

CENTRO UNIVERSITÁRIO DE MARINGÁ
CURSO DE PÓS-GRADUAÇÃO EM AMBIENTES DE DESENVOLVIMENTO PARA
INTERNET

**ESTUDO SOBRE A IMPLANTAÇÃO DE PROPRIEDADES DE
SEGURANÇA EM CORREIO ELETRÔNICO ATRAVÉS DO PGP**

RODRIGO GUEDES DE SOUZA

MARINGÁ
2003

CENTRO UNIVERSITÁRIO DE MARINGÁ
CURSO DE PÓS-GRADUAÇÃO EM AMBIENTES DE DESENVOLVIMENTO PARA
INTERNET

**ESTUDO SOBRE A IMPLANTAÇÃO DE PROPRIEDADES DE
SEGURANÇA EM CORREIO ELETRÔNICO ATRAVÉS DO PGP**

RODRIGO GUEDES DE SOUZA

Trabalho apresentado ao Centro Universitário de Maringá como requisito parcial para obtenção do título de Especialista em Ambientes de Desenvolvimento para internet, sob orientação do Prof. Msc. Edson Yanaga.

MARINGÁ

2003

RODRIGO GUEDES DE SOUZA

**ESTUDO SOBRE A IMPLANTAÇÃO DE PROPRIEDADES DE
SEGURANÇA EM CORREIO ELETRÔNICO ATRAVÉS DO PGP**

Trabalho apresentado ao Centro Universitário de Maringá com requisito parcial para obtenção do título de Especialista em Ambientes de Desenvolvimento para Internet, sob a orientação do Prof. Msc. Edson Yanaga.

Aprovado em: _____

BANCA EXAMINADORA

Prof. Msc. Edson Yanaga
(Centro Universitário de Maringá)

Ao meu Deus Javé, fonte de minha fé, aos meus pais que me deram a oportunidade de estar apresentando este trabalho hoje, aos meus irmãos e a minha namorada por acreditar no meu potencial e me dar incentivo.

AGRADECIMENTOS

Primeiramente a meu Deus Javé, por ter me dado força para conseguir passar por esta fase de minha vida.

A minha família por estarem sempre pertos de mim, a minha namorada por existir e especialmente ao meu orientador Prof. Msc. Edson Yanaga por ter me incentivado e me orientado.

SUMÁRIO

BANCA EXAMINADORA.....	5
AGRADECIMENTOS.....	8
LISTA DE ILUSTRAÇÕES.....	11
LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS.....	12
RESUMO.....	14
1INTRODUÇÃO.....	15
2REVISÃO BIBLIOGRAFICA.....	17
2.1CORREIO ELETRÔNICO.....	17
<i>2.1.1Definição.....</i>	<i>17</i>
<i>2.1.2Estrutura.....</i>	<i>18</i>
<i>2.1.3Funcionalidades.....</i>	<i>18</i>
<i>2.1.4Formato das Mensagens.....</i>	<i>19</i>
<i>2.1.5RFC 822.....</i>	<i>19</i>
<i>2.1.6MIME.....</i>	<i>20</i>
<i>2.1.7Transferência de Mensagens.....</i>	<i>22</i>
<i>2.1.8POP.....</i>	<i>23</i>
<i>2.1.9IMAP.....</i>	<i>24</i>
2.2CRIPTOGRAFIA.....	25
<i>2.2.1Definição.....</i>	<i>25</i>
<i>2.2.2Algoritmos Simétricos.....</i>	<i>26</i>
<i>2.2.3Algoritmos Assimétricos.....</i>	<i>28</i>
2.3CONSIDERAÇÕES FINAIS.....	29
3PGP 31	
3.1HISTÓRICO.....	31
3.2ALGORITMOS UTILIZADOS.....	32

<u>3.2.1 Algoritmo Simétrico.....</u>	<u>33</u>
<u>3.2.2 Algoritmo Assimétrico.....</u>	<u>34</u>
<u>3.2.3 Funções de Hash.....</u>	<u>35</u>
<u>3.3 FUNCIONALIDADES.....</u>	<u>36</u>
<u>3.4 GERENCIAMENTO E DISTRIBUIÇÃO DAS CHAVES.....</u>	<u>39</u>
<u>3.4.1 Chaveiro do PGP.....</u>	<u>39</u>
<u>3.4.2 Forma de distribuição de chave.....</u>	<u>39</u>
<u>3.4.3 Validação de Chaves.....</u>	<u>40</u>
<u>3.4.4 Teia de confiança.....</u>	<u>41</u>
<u>3.4.5 Revogando Chaves Públicas.....</u>	<u>41</u>
<u>3.5 VULNERABILIDADES.....</u>	<u>42</u>
<u>3.5.1 Comprometimento da Frase-senha e Chave Privada.....</u>	<u>42</u>
<u>3.5.2 Falsificação da Chave Pública.....</u>	<u>42</u>
<u>3.5.3 Arquivos Não Apagados Completamente do Disco.....</u>	<u>43</u>
<u>3.5.4 Vírus e Cavalo de Tróia.....</u>	<u>43</u>
<u>3.5.5 Falhas de Segurança Física.....</u>	<u>44</u>
<u>3.5.6 Ataques Tempest.....</u>	<u>44</u>
<u>3.5.7 Exposição de Sistemas Multiusuário.....</u>	<u>45</u>
<u>3.5.8 Análise de Tráfico.....</u>	<u>45</u>
<u>3.5.9 Criptoanálise.....</u>	<u>45</u>
<u>4 INTEROPERABILIDADE (PGP V.S GNUPG).....</u>	<u>47</u>
<u>4.1 GNUPG.....</u>	<u>47</u>
<u>4.2 RELATO DO TESTE EFETUADO.....</u>	<u>48</u>
<u>4.3 CONSIDERAÇÕES FINAIS.....</u>	<u>49</u>
<u>5 UTILIZAÇÃO PRÁTICA DO PGP.....</u>	<u>50</u>
<u>5.1 CRIANDO UM PAR DE CHAVES.....</u>	<u>50</u>
<u>5.2 CONFIGURANDO O ALGORITMO SIMÉTRICO.....</u>	<u>54</u>
<u>5.3 DISPONIBILIZANDO CHAVE PÚBLICA ATRAVÉS DE UM SERVIDOR.....</u>	<u>54</u>
<u>5.4 EXPORTAR A CHAVE PÚBLICA EM TEXTO ASCII.....</u>	<u>56</u>
<u>5.5 ADICIONANDO CHAVES PÚBLICAS.....</u>	<u>57</u>

5.6UTILIZANDO PGP NO OUTLOOK 2000.....	59
5.6.1Enviando uma Mensagem.....	59
5.6.2Lendo uma Mensagem Criptografada.....	60
6CONCLUSÃO.....	62
REFERÊNCIAS.....	

