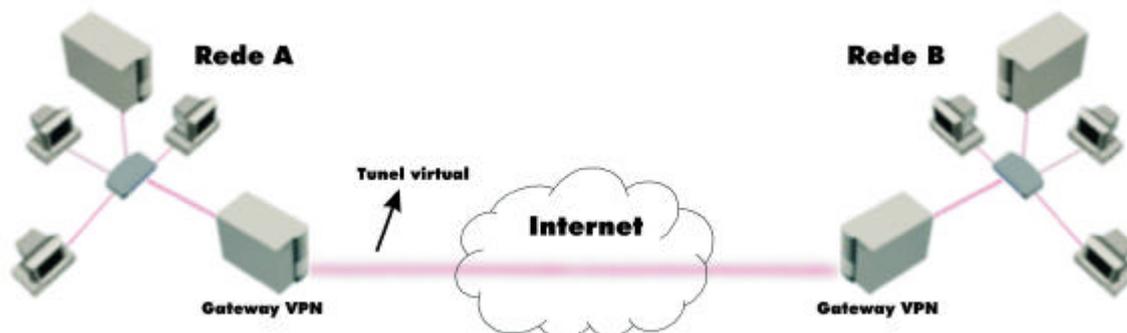


Figura 20: Túnel virtual entre 2 redes.

Fonte: Adaptada de CYCLADES, 2002, p. 118.

4.1.2. Autenticação das extremidades

As mensagens são autenticadas para assegurar que elas vieram de usuários válidos, através da utilização de protocolos de autenticação, que geralmente implementam algoritmos *hash*. Desta forma, se alguma parte da mensagem for alterada durante a transmissão, o pacote é descartado.

Mesmo a mensagem estando encriptada, a razão de se autenticá-la deve-se ao fato da prevenção de ataques do tipo Replay.

4.1.3. Transporte Subjacente

Devido o protocolo TCP/IP ser a base da Internet, ele é amplamente utilizado para a comunicação entre redes. Entretanto, ele é muito inseguro, devido não ter sido projetado para esta finalidade. Por isso, uma VPN utiliza a infra-estrutura de rede já existente do TCP/IP para transmitir os seus pacotes pela Internet, apenas adicionando alguns cabeçalhos, conforme a Figura 21. Isto faz com que os dispositivos VPN utilizem o mecanismo de transporte subjacente para se comunicarem, o que possibilita a instalação destes em qualquer parte da rede, reduzindo-se os custos (KOLENISKOV e HATCH, 2002).

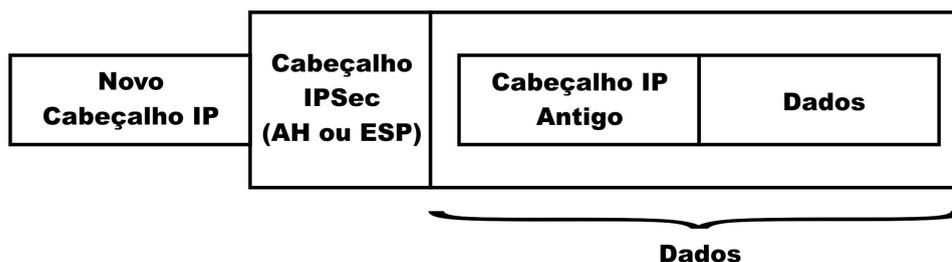


Figura 21: Funcionamento do transporte subjacente. Esta figura mostra um pacote IPSec, que é composto por um pacote IP original, utilizado para transmitir informações pela Internet, e alguns dos cabeçalhos utilizados pelo IPSec.

4.2. VANTAGENS E DESVANTAGENS

De acordo com Kolesnikov e Hatch (2002), as vantagens em utilizar uma VPN estão relacionadas à segurança, transparência, facilidade de administração e redução de custos.

A VPN garante o fornecimento de funções vitais de segurança, como autenticidade, confidencialidade, integridade e controle de acesso, reduzindo os riscos de ataques externos, como *IP Spoofing*, *man-in-the-middle* e injeção de pacotes na comunicação.

A transparência não deixa que os usuários, as aplicações e os computadores percebam a localização física dos recursos que estão sendo utilizados, permitindo que eles sejam acessados em lugares remotos como se estivessem presentes localmente, facilitando o gerenciamento das redes e diminuindo a necessidade de treinamentos para os administradores.

A redução de custos é uma das maiores vantagens de se implementar uma VPN, pois usando conexão local de Internet, não é necessário, por exemplo, o uso de linhas dedicadas e servidores para acesso remoto, que são relativamente mais caros de se manter comparando-se a uma VPN.

Apesar de todas as vantagens citadas anteriormente, uma VPN apresenta como desvantagens o fato de sua implementação poder consumir muito tempo, a dificuldade na localização de seus defeitos, a relação de confiança entre as redes interconectadas e a disponibilidade da Internet.

A implementação de uma VPN pode consumir bastante tempo e tornar-se uma grande desvantagem se não houver um planejamento adequado, preocupando-

se com a gerência das chaves e a resolução dos problemas encontrados. É importante que se tenha conhecimento de como as redes que se pretende interligar estão funcionando, assim como as suas configurações, pois qualquer imperfeição pode resultar em mais tempo gasto para corrigi-la.

Em razão dos dados trafegarem de forma encriptada em uma VPN, a localização de defeitos, como a não sincronização das chaves, falhas de autenticação, pacotes perdidos e a sobrecarga do *gateway* VPN, pode ser um problema.

A relação de confiança entre as redes é uma necessidade e deve ser bem planejada, pois os recursos compartilhados por uma das redes ficarão acessíveis à outra. Isso significa que, se uma das redes não possuir uma segurança adequada, ela está vulnerável a ataques externos e, conseqüentemente, toda a VPN também estará.

Em razão de uma VPN depender da Internet para conectar suas redes, é necessário que ela esteja sempre disponível, o que nem sempre é possível, devido às falhas existentes nos provedores de serviços de Internet.

4.3. COMPARAÇÃO COM OUTRAS TECNOLOGIAS

Uma forma de analisar o comportamento de uma VPN é comparando-a diretamente com outras tecnologias, como linhas dedicadas e servidor de acesso remoto.

4.3.1. VPN x Linhas Dedicadas

Quando se fala em conexão entre redes à longa distância para a troca e a atualização de informações entre banco de dados diariamente, pensa-se primeiramente em Linhas Dedicadas. Um dos benefícios desta utilização está na disponibilidade da conexão, pois nesta topologia os equipamentos que permitem as comunicações são controlados pela própria empresa.

No entanto, os custos de uma implantação de linhas dedicadas poderão quebrar qualquer orçamento. Por isso, utilizar uma solução VPN reduzirá os gastos

com equipamentos, administradores de redes e de sistemas em futuras migrações e expansões destas informações.

4.3.2. VPN x Servidor de Acesso Remoto

Nas empresas, cada vez mais se procura flexibilizar o trabalho de seus funcionários, fazendo com que eles possam exercer suas funções em suas casas, principalmente quando há grande distância entre a empresa e as residências, ou quando estes estão em viagens de negócio.

Tradicionalmente, existem duas opções disponíveis para solucionar estas questões colocadas acima: abrir os recursos da rede interna da empresa para a Internet ou manter um banco de *modems* para permitir o acesso remoto dos usuários.

A primeira opção causa enormes riscos quanto à segurança, pois se os recursos ficarem comprometidos, informações secretas da empresa também ficarão, assim como os servidores da mesma, o que poderá gerar perdas financeiras.

A outra opção eleva bastante os custos, pois são necessários servidores dedicados, placas multiseriais, linhas telefônicas exclusivas, ligações interurbanas, caso o funcionário esteja em outra cidade, além de aumentar os gastos com a administração do banco de *modems*.

4.4. TOPOLOGIAS

4.4.1. Host-host

Um *host* é um computador que possui acesso à Internet. Esta topologia tem como finalidade comunicar dois *hosts* separados fisicamente, fornecendo a segurança necessária para a troca de informações via Internet. Estes *hosts* podem ou não estar presentes em uma rede. A Figura 22 mostra dois *hosts* do departamento de recursos humanos, de filias de uma mesma empresa, sincronizando suas informações. Neste caso, o objetivo não era interligar as redes

inteiras, o que exigiria um esforço maior e desnecessário, além de acarretar em custos extras.