LISTA DE ILUSTRAÇÕES

FIGURA 1 FORMATO DE UMA MENSAGEM DE CORREIO ELETRÔNICO.....	19
FIGURA 2 UMA MENSAGEM NO FORMATO RFC 822 / MIME ENVIADO PELO AGENTE DE USUÁRIO (MICROSOFT OUTLOOK, VERSÃO: 9.0.0.2814).....	22
FIGURA 3 O SMTP PRESSUPÕE COMUNICAÇÃO ON-LINE ENTRE AS MAQUINAS.....	23
FIGURA 4 TRANSFERÊNCIA DE MENSAGENS.....	24
FIGURA 5 O MODELO DE CRIPTOGRAFIA.....	26
FIGURA 6 SISTEMA DE CHAVE PÚBLICA OU ASSIMÉTRICA.....	29
FIGURA 7 ARQUITETURA DO PGP.....	37
FIGURA 8 PGPKEYS.....	50
FIGURA 9 KEY GERENATION WIZARD (WELCOME).....	51
FIGURA 10 KEY GERENATION WIZARD (EXPERT KEY PARAMET SELECTION)	
51	
FIGURA 11 KEY GERENATION WIZARD (PASSPHRASE ASSIGNMENT).....	52
FIGURA 12 KEY GERENATION WIZARD (KEY GERERATION PROGRESS).....	53
FIGURA 13 KEY GERENATION WIZARD (COMPLETING).....	53

FIGURA 14 CONFIGURAÇÃO DO PGP (ESCOLHENDO UM ALGORITMO SIMÉTRICO).....	54
FIGURA 15 CONFIGURAÇÃO DO PGP (SERVIDOR PÚBLICO DE CHAVES).....	55
FIGURA 16 ADICIONANDO UM NOVO SERVIDOR PÚBLICO DE CHAVES.....	56
FIGURA 17 FORMATO DE UMA CHAVE PÚBLICA NO PGPKEYS.....	57
FIGURA 18 MENU DE IMPORTAÇÃO DE CHAVE PÚBLICA NO PGPKEYS.....	57
FIGURA 19 SELEÇÃO DE CHAVE PÚBLICA.....	58
FIGURA 20 ASSINATURA DE CHAVE.....	58
FIGURA 21 CRIPTOGRAFANDO E VERIFICANDO UMA ASSINATURA DE UMA MENSAGEM RECEBIDA NO OUTLOOK 2000.....	60
FIGURA 22 DECRYPTOGRAFANDO E VERIFICANDO UMA ASSINATURA DE UMA MENSAGEM RECEBIDA NO OUTLOOK 2000.....	61
FIGURA 23 MENSAGEM EM FORMATO ORIGINAL.....	61
TABELA 1 OS CAMPOS DO CABEÇALHO RFC 822 RELACIONADOS AO TRANSPORTE DE MENSAGENS.....	61
TABELA 2 OS CABEÇALHOS RCF 822 INCLUÍDOS PELO MIME.....	61
TABELA 3 VERSÕES X ALGORITMOS DO PGP.....	61

LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS

AES	Advance Encryption Standard
DES	Data Encryption Standard
DSS	Digital Signature Standard
IDEA	International Data Encryption Algorithm
IMAP	Internet Message Access Protocol
ITAR	International Traffic in Arms Regulations

MD	Message Digest
MIME	Multipurpose Internet Mail Extensions
MIT	Massachusetts Institute of Technology
NIST	National Institute of Standard and Technology
NSA	National Security Agency
PGP	Pretty Good Privacy
POP	Post Office Protocol
RFC	Request For Comments
SHA	Secure Hash Algorithm
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol

RESUMO

O correio eletrônico (e-mail) é um dos melhores exemplos da onipresença que a Internet apresenta no cotidiano da sociedade moderna. Entretanto, muito embora já considerado um serviço essencial, o correio eletrônico é inerentemente inseguro. Apesar da ignorância da grande maioria dos usuários, o correio eletrônico, como utilizado comumente (através da RFC 822) não é capaz de garantir a entrega, a privacidade, a autenticação e o não repúdio das mensagens enviadas. A utilização de aplicações como o PGP, que permitem acrescentar propriedades de segurança ao correio eletrônico, têm-se demonstrado de mínimo grau de penetração junto à comunidade da Internet. Este trabalho tem portanto o objetivo de realizar um estudo sobre a implantação de propriedades de segurança em correio eletrônico através de uma aplicação baseada em criptografia, cujo representante mais popular é o PGP. A partir dos resultados obtidos com este estudo, espera-se poder divulgar e fornecer uma trilha para servir de guia à popularização dos serviços de segurança em correio eletrônico.

Palavras-chave: Segurança, correio eletrônico, PGP.

1 INTRODUÇÃO

Com a popularização da Internet na sociedade moderna, a aplicação que mais se destacou foi o correio eletrônico. O correio eletrônico (e-mail) por ser de fácil utilização e de entrega rápida ao remetente, vem sendo utilizado comumente para substituir contatos telefônicos e papéis no envio de textos por cartas através do correio convencional, reduzindo custos.

Pelo fato do correio eletrônico ser utilizado cada vez mais pela sociedade, devemos considerar o fator de segurança, o qual freqüentemente não é relevado pelo usuário.

O correio eletrônico é muito vulnerável; pois quando enviamos uma mensagem, a mesma passa por vários computadores e por diversas redes até chegar ao seu destinatário, podendo ser interceptada por um concorrente ou uma pessoa mal intencionada, quebrando a privacidade. Outro problema que pode ocorrer, por exemplo, é um “espião” passar por um funcionário de uma determinada empresa no envio de uma mensagem e solicitar um documento importante, acarretando assim um grave problema, dependendo do grau de importância do documento recebido pelo espião. Por este e vários outros motivos, a adoção de um sistema de segurança é muito importante. Mas esta adoção vem sendo demonstrando de mínimo grau de penetração na sociedade.

A segurança utilizada no correio eletrônico se baseia nas seguintes propriedades: **privacidade** diz respeito ao fato de somente o destinatário poder ler as mensagens; **autenticação** refere-se ao fato da mensagem ser de realmente quem diz ser; **integridade** refere-se ao fato da mensagem não ter sido alterada.

Este trabalho trata do estudo referente a como implantar um sistema baseado em criptografia, cujo representante mais popular é o PGP, para que o correio eletrônico possa conter propriedades de segurança.

O trabalho é subdividido em quatro definições; no segundo capítulo é feito um levantamento bibliográfico, caracterizando o ambiente do correio eletrônico e a criptografia, método utilizado para garantir as propriedades de segurança no envio das mensagens. No terceiro capítulo são mostrados todas as características do sistema PGP, para que possamos ter uma visão de como o PGP trabalha. No quarto capítulo é relatado um teste de interoperabilidade entre o criptosistema PGP e GnuPGP. E para finalizar no quinto capítulo é mostrado como utilizar o PGP, de modo o implanta-lo num ambiente de trabalho.

2 REVISÃO BIBLIOGRAFICA

2.1 CORREIO ELETRÔNICO

O material apresentado nesta seção é baseado nas referências (Tanenbaum, 1997) do livro **Redes de Computadores** e o livro **Usando E-mail na Internet** (Sadler, 1996). A seguir apresentaremos uma breve definição sobre o correio eletrônico e suas funcionalidades.

2.1.1 Definição

O correio eletrônico, o tão propalado e-mail (eletronic mail), iniciou-se na década de 1970, quando surgiram os primeiros sistemas de e-mail na ARPANET (a qual foi base para a Internet). Pode-se dizer então que o e-mail cresceu junto com a Internet, sendo uma das aplicações mais utilizadas até hoje neste ambiente.

O correio eletrônico é o meio pelo qual torna-se possível mandar e receber uma informação eletrônica através da Internet. Estas informações são conhecidas como mensagens. Os primeiros sistemas de correio eletrônico eram simplesmente formados por protocolos de transferência de arquivos, com a convenção de que a primeira linha de cada mensagem contivesse o endereço do destinatário. À medida em que o correio eletrônico foi se destacando na Internet foram surgindo novas implementações que melhoraram a forma de transmissão e o formato das mensagens (itens que veremos nas próximas seções).