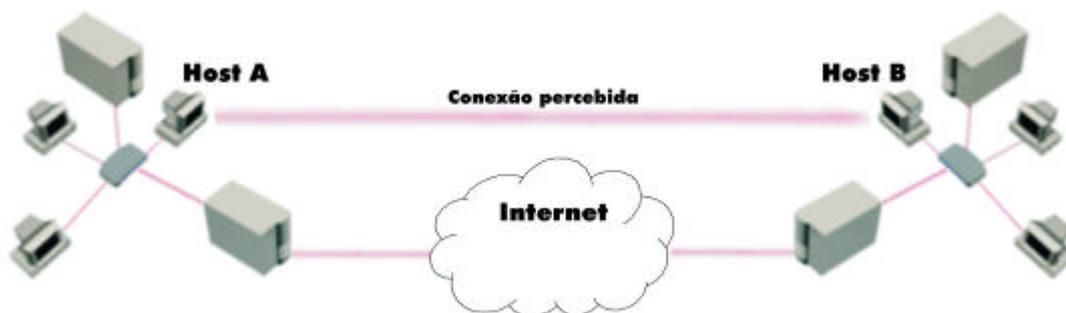


Figura 22: Topologia *host-host*.

Fonte: Adaptada de KOLENISKOV e HATCH, 2002, p.15.

4.4.2. Host-Rede

Também chamada de acesso remoto VPN, esta topologia permite a conexão de um *host* móvel a uma determinada rede através da Internet, de acordo com a Figura 23. Para isso, basta o usuário ter um software para conexões remotas VPN instalado neste *host*. Esta opção é bastante atrativa para funcionários que constantemente viajam ou estão fora da empresa e precisam manter de alguma forma o contato.

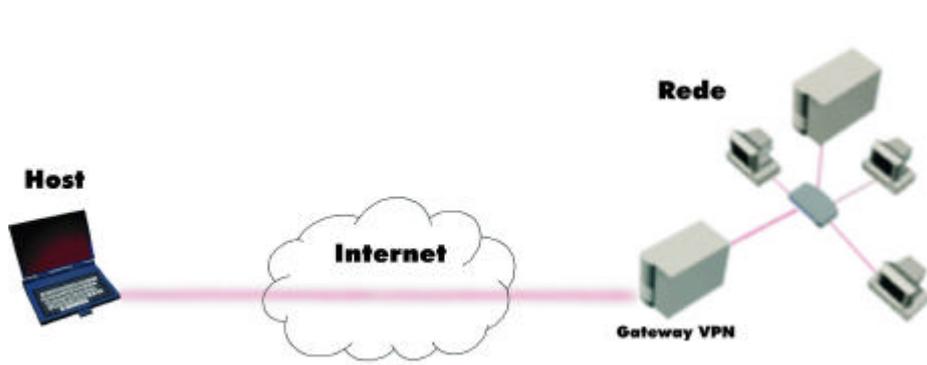


Figura 23: Topologia *host-rede*.

Fonte: Adaptada de KOLENISKOV e HATCH, 2002, p.16.

4.4.3. Rede-Rede

Este tipo de configuração possui um *gateway* VPN nas extremidades de duas ou mais redes fazendo a conexão segura entre elas, segundo a Figura 24. Esta topologia é ideal para redes de uma mesma empresa geograficamente distantes

entre si. Ela também pode ser utilizada para compartilhar recursos específicos com parceiros de negócios, como fornecedores, principais consumidores e outras companhias.

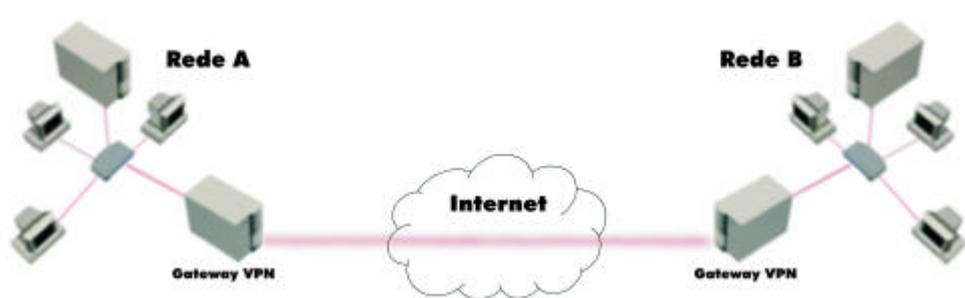


Figura 24: Topologia rede-rede.

Fonte: Adaptada de KOLENISKOV e HATCH, 2002, p.17.

4.5. PPTP

Este protocolo foi desenvolvido pelo Fórum PPTP, um consórcio que inclui *US Robotics, Microsoft, 3Com, Ascend e ECI Telematics*, porém a sua implementação mais conhecida é a da *Microsoft*, que é amplamente utilizada em sistemas *Windows*.

Ele utiliza o *Point-to-point Protocol* (PPP) para fazer as conexões e, em seguida, encapsula os dados através do *Generic Routing Encapsulation* (GRE) e os envia à outra extremidade da VPN.

Para autenticação, o protocolo PPTP utiliza o *Microsoft-Challenge Handshake Authentication Protocol* (MS-CHAP) e para criptografia, o *Microsoft Point-to-point Encryption* (MPPE).

Existem algumas vantagens em se implementar uma VPN utilizando o protocolo PPTP, como, por exemplo, o suporte a outros protocolos diferentes do IP, como o NetBEUI e o IPX. Contudo, uma de suas principais desvantagens é relativa à sua segurança, pois este protocolo fornece suas chaves de encriptação utilizando a senha do usuário como base, ou seja, se esta senha for fraca, como palavras encontradas em dicionários ou números de telefones, a chave também será.

4.6. L2TP

Projetado pela *Cisco Systems* e, posteriormente, homologado pela *Internet Engineering Task Force* (IETF) como protocolo padrão, baseia-se no *Layer Two Forwarding* (L2F) para solucionar os problemas do PPTP, sendo considerado o seu herdeiro. Algumas características, como a utilização do PPP para fornecer o acesso remoto e a operação em ambientes como o NetBEUI e o IPX, são mantidas do PPTP.

No entanto, uma diferença visível em relação a seu predecessor é quanto à forma de autenticação, pois ela é feita em dois níveis. No primeiro, o usuário é autenticado pelo provedor de acesso antes do túnel ser instalado e, no segundo, quando a conexão é estabelecida entre os *gateways*.

Sendo um protocolo padrão, qualquer fabricante pode criar produtos que utilizem o L2TP, de forma que provedores de acesso e consumidores em geral não dependam de produtos fornecidos por uma única empresa.

Apesar de ser atual, o L2TP apresenta como desvantagem não possuir um mecanismo eficiente de encapsulamento, ou seja, para executar esta tarefa, ele necessita do protocolo IPSec, que será explicado posteriormente, para que juntos possam garantir a segurança da VPN.

4.7. IPSec

Segundo Kolesnikov e Hatch (2002, p.133), “IPSec é um conjunto de protocolos desenvolvido para proteger o tráfego dos pacotes IP”. Sua principal característica é garantir a segurança da comunicação. Para isto, ele utiliza os conceitos de autenticação, proteção à integridade, proteção a ataques do tipo Replay e confidencialidade, todos dependendo do conjunto de protocolos a serem utilizados.

O PSec foi desenvolvido pelo Grupo de Trabalho de Segurança do IP da IETF com o intuito de ser o protocolo padrão de endereçamento para a próxima versão do IP, chamada IPv6, que está em fase de testes em diversas instituições no

mundo inteiro. No Brasil, um dos órgãos que está testando o IPv6 é a RNP. Por enquanto, o IPSec é apenas opcional para o IPv4.

Por ser um protocolo padrão, existem diversas implementações disponíveis para ele, tanto em roteadores, quanto em *gateways*.

4.7.1. Componentes

O funcionamento do IPSec depende muito dos componentes a serem utilizados, existindo diversos tipos de proteções disponibilizadas pelo mesmo. Para que se entenda o perfeito funcionamento desta tecnologia, serão abordados os componentes mais importantes, tais como os dois modos de pacotes criados, as associações seguras, o gerenciamento das chaves públicas e privadas, além da manipulação dos pacotes de dados. Todas estas características objetivam o tráfego das informações da maneira mais segura possível.

4.7.1.1. Modo Transporte

Nos pacotes criados no modo transporte são adicionados cabeçalhos IPSec entre o cabeçalho IP original e os dados, conforme a Figura 25.

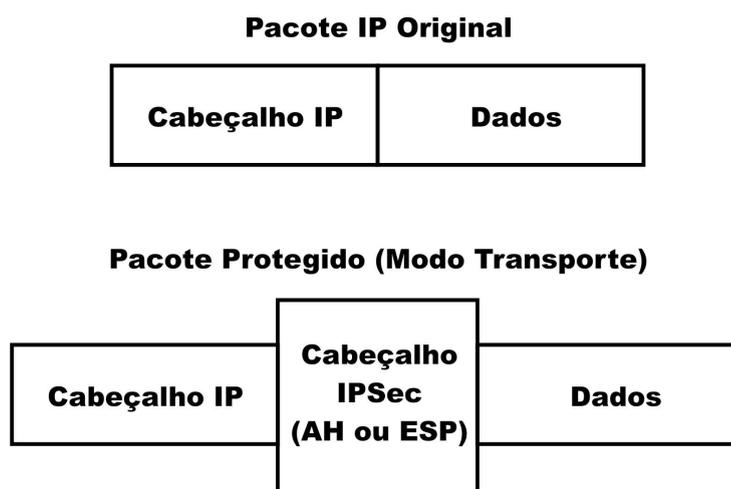


Figura 25: Modo transporte.