2.1.2 Estrutura

Os sistemas de correio eletrônico consistem de dois subsistemas: os **agentes de usuário** são programas locais que interagem com o sistema de correio eletrônico, permitindo às pessoas ler e enviar mensagens; e os **agentes de transferência** de mensagens que são programas que rodam em segundo plano nos servidores com o objetivo de mover as mensagens da origem para o destino.

2.1.3 Funcionalidades

O correio eletrônico fornece cinco funções básicas: composição, transferência, geração de relatórios, exibição e disposição.

A **composição** (feita pelos agentes de usuários) se destina a elaboração de mensagens e respostas, também facilitam as respostas recebidas, preenchendo os campos automaticamente no cabeçalho da mensagem.

A **transferência** (feita entre os agentes de usuários e os agentes de transferências) permite o movimento de mensagens da origem ao destino. Isto é feito através de uma conexão com o destino ou com alguma máquina intermediária, a transmissão de uma mensagem e o encerramento da conexão.

A **geração de relatórios** (feita pelos agentes de transferências) se refere a exibir o que aconteceu antes, durante e após a transmissão da mensagem a sua origem, mostrando se ela foi entregue ou se perdeu no envio, esta é uma informação importante para a pessoa que enviou a mensagem.

A **exibição** (feita pelos agentes de usuários) das mensagens é importante porque permitir que a pessoa possa ler as mensagens que chegaram.

A **disposição** (feita pelos agentes de usuário) se refere a que a pessoa pode fazer após a mensagem ser lida.

2.1.4 Formato das Mensagens

Após a breve definição apresentada sobre o sistema de correio eletrônico e o que este pode oferecer, apresentaremos o formato das mensagens que são enviadas através da Internet.

Uma mensagem do correio eletrônico tem a mesma estrutura de uma carta a ser enviada por uma agência de correios. Ela é formada por um **cabeçalho** que equivale aos dados do envelope de uma carta e o **corpo do texto** que seria no caso de uma carta, o conteúdo de uma correspondência.

2.1.5 RFC 822

O primeiro formato padronizado de mensagem proposto pela ARPANET foi apresentado em 13 de agosto de 1982 na RFC 822 que define o formato de uma mensagem. Na Figura 1 mostra o formato de uma mensagem.

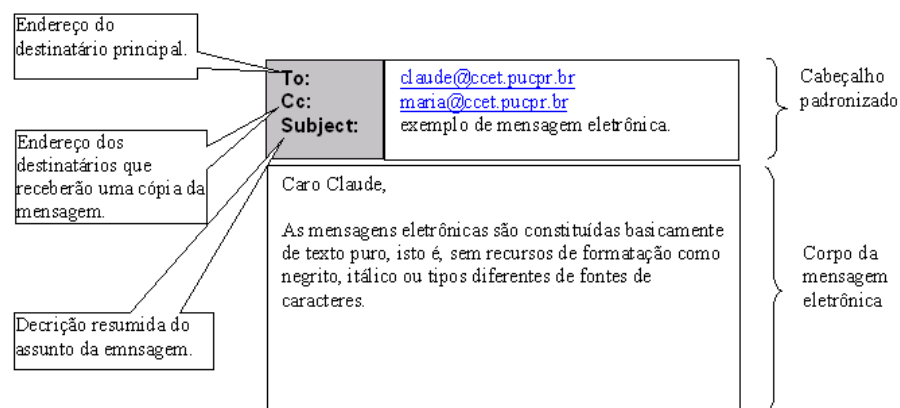


Figura 1 Formato de uma mensagem de Correio eletrônico

A maioria dos campos será utilizada para a transferência de uma mensagem, como por exemplo, para quem deve ser enviada a mensagem. Os cabeçalhos utilizados para a transmissão de uma mensagem estão representados na Tabela 1

Tabela 1 Os campos do cabeçalho RFC 822 relacionados ao transporte de mensagens.

Cabeçalho	Significado
To:	O(s) endereço(s) de correio eletrônico do(s) destinatário(s) principal(is)
Cc:	O(s) endereço(s) de correio eletrônico do(s) destinatário(s) secundário(s)
Bcc:	O(s) endereço(s) de correio eletrônico para cópias carbono ocultas
From:	A(s) pessoa(s) que criou(aram) a mensagem
Sender:	O endereço de correio eletrônico do remetente
Received:	A linha que é incluída por cada agente de transferência durante o percurso.
Return-Path:	Pode ser usada para identificar um caminho de volta ao remetente.

As mensagens da RFC 822 podem ter outros campos de cabeçalhos como os cabeçalhos: Date; Reply-To; Message-Id; In-Reply-To; References; Keyword; Subject. A RFC 822 também menciona que os usuários têm permissão para criarem outros campos de cabeçalhos, mas estes campos devem começar com X-. O formato da mensagem definida pela RFC 822 é formado primeiro pelos campos de cabeçalhos, depois uma linha em branco e finalmente o corpo do texto.

2.1.6 MIME

À medida em que o correio eletrônico foi tornando-se popular, a necessidade de se mandar arquivos (planilhas e programas executáveis, entre outros) pelo correio eletrônico surgiu, e o formato definido pela RFC 822 não os suportava, pois aplicava-se somente ao conteúdo das mensagens e não suportava a inclusão de um anexo em seu corpo, em formato texto, pois todas as mensagens enviadas pelo correio

eletrônico devem ser transmitidas em formato texto, descrito na RFC 821 que define o SMTP (protocolo de transmissão) que será estudado na seção 2.1.7. Deste modo foi criado o MIME, definido pela RFC 1341 e posteriormente atualiza na RFC 1521 que seria uma extensão do antigo formato proposto pela RFC 822, que forneceria uma identificação clara e distinta do conteúdo através de cabeçalhos de tipo de conteúdo.

O MIME acrescenta cinco cabeçalhos a mais, do que definida da RFC 822, visto na Tabela 2.

Tabela 2 Os cabeçalhos RCF 822 incluídos pelo MIME.

Cabeçalho	Significado
MIME-Version:	Identifica a versão do MIME
Content-Description:	String legível que identifica o conteúdo da mensagem
Content-Id:	Identificador exclusivo
Content-Transfer-Encoding:	Como o corpo da mensagem está codificado para transmissão
Content-Type:	Natureza da mensagem

Por exemplo, quando uma pessoa manda uma mensagem e anexa a ela uma imagem, a imagem é codificada em formato texto, e no cabeçalho **Content-Type** é definido o tipo e seu subtipo, no caso: image/gif.

O cabeçalho **Content-Transfer-Encoding** criada pelo MIME especifica também que tipo de codificação foi feito a mensagem.

A Figura 2 mostra o conteúdo de uma mensagem no formato da RFC 822 já com os campos utilizado pelo MIME.

```

T: <gulira@yahoo.com>
Subject: teste
Date: Mon, 7 Jul 2003 11:58:57 -0300
Message-ID: <KNEFIJKCGBHNBDMLKDEIGENDCAA.amhil@vnet.com.br>
MIME-Version: 1.0
Content-Type: text/plain;
      charset="iso-8859-1"
Content-Transfer-Encoding: 8bit
X-Priority: 3 (Normal)
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook IMO, Build 9.0.2416 (9.0.2910.0)
Importance: Normal
X-MimeOLE: Produced By Microsoft MimeOLE V5.00.2919.6700

Rodrigo Guedes de Souza ICC# 171751371
Programador/Analista de Suporte
S.O.S - Soluções Otimizadas em Saúde
E-mail: sos.amhil@vnet.com.br
Site: http://www.amhilkit.net

---
Outgoing mail is certified Virus Free.
Checked by AVG anti-virus system (http://www.grisoft.com).
Version: 6.0.489 / Virus Database: 288 - Release Date: 10/6/2003

```

Figura 2 Uma mensagem no formato RFC 822 / MIME enviado pelo agente de usuário (Microsoft Outlook, versão: 9.0.0.2814)

2.1.7 Transferência de Mensagens

Nesta seção abordaremos como as mensagens são transferidas do remetente até o destinatário.

O protocolo utilizado como padrão nos sistemas de correio eletrônico é o chamado SMTP, definida na RFC 821. Este protocolo tem a funcionalidade de enviar uma mensagem da origem até o seu destino.

Quando foram criadas as RFC 821 (SMTP) e RFC 822 (formato de mensagem) o correio eletrônico era baseado em uma conexão síncrona, pois o SMTP pressupõe de uma conexão online. Isto quer dizer que quando uma pessoa enviava uma mensagem para um determinado destino, estabelecia-se uma conexão TCP na porta 25 direta da máquina de origem, e após estabelecido a conexão era enviada a mensagem. Isto requeria que as duas máquinas estivessem on-line e com os clientes de e-mail abertos. Veja a figura 3.

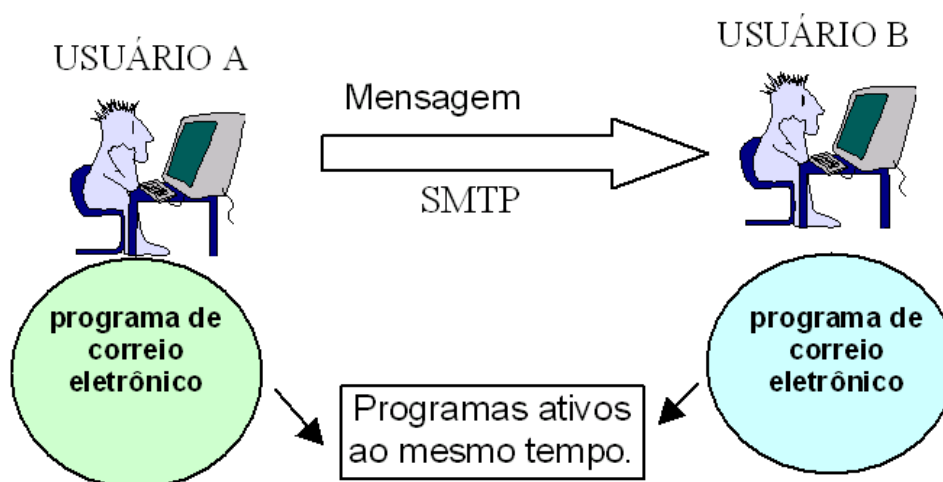


Figura 3 O SMTP pressupõe comunicação on-line entre as máquinas

A transmissão de uma mensagem utilizando-se o SMTP é feita totalmente em texto ASCII, ou seja, todas as mensagens que trafegam na rede podem ser vistas em texto plano.

2.1.8 POP

Com o passar do tempo os sistemas de correio eletrônico permitiram que os usuários não precisassem estar on-line para receber as mensagens. Suas mensagens eram armazenadas na caixa de e-mail. Assim surgiu o protocolo POP definido na RFC 1225, que permite aos usuários obter mensagens de mailbox remotas e armazená-las na máquina local do usuário.

Nesta nova estrutura, os sistemas de correio eletrônico começaram a comunicar-se de forma assíncrona, por intermédio de servidores de correio eletrônico.

A Figura 4 mostra que quando o cliente A manda uma mensagem ao cliente B, o programa de correio eletrônico do cliente A, se comunica via SMTP ao seu servidor de correio eletrônico A1 e ele também se comunicava via SMTP ao servidor de correio eletrônico B1, e num certo momento quando o usuário B quisesse obter as

mensagens, ele se comunicava via protocolo POP porta 110 ao servidor B1 e fazia o recebimento das mensagens.

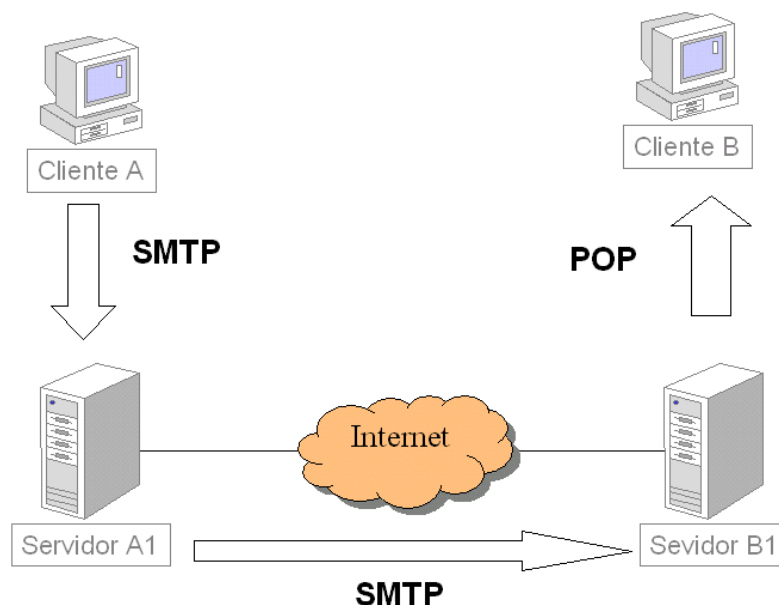


Figura 4 Transferência de Mensagens

O protocolo POP é semelhante ao SMTP: utiliza-se também de texto ASCII.

2.1.9 IMAP

O IMAP foi desenvolvido em 1986 na Universidade de Standford, nos Estados Unidos. Levou quase uma década para chamar a atenção da indústria, que nos anos 90 percebeu as reais vantagens do produto. As vantagens que se tem em relação ao protocolo POP são muitas: o tempo de conexão e o uso de recursos do servidor são mínimos, permite acesso interativo a múltiplas caixas postais a partir de múltiplos clientes, sem falar na habilidade de usar computadores diversos em diferentes momentos, já que nenhum dado fica armazenado na máquina do usuário. O acesso acontece independente da plataforma de leitor de e-mail . Através do protocolo IMAP é possível gerenciar múltiplas caixas postais, incluindo a facilidade de criar, editar, renomear e remover pastas diferente no servidor. Além disso, o IMAP também suporta atualização de pastas compartilhadas. Este recurso é útil quando vários usuários

estão processando as mensagens que são endereçadas para uma mesma caixa entrada.

2.2 CRIPTOGRAFIA

Esta seção se baseia na literatura: **Redes de Computadores** (Tanenbaum, 1997), **Usando E-mail na Internet** (Sadler, 1996) e **Criptografia e Segurança, O Guia Oficial RSA** (Burnett, Paine, 2002). Serão abordados os conceitos de criptografia, para que o leitor tenha o conhecimento desta arte de “ocultar” informações, pois a aplicação PGP, que é objeto deste trabalho baseia-se nestes conceitos que veremos a seguir.