Fonte: KOLENISKOV e HATCH, 2002, p.138.

Este modo é muito utilizado para computadores em diferentes redes, comunicando-se diretamente entre si, e que desejam proteger o seu tráfego IP por encapsulamento, autenticação ou ambos.

4.7.1.2. Modo Túnel

Neste modo, um cabeçalho IPsec também é adicionado, porém a diferença para o modo transporte é que será adicionado um novo cabeçalho IP e o pacote original será tratado como se fosse um dado só, sendo todo ele encriptado pelo cabeçalho IPsec na parte referente ao dado do novo cabeçalho, de acordo com a Figura 26.

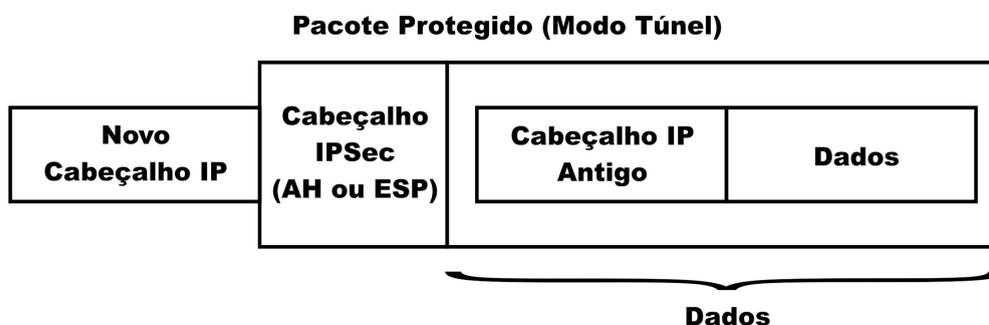


Figura 26: Modo túnel.

Fonte: KOLENISKOV e HATCH, 2002, p.138.

O modo túnel é comumente utilizado na comunicação entre *gateways*, pois fornece maior segurança aos dados originais encriptados dentro do novo pacote.

4.7.1.3. Security Association (SA)

As SAs (em português, Associações Seguras) são responsáveis por alocar informações sigilosas para a proteção do tráfego entre computadores ou *gateways*, como algoritmos de criptografia, chaves secretas ou seqüência de números. Todos os dados das SAs são armazenados dentro de uma banco de dados chamado *Security Association Database (SAD)*, tendo seu acesso restrito à pessoas autorizadas.

Para que sejam feitas as escolhas dos algoritmos de encriptação ou das funções *hash* a serem utilizadas para proteger os pacotes, as chaves ou as

seqüências de números, são definidos parâmetros para cada SA. Os exemplos mais utilizados são: endereços IPs de origem ou destino; números de origem ou destino das portas de comunicação; protocolos de segurança; assinaturas ou certificados digitais; modo transporte ou modo túnel; ou os *Security Parameters Index (SPI)* – índices de 32 bits incorporados aos cabeçalhos dos pacotes que juntamente com o endereço IP de destino e com os protocolos de segurança, identificam a melhor SA para usar no pacote.

4.7.1.4. *Security Policy Database (SPD)*

Este é o banco de dados responsável pela ação mais importante a ser feita sobre o pacote, que consiste em determinar se o mesmo será descartado, seguirá apenas em frente ou seguirá adiante aplicando-se o protocolo IPSec.

Diferentemente do SAD, o SPD força política de segurança sobre os pacotes, enquanto que o outro apenas referencia através de parâmetros as políticas a serem usadas.

Existe uma grande diferença no tratamento de pacotes recebidos (Tráfego que entra) e de pacotes a serem enviados (Tráfego que sai). No Tráfego que entra, o SAD é aplicado antes do SPD, pois serão verificadas a identidade do emissor e a integridade dos pacotes, para depois enviá-los à regra SPD correspondente. No Tráfego que sai, o SPD é acionado para verificar qual ação será aplicada ao pacote e, se necessário, dependendo da política de segurança, o IPSec poderá consultar o SAD.

4.7.1.5. *Gerenciamento de Chaves*

Em redes com poucos computadores pode-se configurar as SAs e os SPDs manualmente. Todavia, se a rede for grande, o recomendado é configurá-los automaticamente, o que acarretará no uso de um gerenciamento de chaves automático.

Alguns protocolos implementam de forma satisfatória este gerenciamento, porém o principal deles é o *Internet Key Exchange Protocol (IKE)*, que combina os

protocolos *Internet Security Association and Key Management Protocol (ISAKMP)*, *SKEME* e *Oakley Key Determination Protocol*.

O IKE negocia automaticamente as SAs, permitindo maior proteção contra ataques e, se habilitado o *Perfect Forward Secrecy (PFS)*, garante que se uma das chaves secretas estiver comprometida, apenas os dados encriptados por esta chave estarão. Caso contrário, todas as outras chaves e SAs podem ser afetadas.

Este protocolo funciona em duas fases: a primeira estabelece um canal de comunicação seguro entre as duas extremidades que desejam trocar dados. Para que isto seja possível é necessária a seguinte seqüência de operações: verificar os protocolos criptográficos e os parâmetros que serão utilizados para escolher os algoritmos de autenticação e de encriptação e, baseado nestas informações, as chaves secretas entre estas extremidades serão trocadas e, em seguida, uma associação segura ISAKMP é feita para então criar o canal; a segunda fase fornece as melhores associações seguras para proteger o tráfego IP pelo canal de comunicação criado na primeira fase. Isto é feito através dos seguintes passos: uma nova associação segura ISAKMP é estabelecida para a proteção das trocas de informações desta fase e, posteriormente, são negociados protocolos ou combinação de protocolos que posteriormente serão armazenados nas SADs de cada extremidade.

4.7.1.6. *Manipulação de Pacote de Dados*

Os protocolos IPSec fornecem segurança na manipulação de pacotes de dados IP através do *Authentication Header (AH)* e do *Encapsulating Security Payload (ESP)*.

O AH fornece autenticação, integridade e proteção a ataques do tipo Replay utilizando algoritmos simétricos para tornar difícil a modificação e a geração de funções *hash* para a interceptação de pacotes protegidos pelo AH. Os comandos para proteger os pacotes com o AH são: inserir um cabeçalho AH dentro do pacote a ser protegido; negociar uma associação segura que incrementará um valor a uma seqüência de números a cada proteção AH a um pacote de dados e, que será zerada cada vez que uma nova SA for negociada; adicionar as funções *hash MD-5*

ou SHA-1 ao AH, para que um novo código de autenticação, baseado no conteúdo do pacote, seja criado; e dividir o pacote já protegido pelo AH, se necessário.

O ESP fornece, além das características do AH, a confidencialidade, pois quando utilizado, ele adiciona seu cabeçalho (logo após o cabeçalho AH, caso este também esteja sendo usado) ao pacote e encripta a parte relativa aos dados (*Payload*) com algoritmos como o 3DES. Para a utilização deste protocolo, os seguintes passos são necessários: inserir os campos relativos ao seu cabeçalho no pacote; encriptar o *Payload* do pacote com o algoritmo e os parâmetros especificados na SA correspondente; incrementar um valor a uma seqüência de números da mesma forma que o AH; caso o AH não esteja sendo usado, calcular um valor para autenticação; e, como feito no AH, dividir o pacote, se necessário.

4.7.2. Arquitetura IPSec

A arquitetura deste protocolo funciona de duas formas: para pacotes IPs enviados (Tráfego que sai) ou para pacotes IPs recebidos (Tráfego que entra).

No primeiro tipo, o tráfego que sai, o pacote IP que se deseja enviar, é analisado pelo IPSec a fim de verificar se o mesmo irá prosseguir para seu destino aplicando ou não a proteção do protocolo. Para que isto seja determinado, o SPD é acionado e verifica o tipo, a origem e o destino do pacote, e então, dependendo destas características, determina a melhor opção. No exemplo do Quadro 1 os pacotes IPs, originados de qualquer porta da rede 192.168.1.1, com destino a porta 80 da rede 192.168.2.1, precisarão passar pela regra número 1, a qual coloca que é necessário o IPSec utilizar o modo túnel, o AH e o parâmetro 400⁹ da SA.

⁹ Parâmetro SA que define as chaves e as seqüências de números, negociados previamente pelo IKE, para proteção de ataques do tipo Replay.