2.2.1 Definição

A criptografia é arte de ocultar informações para que pessoas indevidas não a tenham. A criptografia é formada por três elementos: **criptografia** é quando ocorre o embaralhamento do texto da mensagem para que fique difícil o entendimento do texto para quem não tenha a chave. Quando aplicamos a criptografia em um texto (ou chamado texto plano), falamos que o texto plano está criptografado; **algoritmo de criptografia** é o método utilizado para fazer a criptografia do texto plano para o texto cifrado. Os algoritmos de criptografia são formados por regras que dificultem a leitura do texto plano; **Chave** é o elemento que interage com o algoritmos de criptografia para que haja a possibilidade de diversas combinações de criptografia do mesmo texto plano. A chave tem a função muito importante, porque se uma pessoa não autorizada a ler a mensagem, souber a chave, o remetente poderá escolher uma nova chave, mudando-se assim o texto cifrado. A figura 5 mostra a aplicação dos elementos da criptografia.

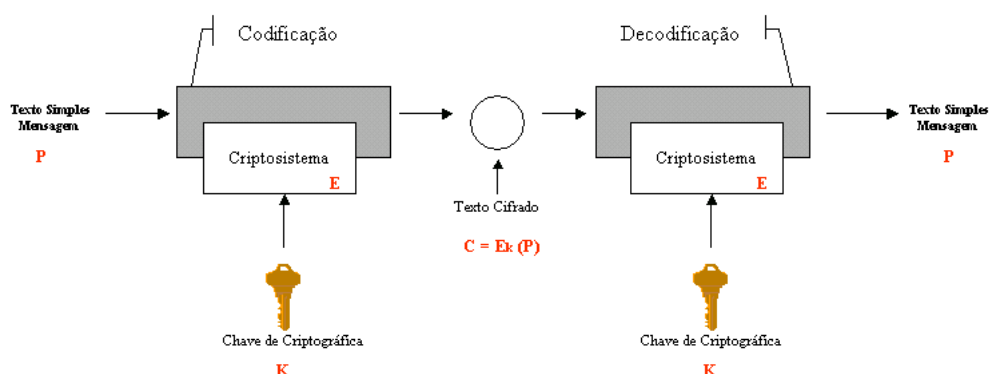


Figura 5 O modelo de Criptografia

A aplicação da criptografia vem de longa data, sua utilização geralmente é por grupos militares. Ela é conhecida desde a época de César, que usava uma técnica de criptografia simples quando enviava os mensageiros para retransmitir ordens para os generais nos campos de batalhas. Se os mensageiros fossem capturados ou subornados, as ordens não poderiam ser entendidas pelos inimigos. Os grupos militares foram os responsáveis mais importantes para o desenvolvimento desta tecnologia.

Nas seções 2.2.2 e 2.2.3 veremos os dois tipos de criptografias existentes.

2.2.2 Algoritmos Simétricos

Os algoritmos simétricos ou algoritmos de criptografia de chave secreta são aqueles que se utilizam da mesma chave de criptografia para recuperar o texto plano. Os primeiros sistemas de criptografia baseiam-se nesta regra.

Os algoritmos simétricos são chamados de criptografia de chave secreta porque o remetente e o destinatário devem manter em segredo a chave, pois somente com esta

chave poderá recuperar o texto normal. Este sistema oferece a privacidade do conteúdo de uma mensagem.

A principal vantagem deste tipo de criptosistema é a velocidade de ciframento de um texto plano.

O grande problema encontrado nos algoritmos simétricos é o gerenciamento eficiente da chave. Para que este criptosistema funcione eficientemente você terá que definir junto com a pessoa que deseja compartilhar mensagens cifradas a chave. Este processo terá que ser feito por um canal seguro (por exemplo uma conversa pessoal) pois se alguém interceptar a chave através de um meio inseguro como a Internet, telefone ou correio postal, as mensagens poderão ser decifradas ou o interceptador poderá se disfarçar como um dos integrantes da troca de mensagens, podendo-se assim quebrar a segurança proporcionada pela criptografia simétrica.

A aplicação de um sistema simples de criptografia simétrica podemos ver na Figura 5, a partir desta figura definimos matematicamente que para criptografar um texto plano (**P**) temos $C = E_k(P)$, onde (**C**) é o resultado da criptografia (Texto cifrado), (**E**) é o algoritmo simétrico utilizado e (**k**) é a chave. Para que possamos recuperar a mensagem utilizamos $P = E_k(C)$, assim podemos notar realmente que a chave (**k**) é utilizada tanto para cifrar como para decifrar uma mensagem, e o segredo desta chave é essencial.

Os algoritmos modernos são baseados nas mesmas idéias básicas das tradicionais, mas sua ênfase é diferente. Hoje as pessoas que criam a criptografia tem o objetivo de se tornarem os algoritmos de criptografia mais complexos para dificultar o trabalho do criptoanalista (pessoas que tentam quebrar os algoritmos de criptografia).

Entre os algoritmos modernos os mais conhecidos e utilizados hoje são o DES, IDEA e o AES.

2.2.3 Algoritmos Assimétricos

Para minimizar o problema do gerenciamento das chaves nos algoritmos simétricos entre as pessoas, na década de 70 surgiu a idéia de que a chave para criptografar a mensagem poderia ser diferente da chave para descriptografar a mensagem. Assim uma chave poderia ser distribuída para diversas pessoas (chave pública) e a outra chave teria que ser conhecida somente por uma pessoa (chave privada). A partir desta idéia surgiram os algoritmos assimétricos ou chamados de criptografia de chave pública.

A grande vantagem que os algoritmos assimétricos têm sobre os algoritmos simétricos é que o remetente não precisa entrar em contato com seu destinatário para definir a chave, pois o remetente poderá utilizar a chave pública de seu destinatário para criptografar a mensagem, deste modo somente o destinatário poderá recuperar a mensagem enviada pelo remetente porque somente a chave privada poderá decifrar o texto.

O problema que pode ocorrer com a utilização de algoritmos assimétricos é em relação a distribuição das chaves públicas, porque o que pode acontecer é se você pegar uma chave pública de uma pessoa que na realidade não é quem diz ser, você poderá estar trocando informações importantes com seu inimigo ou seu concorrente.

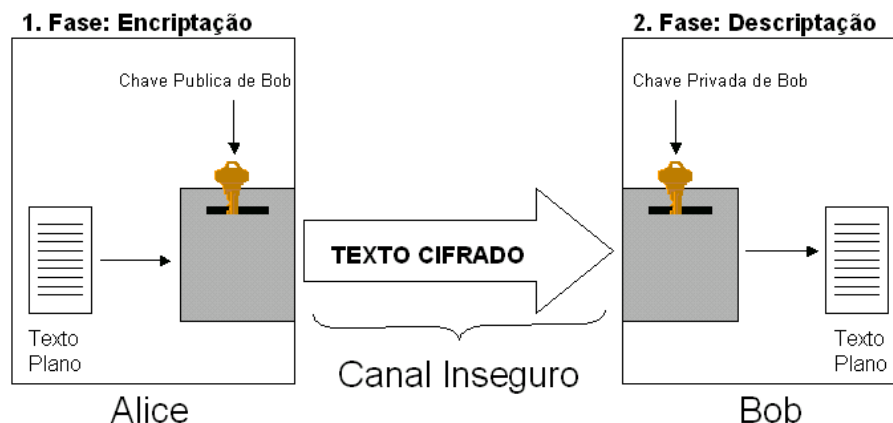


Figura 6 Sistema de Chave Pública ou Assimétrica

A Figura 6 exemplifica o processo de utilização do sistema de chave pública, o primeiro passo é, Alice criptografa a sua mensagem com a chave pública de Bob, esta chave pública Alice tem certeza que é de domínio de Bob, então ela envia para Bob. Segundo passo Bob, recebe a mensagem de Alice e descriptografa a mensagem enviada de Alice. Caso alguma pessoa intercepte a mensagem não poderá descriptografar, porque não tem a chave privada. Más como saber que a mensagem veio realmente de Alice? Para isto se utiliza uma assinatura digital que veremos sua aplicação no PGP.

2.3 CONSIDERAÇÕES FINAIS

O grande problema hoje no correio eletrônico é a falta de privacidade que o correio eletrônico possui. Como vimos anteriormente, tanto o protocolo SMTP como o POP trabalham com texto ASCII para as mensagens, e com a estrutura do sistema de correio eletrônico na Internet hoje, descrita na Figura 4, qualquer pessoa pode ler uma mensagem.

Alem das mensagens de correio eletrônico, serem transmitidas em texto visível, é muito fácil, a mensagem ser interceptada por uma pessoa não autorizada, porque uma

mensagem ao chegar ao servidor de correio eletrônico do destinatário pode passar por varias redes.

3 PGP

Neste capítulo será apresentado ao leitor a ferramenta PGP baseada em criptografia, para que possamos fazer a implantação de propriedades de segurança no correio eletrônico.

3.1 HISTÓRICO

O PGP, sigla de Pretty Good Privacy (em português “Privacidade Muito Boa”), criado por Philip Zimmermman: um ativista que quando jovem, sempre esteve em controvérsias principalmente com o governo. A idéia do projeto PGP começou em 1977 quando ele se interessou pelo algoritmo RSA, recentemente criado na época, e decidiu implementar esta idéia em computadores pessoais. Em 1984 ele começou realmente a trabalhar no projeto, e em 1986 já tinha algumas partes do projeto pronto. Somente em 1991 ele liberou a primeira versão (PGP 1.0) para o mundo todo através de BBSs via Internet, tornando-se assim uma das principais ferramentas de segurança criptográfica de correio eletrônico oferecendo privacidade, autenticação, assinaturas digitais e compactação de uma forma fácil de usar.

De acordo com Philip Zimmermman o principal motivo de distribuir de forma gratuita para o mundo todo, foi devido a um projeto de lei n.º 266 anti-crime criada pelo senado americano em 1991 que propõe que fabricantes de equipamentos de comunicação segura a inserir “armadilhas” para que o governo pudesse ler mensagens criptografadas de qualquer pessoa quando apropriadamente autorizados por lei. Deste modo Zimmermman tinha a idéia de tornar o PGP bastante popular o que criaria uma pressão maior por parte da população civil e de grupos industriais contra a promulgação de tal lei.

Sua disponibilização na internet foi surpreendente, pois se espalhou pelo mundo inteiro em pouco tempo. Pelo motivo do sistema estar se espalhando fora dos Estados Unidos, o governo através do ITAR montou um processo em oposição a Philip Zimmermann contra exportação de criptografia pesada, pois o ITAR proíbe exportação de software criptográfico sem uma licença, o que é quase impossível de se ter.

Outro fator que causou problemas a Philip Zimmermann é que ele foi acusado de infringir regras de patente do RSA, fato que foi regulamentado na versão 2.5 do PGP onde se utilizava uma biblioteca (rotinas criptográficas “coração da criptografia de chave pública”) chamada RSAREF que é freeware sob responsabilidade da RSA Data Security Inc. e fornecida no PGP distribuído pelo MIT e tinha certas restrições. Deste modo surgiu o projeto PGPi que seria uma versão alternativa às versões americanas do PGP, mas esta versão do PGPi, somente poderiam ser utilizada por pessoas que não moram ou não se localizam dentro dos Estados Unidos. A versão PGPi utiliza uma outra biblioteca chamada MPILB que tem as mesmas rotinas da RSA encontradas na RSAREF mas escritas por Philip Zimmermann. Segundo Philip Zimmermann em certos casos a utilização do MPILB era mais rápida do que a RSAREF.

3.2 ALGORITMOS UTILIZADOS

O PGP é um sistema híbrido que utiliza algoritmos de criptografia de chaves secretas e algoritmos de chaves públicas. Ele utiliza de algoritmos criptográficos que já existem há mais de uma década que são comprovados como seguros até hoje por especialistas, sendo assim os desenvolvedores do PGP não precisaram criar novos. Os algoritmos utilizados pelo PGP são conhecidos mundialmente pelas suas credibilidades em relação a propriedades de segurança que exige.

Os algoritmos de chaves secretas no PGP têm a função de manter a privacidade das mensagens, em quanto as de chave pública, são utilizadas para fazer autenticação e

não repúdio de uma mensagem junto com algoritmos de assinaturas (hash) para cumprir a integridade da mensagem.

Os algoritmos utilizados nas versões anteriores à versão 5.0 se baseiam fortemente no IDEA como simétrico, RSA como assimétrico e MD5 como o hash. Em quanto às versões posteriores à versão 5.0 são utilizadas de diversos algoritmos, sendo que para os algoritmos simétricos foram implementados: CAST, TRIPLE DES, AES e TWOFISH, e nos assimétricos foi implementado DH em conjunto com DSS.

Nas próximas seções, serão apresentados os principais algoritmos criptográficos usado no decorrer do projeto PGP.

3.2.1 Algoritmo Simétrico

O IDEA foi desenvolvido em 1990 por dois pesquisadores da Suíça James L. Massey e Xuejia Lai, é um algoritmo que trabalha com chave de 128 bits. Para quem não tenha contradições com o governo este é um ótimo algoritmo a ser utilizado pois não teve influência da NSA a respeito das “armadilhas”(Tanenbaum, 1997). Ele é utilizado a partir da versão 2.0, o único obstáculo que se tem para utilizar este algoritmo é que a Ascom Sytem mantém a patente de seu projeto, assim não é disponível para qualquer pessoa livre de royalties.

O CAST (referente aos seus criadores Carlisle, Adams, Stafford e Tavares) é um recente algoritmo de cifragem em blocos que utiliza uma chave do tamanho de 128 bits, criado por pesquisadores da Northern Telecom. A Northern fez a patente do CAST mas fizeram um compromisso por escrito para disponibilizar a qualquer pessoa a utilizá-lo. Philip Zimmermann vem tendo o mesmo grau de confiança com CAST quando implantou o IDEA há tempos atrás (Network Associates Inc, 1998). O CAST é utilizado a partir da versão 5.0.

O Triple DES é um substituto do DES desenvolvido pelos pesquisadores da IBM que utilizava uma chave de 56 bits. O Triple DES como o nome já diz realiza três vezes o algoritmo DES. Nelas utiliza-se duas chaves, cada uma delas contém 56 bits. Essencialmente, isso é a mesma coisa que utilizar uma chave de 112 bits (Network Associates Inc, 1998). A vantagem de se utilizar este algoritmo é que ele não é retido por qualquer patente e sua desvantagem é que existem boatos que o DES (a base para TRIPLE DES) tem uma “armadilha” utilizada pelo governo dos Estados Unidos para quebrar o algoritmo, deste modo a privacidade da mensagem pode estar comprometida. Ele foi utilizado a partir da versão 5.0.