Regra			Porta	Porta	Protocolo			Índice
#	IP Origem	IP Destino	Origem	Destino	Ação	IPSec*	Modo*	SA que Sai
1	192.168.1.1	192.168.2.1	Qualquer	80	IPSec	AH	Túnel	400
2	192.168.1.23	192.168.2.5	Qualquer	22	Aceitar	-----	-----	8500
3	192.168.1.99	192.168.2.1	Qualquer	Qualquer	IPSec	ESP	Transporte	6025

Quadro 1: Exemplo de um SPD. Os campos marcados com um asterisco podem ser omitidos, pois eles estão presentes na associação segura (SA) a qual este SPD se refere.

Fonte: KOLENISKOV e HATCH, 2002, p.134.

O tráfego que entra, ao contrário do que sai, analisa o pacote primeiramente no SAD, utilizando o SPI, o AH e o ESP, pois é necessário verificar o índice que será referenciado no banco de dados da associação segura, a identidade do emissor do pacote e extrair o dado do pacote encapsulado, pois este poderá conter um cabeçalho TCP encriptado, com, por exemplo, um número de porta de comunicação, que será necessário quando se for fazer a referência ao SPD. Após esta primeira parte, será referenciada a regra apropriada ao SPD que, então, checará, a partir das informações do pacote, o que fazer com o mesmo. O Quadro 2 mostra o funcionamento de um SAD que posteriormente referenciará a um SPD para tratar um pacote recebido.

SPI	IP Origem	IP Destino	Porta	Porta	Vários Parâmetros (AH/ESP, Modo, Tempo de Vida do SA, Chaves, etc.)	Tipo	Ponteiro para Entrada SPD
			Origem	Destino			
1	192.168.2.1	192.168.1.1	Qualquer	Qualquer	...	Que Entra	4
3	192.168.1.1	192.168.2.1	Qualquer	80	...	Que Sai	7

Quadro 2: Exemplo de um SAD.

Fonte: KOLENISKOV e HATCH, 2002, p.142.

4.7.3. *Vantagens e Desvantagens*

O IPSec foi desenvolvido para trazer segurança à VPN, porém torna-se uma grande desvantagem a sua utilização em sistemas já comprometidos em relação a segurança dos dados, pois sua implementação é complexa e requer alto grau de esforço computacional por utilizar algoritmos de criptografia muito pesados. Outra desvantagem é não proteger contra ataques do tipo DoS. Para ter proteção contra este tipo de ataque, é necessária a utilização de um *firewall*, que poderá ser utilizado juntamente com o IPSec.

Entretanto, os bons resultados apresentados ao se utilizar esta tecnologia, fazem dela uma grande arma contra vários tipos de ataques e invasões, pois se for utilizada de forma correta, ou seja, combinar bem os recursos de protocolos como o AH e o ESP, ou saber quando tirar os melhores proveitos do modo Transporte e do modo Túnel, ou ainda, compartilhar seguramente as chaves secretas através do protocolo IKE, dificilmente um invasor descobrirá algum ponto falho na conexão.

Este capítulo abordou as principais características, componentes e topologias de uma Rede Privada Virtual, mostrando as suas vantagens e desvantagens, comparando-a com outras tecnologias. Além disso, os protocolos mais utilizados foram mostrados, para que se pudesse entender melhor como funciona uma VPN.

Estes conceitos serão utilizados no próximo capítulo, onde duas redes serão interligadas através de uma VPN em ambiente Linux, detalhando a sua instalação, configuração e resultados.

5. IMPLEMENTAÇÃO DE UMA VPN EM LINUX

Para configurar uma VPN é necessário escolher o sistema operacional dos *gateways* e a ferramenta que eles utilizarão para implementar um dos protocolos anteriormente citados. Existem inúmeras ferramentas disponíveis, com preços variados ou até gratuitas. Isso depende principalmente do sistema operacional utilizado e do tipo de suporte que se deseja obter.

Este trabalho preocupou-se com dois fatores básicos: segurança e baixo custo de implementação. Por isso, foi utilizado o Linux, sistema operacional bastante flexível, podendo ser configurado com o mínimo de opções possíveis, possibilitando maior segurança da rede. Além disso, por ser de livre distribuição, estando disponível gratuitamente pela Internet ou sendo facilmente encontrado em bancas de revistas, ele reduz drasticamente os gastos com licenças e custo total de propriedade.

A distribuição escolhida foi a *Red Hat* (RED..., 2002), pois é a mais utilizada em servidores do mundo inteiro, sendo reconhecida por sua estabilidade e servindo de modelo para o desenvolvimento de outras distribuições, como a Conectiva Linux.

A ferramenta adotada para a configuração da VPN no sistema operacional acima citado foi o *FreeS/WAN* (FREESWAN, 2002), cuja denominação origina-se do termo *Secure Wide Area Network* (S/WAN), sendo a palavra *Free* devido esta ser gratuita.

Apesar do sistema operacional e da ferramenta serem completamente gratuitas, as orientações para instalá-los e configurá-los são obtidas apenas através de livros, documentações on-line e listas de discussão, onde participam especialistas no assunto. Poder-se-á ainda pagar ao fabricante do sistema operacional para obter suporte técnico por um certo período ou contratar-se uma empresa especializada para configurar a VPN.

A seguir será mostrado como a VPN foi implementada em caráter experimental, detalhando o cenário dos testes e os aspectos de sua configuração, além dos resultados obtidos.

5.1. CENÁRIO

Baseando-se na Figura 27, os testes foram realizados simulando-se duas redes distintas de uma mesma empresa, cujas denominações adotadas foram Rede A e Rede B, onde cada uma delas possui um *gateway* VPN e um *host*. Para simular o meio público de comunicação, ou seja, a Internet, foi postado um computador entre elas, cuja função é rotear os dados entre as redes, além de monitorar o fluxo da comunicação, através de um *sniffer*¹⁰.

Os *gateways* são compostos por duas placas de rede, sendo uma voltada para a rede interna da empresa (chamada Eth1 no Linux) e a outra para a Internet (chamada Eth0). Os *hosts*, por sua vez, possuem apenas uma placa de rede e funcionam em ambientes *Windows*.

O computador que simula a Internet também possui duas placas de rede, sendo uma delas voltada para a Rede A e a outra para a Rede B, servindo como roteadores de ambas.

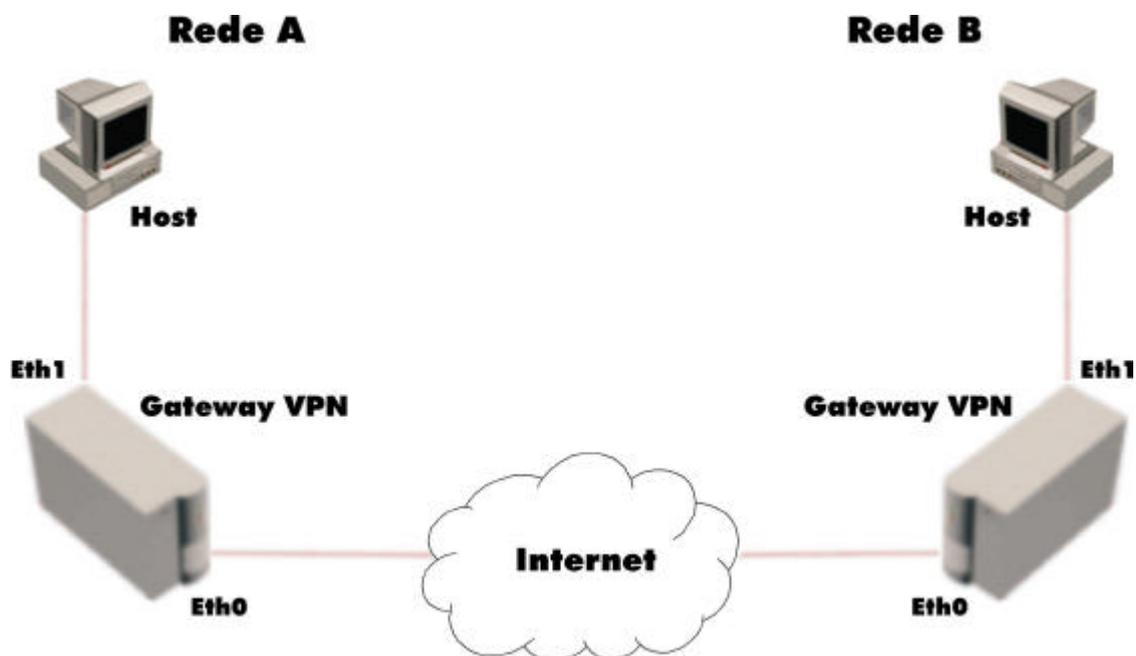


Figura 27: Cenário da implementação.

¹⁰ Programa que monitora o tráfego de uma rede.

5.2. FreeS/WAN

Desenvolvida em 1996, esta ferramenta implementa os protocolos do IPSec, com a finalidade de proteger a comunicação da Internet contra ataques de interceptação, utilizando um mecanismo gratuito.

Por usar recursos criptográficos, o *FreeS/WAN* sofre com algumas leis de exportação, principalmente as dos Estados Unidos, que proíbem a exportação de qualquer software criptográfico de computadores deste país, sem a prévia autorização do governo. Em geral, estas leis restringem inclusive quando o software já é de domínio público ou são oriundos de outros países. Por esta razão, o projeto não pode aceitar qualquer contribuição de cidadãos americanos, nem mesmo para correções de problemas no software.

Como o fundador do projeto do *FreeS/WAN*, John Gilmore, é americano, ele não pode desenvolver o código, função essa desempenhada por uma equipe de canadenses, residentes no Canadá, a fim de evitar problemas com o governo.

Esta ferramenta implementa como padrão os algoritmos de chaves assimétrica RSA e simétrica 3DES, além das funções *hash* MD-5 e SHA-1. Entretanto, diversos programadores contribuem através de *patches*¹¹ com funções extras não incorporadas ao programa original, seja para aumentar a segurança da VPN, seja para permitir a utilização deste em diferentes ambientes. Os principais são: suporte a certificados digitais X.509¹²; adição de algoritmos de criptografia de chave simétrica, como o AES, *Blowfish* e *Serpent*¹³; passagem pelo *Network Address Translation* (NAT), quando este estiver implementado¹⁴, entre outros. Existe ainda um *mirror* do *FreeS/WAN*, que incorpora os principais *patches* existentes ao programa original, chamando-o de Super *FreeS/WAN*¹⁵, facilitando a instalação desta ferramenta.

¹¹ Alteração de um programa, que acrescenta ou modifica apenas uma pequena parte deste.

¹² (X.509..., 2002)

¹³ (MODULAR..., 2002)

¹⁴ (NAT..., 2002)

¹⁵ (SUPER..., 2002)

Dois componentes fundamentais fazem parte do *FreeS/WAN: Kernel*¹⁶ *IPSec Support* (KLIPS) e Pluto.

5.2.1. KLIPS

Este componente faz a interação necessária do IPSec com o *kernel* do Linux, permitindo o tratamento de pacotes IPSec, incluindo encriptação, cálculos de autenticação de pacote, criação dos cabeçalhos AH e ESP para pacotes que saem e a interpretação dos mesmos para os que entram. Ele é responsável, ainda, por verificar todos os pacotes não-IPSec para assegurar que eles não estão passando pelas políticas de segurança deste protocolo.

5.2.2. Pluto

Este *daemon*¹⁷ implementa o protocolo IKE e é responsável pelo tratamento de toda a fase um das associações seguras do ISAKMP; execução das autenticações do *host* e das negociações com os demais *gateways*; criação dos SAs do IPSec e repasse dos dados requeridos para rodá-los ao KLIPS; e ajuste do roteamento e das configurações de *firewall* necessárias para adequação aos requerimentos do IPSec.

5.3. IMPLEMENTAÇÃO

Por não fazer parte dos objetivos deste trabalho, a instalação do Linux não será abordada neste capítulo. Contudo, alguns pacotes da distribuição *Red Hat 7.3* são necessários para a correta instalação e utilização do *FreeS/WAN*, conforme abordado na seção 5.3.1.

A implementação foi realizada baseando-se nas seguintes configurações de rede dos computadores utilizados:

¹⁶ Parte do Sistema Operacional que trata do hardware básico.

¹⁷ Programa permanentemente ativo em um sistema ligado na rede que aguarda instruções externas para disparar ações específicas.

- **Rede A:**

- **Gateway:**

- Hostname - gwredea*

- Gateway - 200.240.1.1*

- Placa de rede Eth0:

- IP - 200.240.1.2*

- Network - 240.240.1.0*

- Netmask - 255.255.255.0*

- Broadcast - 200.240.1.255*

- Placa de rede Eth1:

- IP - 192.168.1.1*

- Network - 192.168.1.0*

- Netmask - 255.255.255.0*

- Broadcast - 192.168.1.255*

- **Host:**

- Placa de rede:

- IP - 192.168.1.5*

- Hostname - hosta*

- Netmask - 255.255.255.0*

- Gateway - 200.240.1.2*

- Network - 192.168.1.0*

- DNS Primário - 200.240.1.2*

- Broadcast - 192.168.1.255*

- DNS Secundário - 200.240.2.2*

- **Rede B:**

- **Gateway:**

- Hostname - gwredeb*

- Gateway - 200.240.2.1*

- Placa de rede Eth0:

- IP - 200.240.2.2*

- Network - 240.240.2.0*

- Netmask - 255.255.255.0*

- Broadcast - 200.240.2.255*

- Placa de rede Eth1:

- IP - 192.168.2.1*

- Network - 192.168.2.0*

- Netmask - 255.255.255.0*

- Broadcast - 192.168.2.255*

- HostPlaca de rede:*IP* - 192.168.2.5*Hostname* - hostb*Netmask* - 255.255.255.0*Gateway* - 200.240.2.2*Network* - 192.168.2.0*DNS Primário* - 200.240.2.2*Broadcast* - 192.168.2.255*DNS Secundário* - 200.240.1.2• **Internet:***Hostname* - internet*Gateway* - 200.240.1.1Placa de rede Eth0:*IP* - 200.240.1.1*Network* - 240.240.1.0*Netmask* - 255.255.255.0*Broadcast* - 200.240.1.255Placa de rede Eth1:*IP* - 200.240.2.1*Network* - 240.240.2.0*Netmask* - 255.255.255.0*Broadcast* - 200.240.2.255

Como pré-requisito para o correto funcionamento de uma VPN, é necessário que o *Domain Name System* (DNS) das redes interligadas esteja bem configurado. Neste trabalho, optou-se pela criação de dois subdomínios fictícios, *redea.vpn.com.br* e *redeb.vpn.com.br*, onde cada *gateway VPN* é o servidor primário de seus respectivos subdomínios.

5.3.1. Instalação do FreeS/WAN

Existem duas formas de instalar esta ferramenta: através de pacotes *Red Hat Packet Manager* (RPM) ou compilando-a dos códigos fontes. Em ambos, é necessário que você esteja utilizando a conta *root*¹⁸.

¹⁸ Conta de usuário que possui todos os privilégios no Linux.

No primeiro método, são necessários dois arquivos, que variam de acordo com a versão do *kernel* utilizado¹⁹. Por exemplo, se a versão do kernel for 2.4.18-3, serão necessários os arquivos `freeswan-module-1.99_2.4.18_3-0.i386.rpm` e `freeswan-1.99_2.4.18_3-0.i386.rpm`. Estes arquivos são encontrados a partir da seção de *downloads* do *site* oficial do FreeS/WAN²⁰.

Em seguida, a instalação dos pacotes pode ser feita através dos comandos:

```
rpm -ivh freeswan-module-1.99_2.4.18_3-0.i386.rpm
rpm -ivh freeswan-1.99_2.4.18_3-0.i386.rpm
```

O segundo método, adotado neste trabalho, é um pouco mais complicado e requer a compilação de um novo *kernel*, com suporte ao IPSec. Para esta tarefa, são necessários alguns pacotes, tais como:

- GNU C Compiler (GCC ou Egcs)
- Glibc-devel
- Glibc-kernheaders
- Gmp
- Gmp-devel
- Kernel-source
- Make
- Ncurses-devel
- Patch

É importante ressaltar que esses pacotes são válidos apenas para a distribuição utilizada nesta implementação, variando em relação às demais.

O primeiro passo é fazer o *download* do arquivo `freeswan-1.99.tar.gz`²¹ e copiá-lo para a pasta `/usr/src`. Em seguida, executar os comandos:

```
cd /usr/src
tar xvfz freeswan-1.99.tar.gz
```

¹⁹ Para saber a versão do *kernel* instalado, basta digitar o comando `uname -a` no Linux.

²⁰ A partir da seção *downloads*, encontra-se o endereço `<ftp://ftp.xs4all.nl/pub/crypto/freeswan/binaries/RedHat-RPMs/>`, onde é possível adquiri-los.

²¹ Este arquivo também é encontrado a partir da seção *downloads* do *site* oficial do FreeS/WAN, no endereço `<ftp.xs4all.nl/pub/crypto/freeswan/>`.

Isso gerará a pasta `/usr/src/freeswan-1.99`. Se for necessária a inclusão de patches, como algoritmos de chave privada extras, suporte a NAT, certificados X.509 ou outros, esse é o momento de instalá-los.