O AES foi proposto em 1997 pelo NIST dos Estados Unidos, através de um projeto convidando a comunidade mundial especializada no assunto, para substituir o DES que era utilizado pelo governo como padrão e que estava ultrapassado, o algoritmo escolhido foi o “Rijndael” criado pelos belgas Joan Daemen e Vicente Rijmen e assessorado pelo criptografo brasileiro Paulo Barreto (Gonçaves; Ribeiro, 2001). Este algoritmo gera chave de 128, 192 e 256 bits.

O Twofish que utiliza uma chave de 128 bits para a criptografia, foi desenvolvido por Bruce Schneier, John Kelsey, Doug Whithing, David Wagner, Chris Hale e Nels Ferguson para que fosse o sucessor do algoritmo de cifragem em blocos chamado de Blowfish que trabalha com chave de 56 bits (Ribeiro, 2001). O Twofish foi um dos algoritmos que participou do projeto AES criado pelo NIST, ele foi submetido à teste atendendo aos critérios requeridos, assim ele tornou um dos finalistas do projeto que seria adotado pelas agências federais dos EUA, mas infelizmente perdeu. Este algoritmo foi introduzido no PGP a partir da versão 7.0. Ele foi selecionado porque antes de escolher o Rijndael, o Twofish foi muito bem falado pelos especialistas.

3.2.2 Algoritmo Assimétrico

O RSA criado em 1978 pelos pesquisadores do MIT por Ron Rivest, Adi Shamir e Len Adleman foi utilizado como único até a versão 2.6.3, após esta versão foi implementado outros algoritmos, o principal motivo pela mudança foi a patente RSA.

O Diffie Hellman criado em 1976 por Whitfield Diffe e Martin Hellman foi o primeiro sistema assimétrico a ser publicado. Este algoritmo se baseia na distribuição da chave privada através de um meio inseguro baseado na dificuldade de se calcular logaritmos discretos.

O DSS foi criado em 1992 pelo NIST como padrão para as assinaturas digitais.

3.2.3 Funções de Hash

O Hash é um método de autenticação pelo qual, verificamos a integridade de um arquivo. O procedimento é simples por meio de funções específicas um conjunto de letras e números de tamanho fixos são associados ao arquivo. Para ter certeza que o arquivo é o mesmo, executa-se o programa e gera-se a chave, se não ocorreram mudanças no conjunto de “hash”, o arquivo não foi alterado. Fazendo uma analogia podemos dizer que o Hash equivale a um código verificador de erros, e representa compactamente a mensagem, usada para detectar mudanças em seu conteúdo.

O SHA é uma função hash desenvolvido pelos pesquisadores da NIST , baseado no MD4 em 1994. Este algoritmo gera um resumo de 160 bits. A principal vantagem de se utilizar o SHA em vez do MD5 é que ele gera o resumo maior que o MD5. O SHA foi implantado no PGP a partir da versão 5.0

O MD5 foi criado em 1992 por Ron Rivest do MIT. Esta função hash gera um resumo de 128 bits. O MD5 foi implantado no PGP a partir da versão 2.6.3

Tabela 3 Versões x Algoritmos do PGP

Versão	Algoritmo Simétrico	Algoritmo Assimétrico	Algoritmo Hash
Anterior a 2.6.3	DES	RSA	MD4
2.6.3	IDEA	RSA	MD5
5.0	IDEA, TRIPLE DES e CAST	RSA e DH-DSS	MD5 e SHA
6.0	IDEA, TRIPLE DES e CAST	RSA e DH-DSS	MD5 e SHA
7.0	IDEA, TRIPLE DES, CAST, AES e TWOFISH	RSA e DH-DSS	MD5 e SHA
8.0	IDEA, TRIPLE DES, CAST, AES e TWOFISH	RSA e DH-DSS	MD5 e SHA

A Tabela 3 mostra os algoritmos utilizados por cada versão.

3.3 FUNCIONALIDADES

Nessa seção serão apresentadas na prática o funcionamento interno do PGP, utilizaremos como exemplo: IDEA para privacidade das mensagens, RSA para autenticação das mensagens e MD5 para a integridade das mensagens.

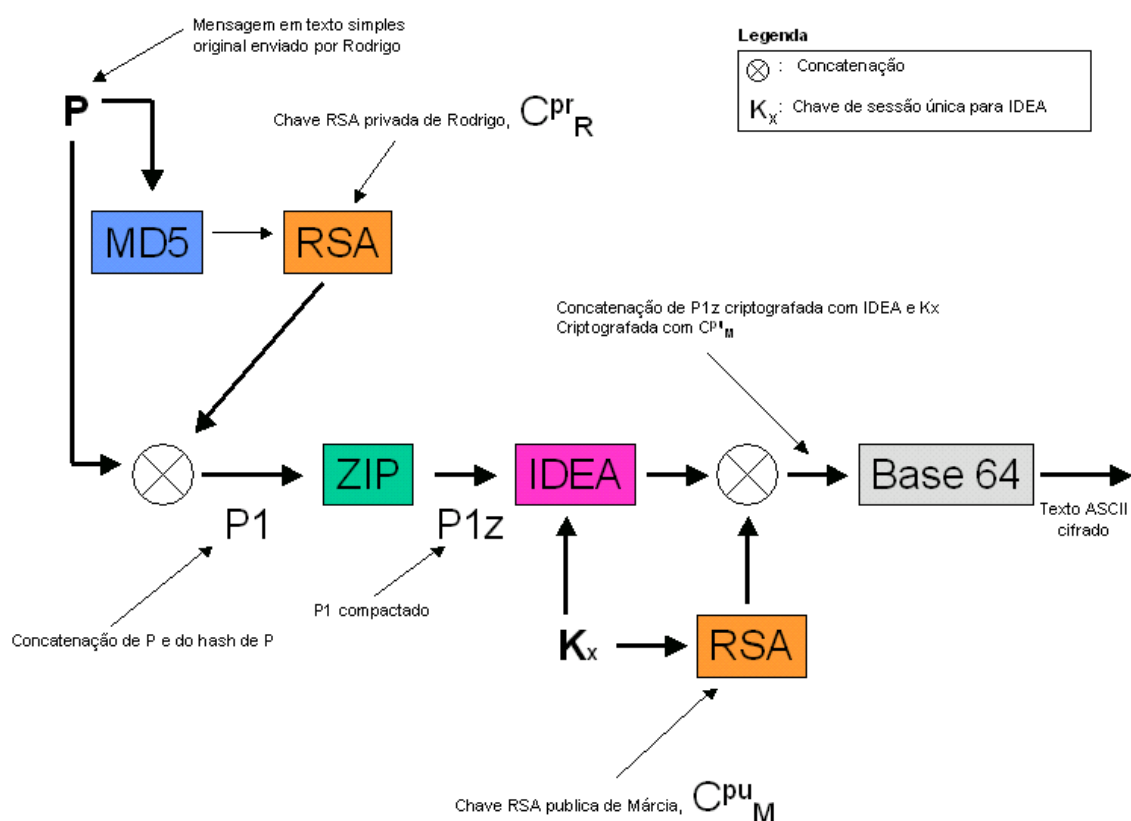


Figura 7 Arquitetura do PGP

Para mostrar a funcionalidade do PGP consideremos a figura 3.2 onde Rodrigo quer enviar uma mensagem para Márcia de uma forma que ninguém a não ser ela possa ler essa mensagem e que tenha certeza que a mensagem é realmente de Rodrigo, e também na transmissão da mensagem do computador de Rodrigo até seu computador (Márcia) não tenha sido alterada por uma outra pessoa. Para que o PGP funcione ambas partes Rodrigo e Márcia contenham um par de chaves RSA (no caso da figura 3.2) onde C^{pr}_x seja chave privada e C^{pu}_x seja a chave pública, e também, ambos conheçam a chave pública um do outro.