Após a instalação dos *patches*, se houver, deve-se compilar o *kernel* e o *FreeS/WAN*. Para melhor entender esses procedimentos, eles serão divididos em duas partes:

- **Instalação do novo *kernel* antes de compilar o *FreeS/WAN*:**

Antes de compilar o *kernel*, o código fonte do mesmo deve estar presente em `/usr/src/linux`. Por isso, no *Red Hat 7.3*, é preciso fazer um link simbólico para esta pasta, através do comando:

```
ln -s /usr/src/linux.2.4 /usr/src/linux
```

Para facilitar a compilação de um novo *kernel*, pode-se fazer uma cópia do arquivo de configuração do *kernel* original, seguindo estes passos:

```
cd /usr/src/linux
```

```
cp configs/kernel-2.4.18-ix86.config .config
```

(onde *x* corresponde à arquitetura do processador existente no computador)

Em seguida, começa-se a fase de compilação do *kernel* com um dos comandos a seguir, onde são escolhidas as opções que o *kernel* irá suportar, como periféricos, tipos de hardware, configurações de rede, entre outros.

```
make menuconfig
```

(Escolhe-se as opções utilizando um menu gráfico. Esta opção é de fácil utilização, por isso foi adotada neste trabalho)

```
make config
```

(Escolhe-se as opções através de perguntas em modo texto. Se você errar uma opção, não tem como retornar)

```
make xconfig
```

(Similar ao *menuconfig*, porém utilizada quando uma interface gráfica, como KDE ou Gnome, está instalada)

```
make oldconfig
```

(Utiliza-se as mesmas opções do *kernel* atual)

Para o *FreeS/WAN* funcionar, não é preciso fazer nenhuma alteração. Contudo, deve-se salvar o que foi escolhido. O próximo passo é executar a seqüência de comandos a seguir.

```
make dep
```

(Encontra dependências entre os arquivos)

`make bzImage` (Constrói uma imagem do *kernel*, para que ele possa ser inicializado quando o computador for ligado)

`make install` (Instala o *kernel*)

`make modules` (Constrói os módulos que podem ser adicionados ao *kernel* que está sendo executado)

`make modules_install` (Instala os módulos)

- **Compilação do *FreeS/WAN*:**

Após a compilação do novo *kernel*, começa a fase de compilação do *FreeS/WAN*, entrando na pasta onde ele foi instalado (`cd /usr/src/freeswan-1.99`) e digitando um dos comandos a seguir, que faz com que as opções da ferramenta sejam configuradas para o *kernel* atual.

`make menugo` (Invoca o *menuconfig*. Opção utilizada por este trabalho)

`make ogo` (Invoca o *config*)

`make xgo` (Invoca o *xconfig*)

`make oldgo` (Usa as configurações padrões, mantendo todas as opções utilizadas da última configuração do *kernel*)

Se o primeiro comando for utilizado, para saber se está tudo correto com as opções do *FreeS/WAN*, pode-se ir até à seção *Networking Options* (Opções de Rede) e procurar as seguintes opções:

```
< M > IP Security Protocol (FreeS/WAN IPSEC)
  --- IPsec Options (FreeS/WAN)
  [ * ]      IPSEC: IP-in-IP Encapsulation (Tunnel mode)
  [ * ]      IPSEC: Authentication Header
  [ * ]          HMAC-MD5 authentication algorithm
  [ * ]          HMAC-SHA1 authentication algorithm
  [ * ]      IPSEC: Encapsulation Security Payload
  [ * ]          3DES encryption algorithm
  [ * ]      IPSEC: IP Compression
  [ * ]      IPSEC: IP Debugging Option
```

Onde M significa que a opção será instalada como um módulo. De acordo com Koleniskov e Hatch (2002, p. 174), é recomendado que o IPsec seja instalado com um módulo, pois o estado das informações do IPsec no *kernel* pode ser completamente zerado simplesmente removendo o módulo e inserindo-o novamente. O asterisco significa que a opção será instalada junto com o IPsec. Para remover uma das opções, basta deixá-la em branco.

As configurações devem ser salvas, mesmo que nenhuma mudança tenha sido feita. Isto é necessário para que o *FreeS/WAN* seja visto pelo sistema.

Para adicioná-lo ao *kernel*, deve-se digitar o comando:

```
make kinstall
```

Ele equivale à seqüência de comandos:

```
make
make install
make modules
make modules_install
```

Após esses passos, deve-se adicionar o novo *kernel* ao Linux *Loader* (LILO) ou ao *Grand Unified Bootloader* (GRUB) para que o mesmo esteja presente na inicialização do computador, finalizando o processo de instalação do *FreeS/WAN*.

Para saber se ele foi instalado corretamente, reinicia-se o computador e digita-se o comando `ipsec verify` em qualquer diretório do Linux. Isso deve gerar a saída a seguir, onde mostra que o novo *kernel* e a ferramenta foram compilados corretamente.

```
Checking your system to see if IPsec got installed and started
correctly
  Version check and ipsec on-path                               [OK]
  Checking for KLIPS support in kernel                         [OK]
  Checking for RSA private key (/etc/ipsec.secrets)           [OK]
  Checking that pluto is running                               [OK]
```

Para a VPN funcionar sem impedimentos, é necessário ainda habilitar o repasse de pacotes IP (*IP Forwarding* ou *ip_forward*) e desabilitar o filtro de caminho inverso (*Reverse Path Filter* ou *rp_filter*) nos *gateways* VPN. Este filtro é uma característica de segurança do Linux que previne ataques do tipo DoS. Ele permite que o *kernel* abandone os pacotes que entram na rede e cujas respostas não

correspondem à mesma interface daqueles que chegaram. Isso é um problema, visto que o *FreeS/WAN* cria interfaces do tipo *ipsecN*, onde *N* corresponde a um número que começa em 0, fazendo com que o *kernel* abandone os pacotes pertencentes à VPN. Para executar tais tarefas, pode-se editar o arquivo `/etc/rc.d/rc.local`, por ser o último arquivo de inicialização do Linux, e adicionar as linhas:

```
echo 1 /proc/sys/net/ipv4/ip_forward
echo 0 /proc/sys/net/ipv4/conf/eth0/rp_filter
```

Se existir um *firewall* na rede, é preciso que todo tráfego que passa pela porta UDP 500 (IKE) e pelos protocolos 50 (ESP) e 51 (AH) seja permitido.

5.3.2. Configuração do FreeS/WAN

Depois da instalação da ferramenta, será necessário configurar dois arquivos: `/etc/ipsec.secrets` e `/etc/ipsec.conf`. Estes são a base do funcionamento da VPN, pois possuem as configurações das chaves utilizadas e dos túneis criados, respectivamente.

A partir da versão 1.98b do *FreeS/WAN*, as chaves RSA são geradas automaticamente no ato da instalação, poupando, assim, tempo e trabalho do administrador da rede.

A seguir está o arquivo `/etc/ipsec.secrets`²² do *gateway* VPN da Rede B:

```
: RSA {
# RSA 2192 bits gwreddeb Fri Dec 1 12:13:34 2002
# Para assinaturas apenas, INSEGURO PARA ENCRIPTAÇÃO
#pubkey=0sAQOgYlOR/M8ZPGVY8vyOhku...
#IN KEY 0x4200 4 1 AQOgYlOR/M8ZPGVY8vyOhkutyQ...
# (0x4200 = auth-only host-level, 4 = IPsec, 1 = RSA)
Modulus: 0xa0625391fccf193c6558f2fc8e864bad...
PublicExponent: 0x03
```

²² Foram colocadas reticências em alguns itens, devido o conteúdo destes ser muito extenso.

```

# Tudo a partir daqui é secreto
PrivateExponent: 0x1abb0dedaa228434bb8ed32a17...
Prime1: 0xe7537a53a6aee41fd0ab65205...
Prime2: 0xb17dba2d1e104c157af9dbdd1...
Exponent1: 0x9a37a6e26f1f42bfe072436ae59...
Exponent2: 0x7653d173696032b8fca6929361d...
Coefficient: 0x2fa20d41911269b1f8e18552b16a0...
}

```

A linha deste arquivo que interessa para a implementação é a *pubkey*, pois é nela que está presente a chave pública do servidor e que, posteriormente, será utilizada para fazer a autenticação dos *gateways* VPN.

O arquivo `/etc/ipsec.conf` do *gateway* VPN da Rede A está configurado da seguinte forma:

```

config setup
    interfaces=%defaultroute
    klipsdebug=none
    plutodebug=none
    plutoload=%search
    plutostart=%search
    uniqueids=yes
conn %default
    keyingtries=0
    authby=rsasig
conn redea-redeb
    left=200.240.1.2
    leftsubnet=192.168.1.0/24
    leftnexthop=200.240.1.1
    leftid=@gwredea.redea.vpn.com.br
    leftrsasigkey=0sAQOcMDf3Dd7OYEqzdgXf7fphiWD...
    right=200.240.2.2
    rightsubnet=192.168.2.0/24
    rightnexthop=200.240.2.1
    rightid=@gwredeb.redeb.vpn.com.br
    rightrsasigkey=0sAQOgYlOR/M8ZPGVY8vyOhkutYQb...
    auto=add

```

Este arquivo é dividido em dois tipos de seções: *config* e *conn*. O primeiro é composto da seção *config setup*, que define as informações que serão utilizadas quando o programa estiver sendo iniciado. Ela é única, não existindo outras seções que iniciem com *config*. O segundo tipo é formado por uma ou várias seções *conn*, que definem as conexões de rede que serão feitas usando o IPSec. A conexão *%default* especifica parâmetros que serão utilizados por todas as outras. É possível criar uma conexão para cada túnel que se pretenda construir. Cada um dos parâmetros utilizados por este trabalho será explicado a seguir.

- *Interfaces* – Define as interfaces físicas e virtuais que o IPSec utiliza. A opção *%defaultroute* faz com que o programa faça a procura automaticamente. Outra forma de configurar essa opção seria *interfaces="ipsec0=eth0"*.
- *Klipsdebug* – Determina quanto da saída de depuração do KLIPS deve ser armazenado. O valor *none* significa que não será armazenado nada.
- *Plutodebug* – Determina quanto da saída de depuração do Pluto deve ser armazenado. O valor *none* significa que não será armazenado nada.
- *Plutoload* – Identifica quais conexões são carregadas à base de dados do Pluto na inicialização do *FreeS/WAN*. O valor *%search* define que as conexões que contenham *auto=route*, *auto=add* ou *auto=start* são carregadas.
- *Plutostart* – Identifica quais conexões tentam ser negociadas no início do programa, onde o valor *%search* equivale a rotear todas as conexões que possuem as linhas *auto=route* ou *auto=start*, além de iniciar as que contenham *auto=start*.
- *Uniqueids* – Estipula se o ID de um *gateway* deve ser único.
- *Keyingtries* – Estipula quantas tentativas devem ser feitas para negociar uma conexão. O valor 0 significa que o protocolo nunca desiste da negociação.

- *Authby* – Define como os *gateways* farão a autenticação entre eles. Os possíveis valores são *secret* (utilizando chaves secretas compartilhadas) e *rsasig* (utilizando assinaturas digitais RSA).
- *Left* – IP do *gateway* da esquerda. A ordem dos *gateways* (esquerda ou direita) fica a critério do administrador e não interfere no funcionamento da VPN.
- *Leftsubnet* – Subrede privada por trás do *gateway* esquerdo.
- *Leftnexthop* – Próximo salto do *gateway* da esquerda em relação à rede pública.
- *Leftid* – Define como o *gateway* esquerdo deve ser identificado para a autenticação.
- *Leftrsasigkey* – Contém a chave pública do *gateway* esquerdo.
- *Right* – IP do *gateway* da direita.
- *Rightsubnet* – Subrede privada por trás do *gateway* direito.
- *Rightnexthop* – Próximo salto do *gateway* da direita em relação à rede pública.
- *Rightid* – Define como o *gateway* direito deve ser identificado para a autenticação.
- *Rightrsasigkey* – Contém a chave pública do *gateway* direito.
- *Auto* – Especifica que operação deve ser feita automaticamente, quando o IPSec for iniciado. O valor *add* apenas adiciona a conexão, não a iniciando. Os demais valores são *route*, *start* e *ignore*.

Após a configuração deste arquivo em um dos *gateways*, o mesmo foi copiado para o outro e nele digitou-se o comando abaixo para iniciar o túnel:

```
ipsec auto --up redea-redeb
```

Isso finaliza o processo de configuração do *FreeS/WAN* e o próximo passo é verificar se o túnel foi estabelecido corretamente. Neste trabalho, tal resposta foi obtida através de testes de *ping*²³ entre os *hosts*, da consulta pelo *host* remoto no ambiente de rede e da monitoração utilizando um *sniffer* no computador que representava a Internet, sendo todos eles bem sucedidos.

5.3.2.1. *Encriptação Oportunista*

Um dos grandes objetivos do *FreeS/WAN* está na criação da Encriptação Oportunista. Ela não é considerada um padrão do protocolo IPSec, mas está sendo proposta para ser uma extensão dele. Portanto, ela só é possível quando existem, pelo menos, dois *gateways* com o *FreeS/WAN* instalado.

Este tipo de encriptação permite que os *gateways* sejam hábeis a criptografar o tráfego existente entre eles, mesmo que não exista qualquer relação entre os mesmos. Para isso, eles precisam ser configurados de forma que as informações de autenticação estejam presentes no DNS e o *FreeS/WAN* seja habilitado para encriptar oportunamente. Isso faz com que todo o processo aconteça automaticamente.

A Encriptação Oportunista reduz drasticamente a sobrecarga administrativa em cima do IPSec, uma vez que não são necessárias configurações específicas de túneis, apesar de ainda ser permitido configurá-los para casos específicos. Outra vantagem está no fato dela deixar o tráfego da Internet mais seguro, pois todo o conteúdo transmitido e recebido é encriptado.

Existem dois tipos de encriptação oportunista: parcial e total. A principal diferença está na forma que a conexão é feita. Na parcial, apenas é possível iniciar conexões oportunistas, enquanto que na total, é possível tanto iniciá-las quanto aceitá-las.

²³ Diagnóstico utilizado para constatar se um computador está ativo e verificar o tempo que o mesmo demora a responder uma solicitação.

5.4. CONSIDERAÇÕES SOBRE A IMPLEMENTAÇÃO

De acordo com o cenário utilizado durante a implementação e dos testes realizados, pôde-se verificar através do *sniffer*, que todo o conteúdo trafegado entre os *hosts* estavam encriptados e que o protocolo reconhecido por ele era o ESP. Desta forma, não foi possível identificar o conteúdo destes dados, o que garantiu a confidencialidade das informações transmitidas, característica primordial para o correto funcionamento de uma VPN.

No entanto, algumas dificuldades foram enfrentadas na instalação do *FreeS/WAN*, pois o trabalho de compilação do *kernel* é complexo e requer conhecimentos específicos durante a configuração deste. A própria documentação do programa não é clara em relação a este aspecto, solicitando que o usuário consulte outras fontes de pesquisa.

A configuração do *FreeS/WAN* é relativamente complicada, pois é necessário que o administrador tenha domínio sobre a arquitetura IPsec e completa noção do funcionamento das redes interligadas, a fim de não enfrentar problemas com a configuração desta ferramenta.

Por utilizar recursos criptográficos, o desempenho da comunicação dos dados entre as redes é ligeiramente pior, pois o tamanho dos pacotes aumentam, porém o tráfego é seguro e as informações sigilosas podem ser transmitidas. Para solucionar este problema podem-se utilizar outros algoritmos criptográficos, como o AES, que, segundo testes realizados pela equipe do *FreeS/WAN* e por entidades como a NIST, mostrou ser mais veloz e seguro que o 3DES, seu antecessor.

6. CONCLUSÃO

Este trabalho se propôs a mostrar como implementar uma VPN em ambiente Linux utilizando o protocolo IPSec, detalhando aspectos de suas configurações, além de analisar a funcionalidade e a segurança da mesma.

Pôde-se observar que a VPN é uma tecnologia em crescimento, sendo comercializada, por exemplo, pelas principais empresas de telecomunicações no Brasil e no mundo. São inúmeros os serviços oferecidos por elas e por outras companhias que trabalham com interligação de redes.

As VPNs cada vez mais são pesquisadas e incorporadas às organizações, sejam privadas ou governamentais, em diversos sistemas e ambientes computacionais. Os principais motivos desta crescente procura estão relacionado à segurança e economia no tráfego das informações.

Esta segurança é obtida principalmente através da utilização de algoritmos de criptografia e protocolos específicos, que geram grandes obstáculos aos invasores. Algumas das funções de segurança garantidas por uma VPN são a integridade e a confidencialidade das informações transmitidas, assim como a autenticação e o controle de acesso dos *gateways*, permitindo maior confiança por parte das empresas que adotam esta solução.

Alguns dos protocolos abordados neste trabalho foram o PPTP e o L2TP, que se mostraram boas alternativas de implementação, porém pecam na segurança. Por sua vez, o IPSec mostrou-se um protocolo mais completo em relação a estes, possuindo características mais expressivas no intuito de proteger o tráfego das redes interligadas. Todavia, ele ainda é muito complexo e, por isso, é indispensável entender exatamente o seu funcionamento antes de implementá-lo em uma VPN.

É importante saber que nem o IPSec nem uma VPN sozinhos conseguem garantir a segurança de uma rede. É essencial que um planejamento cuidadoso seja feito, envolvendo políticas rígidas de segurança, permitindo que haja proteção física e lógica dos servidores.

O sistema operacional Linux mostrou-se muito eficaz, sendo uma solução atrativa para as organizações que não disponibilizam de muitos recursos financeiros

para a área de informática ou que não pretendem gastar valores exorbitantes com tecnologias proprietárias.

A ferramenta *FreeS/WAN* atendeu as demandas exigidas na implementação deste trabalho, mostrando-se capaz de implementar os protocolos do IPSec e fornecer características interessantes de segurança, como a encriptação oportunista. Contudo, ela precisa ser ajustada para suportar funções extras, como certificados digitais e outros algoritmos de criptografia simétricos, como o AES, de modo a deixar a VPN mais segura.

Devido às restrições de exportação de alguns países para programas que utilizam criptografia forte, o *FreeS/WAN* não está presente no *kernel* padrão do Linux, sendo necessária a sua inserção no mesmo. Além do mais, se uma ou mais funções extras citadas anteriormente precisassem ser instaladas, o processo de compilação de um novo *kernel* seria obrigatório. Por isso, neste trabalho, optou-se por este método, que, apesar de ser árduo e consumir mais tempo, facilita o entendimento sobre este protocolo. Por exemplo, durante a configuração das opções do *FreeS/WAN* que seriam adicionadas ao novo *kernel*, observou-se os principais componentes do IPSec, onde era possível compará-los com a teoria estudada e, ainda, desmembrar qualquer um deles do protocolo, personalizando a instalação e a configuração da VPN.

Analisando os tópicos abordados e relacionando-os com os resultados obtidos, pôde-se concluir que este trabalho, através de ampla pesquisa bibliográfica utilizada, contribuiu academicamente para conceituar os principais aspectos de segurança utilizados em VPNs, além de analisar diversas características das Redes Privadas Virtuais, como seus elementos, topologias e principais protocolos. Outra grande contribuição, esta de caráter científico e prático, foi a implementação de uma VPN utilizando o protocolo IPSec, interligando duas redes em caráter experimental, mostrando como esta foi configurada.

Os resultados alcançados por este trabalho foram satisfatórios e, por não serem necessários equipamentos complexos e caros, esta tecnologia pode contribuir para a diminuição do custo total de propriedade de uma empresa. A tendência é que ela seja ainda mais utilizada com o desenvolvimento do acesso à Internet em banda larga, tornando-se um atrativo para qualquer tipo de organização.

Como sugestão para trabalhos futuros e de forma a aprimorar os conhecimentos nesta tecnologia pode-se, por exemplo, implementar uma VPN em ambiente corporativo, como estudo de caso, interligando duas ou mais redes, e permitindo o acesso remoto de diretores e dos administradores das redes, para que sejam possíveis atualizações em tempo real, recuperação de possíveis erros na VPN e, principalmente, consultas da base de dados à longa distância, utilizando esta mesma ferramenta, além de dar suporte a certificados digitais X.509 e ao algoritmo simétrico AES, tendo o presente trabalho como subsídio básico para tal.

REFERÊNCIAS

3COM CORPORATION. 3Com OfficeConnect: Network Assistant. Santa Clara, EUA: 3Com, versão 2.02, 2000. 1 CD-ROM.

ALBUQUERQUE, Fernando. TCP/IP Internet: Protocolos e Tecnologias. 3ª Edição. Rio de Janeiro: Axcel Books, 2001. 362 p.

BASTOS, Eri Ramos. Configurando uma VPN IPSec FreeSwan no Linux. [S.l.], 2002. Disponível em: <<http://www.secforum.com.br/article.php?sid=1033>>. Acesso em: 08 nov 2002.

BATISTA, Thaís Vasconcelos. Segurança em Redes de Computadores. Natal, 2002. Aula 1. Disponível em: <<http://www.dimap.ufrn.br/~thais/Seguranca/home.html>>. Acesso em: 17 nov 2002.

BROCARD, Marcelo Luiz. I2AC: *um Protocolo Criptográfico para Análise Segura de Crédito*. 2001. 122 f. Dissertação de Mestrado em Ciência da Computação - Universidade Federal de Santa Catarina, Florianópolis.

BROWNE, Brian et al. Best Practices For VPN Implementation. [S.l.], 2001. Disponível em: <<http://www.bcr.com/bcsmag/2001/03/p24.asp>>. Acesso em: 12 nov 2002.

CCONE@NETWORK. Installing FreeS/WAN. [S.l.], 2001. Disponível em: <<http://info.ccone.at/INFO/freeswan/install.html>>. Acesso em: 01 dez 2002.

CYCLADES. Guia Internet de Conectividade. 6ª Edição. São Paulo: SENAC, 2000. 167 p.

D'ÁVILLA, Márcio H. C. Segurança de Redes. Belo Horizonte, 2001. Aula 2. Disponível em: <<http://www.inet.com.br/~mhavila/aulas/seguranca/material.html>>. Acesso em: 19 nov 2002.

FREES/WAN, Version 1.99. FreeS/WAN. [S.l.], 2002. Disponível em: <www.freeswan.org/download.html>. Acesso em: 13 nov 2002.

_____. FreeS/WAN 1.99 Documentation. [S.l.], 2002. Disponível em: <http://www.freeswan.org/freeswan_trees/freeswan-1.99/doc/index.html>. Acesso: 15 nov 2002.

GODIM, Marcelo. VPN quase fácil. [S.l.], [ca. 2001]. Disponível em: <<http://www.underlinux.com.br/modules.php?name=Sections&op=viewarticle&artid=63>>. Acesso em: 06 nov 2002.

KOLENISKOV, Oleg; HATCH, Brian. Building Linux Virtual Private Networks (VPNs). 1ª Edição. EUA: New Riders, 2002. 385 p.

MODULAR Algo Support, Version 0.8.0. Desenvolvido por Juan Jose Ciarlante. [S.I.], 2002. Disponível em: <<http://www.irrigacion.gov.ar/juanjo/ipsec>>. Acesso em: 20 nov 2002.

MONTEIRO, Edmundo. Segurança em Redes. Coimbra, Portugal, 1999. Capítulo 1. Disponível em: <<http://eden.dei.uc.pt/~sr/Teoricas/>>. Acesso em: 15 nov 2002.

NAPIER, Duncan. Introducing FreeS/WAN and IPsec. [S.I.], 2000. Disponível em: <<http://www.samag.com/documents/s=1159/sam0011i/0011i.htm>>. Acesso em: 15 dez 2002.

_____. Administering Linux IPSec Virtual Private Networks. [S.I.], 2002. Disponível em: <<http://www.samag.com/documents/s=4072/sam0203c/sam0203c.htm>>. Acesso em: 15 dez 2002.

_____. Setting up a VPN Gateway. [S.I.], 2002. Disponível em: <<http://www.linuxjournal.com/article.php?sid=4772>>. Acesso em: 15 dez 2002.

NAT Traversal, Version 0.4. Desenvolvido por Mathieu Lafon. [S.I.], 2002. Disponível em: <<http://open-source.arkoon.net/>>. Acesso em: 20 nov 2002.

RED Hat Linux, Version 7.3. Red Hat Inc. North Carolina, EUA, 2002. Disponível em: <www.redhat.com>. Acesso em: 13 nov 2002.

SCHNEIER, Bruce. Applied Cryptography. 2ª Edição. EUA: JohnWiley & Sons, 1996. 758 p.

SCRIMGER, Rob et al. TCP/IP: A Bíblia. 1ª Edição. Rio de Janeiro: Campus, 2002. 642 p.

SSH COMMUNICATIONS SECURITY CORP. VPN Connection to FreeS/WAN IPSec Gateway. Helsinki, Finlândia, 2002. Disponível em: <http://www.ssh.com/documents/31/ssh_sentinel_14_freeswan.pdf>. Acesso em: 15 nov 2002.

SUPER FreeS/WAN, Version 1.99_kb2. Desenvolvido por Ken Bantoft. [S.I.], 2002. Disponível em: <<http://www.freeswan.ca>>. Acesso em: 25 nov 2002.

TANENBAUM, Andrew S. Redes de Computadores. 3ª Edição. Rio de Janeiro: Campus, 1997. 923 p.

TORRES, Gabriel. Redes de Computadores: Curso Completo. 1ª Edição. Rio de Janeiro: Axcel Books, 2001. 664 p.

X.509 Certificate Support, Version 0.9.15. Desenvolvido por Andreas Steffen. [S.I.], 2002. Disponível em: <<http://www.strongsec.com/freeswan/>>. Acesso em: 20 nov 2002.