Rodrigo inicia o funcionamento do PGP invocando-o em seu computador, inicialmente o PGP aplica um hash utilizando MD5 em seu texto plano, depois de efetuado o hash, o PGP cifra com chave privada C^{pr}_R de Rodrigo. Deste modo quando Márcia receber a mensagem e aplicar a chave pública de Rodrigo, poderá verificar se o hash está correto. O texto plano é concatenado com o resumo do texto (hash) tornando-se assim

P1 e depois é compactado com o algoritmo de compactação de ZivLempel criado em 1977, produzindo **P1z**.

“Este processo tem duas vantagens: reduz o tamanho da mensagem que será comprimida e fortalece a criptografia, já que impede que um criptoanalista tente quebrar a codificação usando tabelas de palavras de dicionário” Cavalheiro (2003, p. 45).

Após a compactação o PGP cria uma chave **Kx**, que é gerada com base no movimento do mouse e de teclas que foram pressionadas, esta chave **Kx** tem o tamanho de 128 bits e será utilizada no IDEA para cifrar **P1z**. Deste modo percebemos que o IDEA forneceu privacidade a mensagem.

Segundo Tanenbaum (1997, p. 759) “Kx Chamada de Chave de sessão na literatura PGP; no entanto, essa denominação não é adequada, pois não há sessão”.

O PGP utiliza o RSA e cifra chave de sessão **Kx** com a chave pública de Márcia, assim somente ela poderá decifrar o texto gerado pelo IDEA. Após **P1z** ser cifrado pelo IDEA e a chave de sessão **Kx** ser cifrada com o RSA com a chave pública de Márcia C^{pu}_M , o PGP concatena ambas e transforma em BASE64. Este **texto ASCII** “resultado da conversão Base64” (como na figura 2.3) é formado apenas por letras, números, e os símbolos +, / e =, com a mensagem, assim pode ser colocado no campo da mensagem RFC 822 e chegar intacta a seu destino.

Quando Márcia receber a mensagem o PGP terá como primeiro passo: reverter a Base64 e separar **P1** (sobre ciframento IDEA com chave **Kx**) e **Kx** (sobre ciframento RSA com chave pública de Márcia). Tendo o segundo passo a descryptografia de **Kx** com a chave privada de Márcia C^{pr}_M , podendo-se assim decifra **P1z** (cifrado). Após obter **P1z** o terceiro passo é descompactá-lo, chegando a **P1**, e então o PGP faz o quarto passo, separando o texto plano do hash cifrado, o quinto e ultimo passo é feito após ser descryptografado o hash com a chave pública (RSA) de Rodrigo C^{pu}_R , em

seguida é verificado se o hash gerado pelo texto é o mesmo que o rash enviado por Rodrigo.

3.4 GERENCIAMENTO E DISTRIBUIÇÃO DAS CHAVES

3.4.1 Chaveiro do PGP

Todos os usuários que utilizam o PGP tem dois anéis de chaves: um anel de **chaves-privada** (private key ring) que ficam armazenadas localmente no arquivo secring.skr onde contem uma ou mais chaves pública/privadas. E um anel de **chaves-públicas** (public key ring) que são armazenadas as chaves públicas dos correspondentes do usuário e ficam armazenadas localmente no arquivo pubring.pkr.

A possibilidade do usuário estar criando mais de um par de chaves no chaveiro de **chave-privada** é devido que o usuário possa estar alterando suas chaves públicas periodicamente ou quando necessário, sem invalidar as mensagens que estiverem sendo preparadas (Tanenbaum, 1997).

O gerenciamento de chaves no PGP esta completamente ligado aos dois arquivos: secring.skr e pubring.pkr. O arquivo secring.skr ao qual contem os pares de chaves privadas/pública do usuário, podem ser importadas em um outro computador, por este motivo, o arquivo é cifrados através do algoritmo simétrico IDEA.

3.4.2 Forma de distribuição de chave

Após o usuário criar o par de chaves ele terá que distribui sua chave pública para que outras pessoas possam enviar mensagens codificadas e verificar sua assinatura. O

PGP fornece três tipos de distribuições de chaves públicas: Através de um **servidor público de chaves, e-mail ou exportação em arquivo**.

O servidor público de chaves é o melhor meio de tornar a chave pública disponível, porque qualquer pessoa poderá entrar neste servidor de chave e localizar a chave pública do usuário. Os endereços a seguir fornece serviços de armazenamento de chaves públicas gratuitamente: `ldap://keyserver.pgp.com`, `ldap://europe.keys.pgp.com:11370`.

O email pode ser utilizado também como meio de distribuição da chave pública do usuário, pois esta chave é composta basicamente por um bloco de texto.

A exportação de arquivo possibilita que o usuário possa levar em meios magnéticos o arquivo gerado pela exportação.

3.4.3 Validação de Chaves

Um problema que se tem em qualquer cripto-sistema de chave pública é verificar se a chave pública é realmente de quem deveria ser. Existem algumas formas de acabar com este problema, uma delas é que o destinatário forneça em mão a cópia de sua chave pública, esta troca de chave é inviável pois em certos casos os usuários estão localizados a muitos quilômetros de distância. Outra forma de validar a chave é através da impressão digital chamada de “fingerprint “ no PGP, esta impressão está presente como propriedade da chave pública, com esta impressão digital o remetente pode entrar em contato por telefone com o destinatário e pedir para ele falar a impressão digital de sua chave pública.

3.4.4 Teia de confiança

No PGP cada usuário é responsável pela geração e gerência de suas chaves públicas e privadas, as quais estão associadas a um identificador (tipicamente formado pelo e-mail do mesmo). Não existe uma autoridade ao qual comprove realmente o pertence da chave pública, em que todos devam confiar. No PGP indivíduos fazem a certificações de chaves públicas de outros de sua confiança, formando uma rede de chaves públicas individuais interconectadas por links formados por suas assinaturas, esta rede de chaves é chamada de teia de confiança.

Por exemplo: Rodrigo quer certificar a chave pública de Márcia para manter contato, um fato extremamente importante é que Rodrigo conhece pessoalmente Márcia, e tendo absoluta certeza que a chave pertence a ela, da mesma maneira Márcia certifica a chave pública de Lourdes. Deste modo Rodrigo mesmo não conhecendo Lourdes passará a existir um vínculo indireto de certificado. Esta cadeia de segurança de assinatura avalizada e certificada, com o tempo vai estabelecendo elos de segurança entre usuários que muitas vezes não se conhecem pessoalmente.

3.4.5 Revogando Chaves Públicas

Um usuário pode revogar sua chave pública corrente tanto pela desconfiança de que a respectiva esteja sob suspeita, ou simplesmente para evitar que uma chave seja utilizada por um longo período. É importante notar que uma suspeita sobre a confidencialidade da chave privada deve requerer do oponente a posse da chave privada decifrada, ou então de ambos: a chave privada cifrada e a frase-senha.

A convenção para se revogar uma chave pública é um certificado assinado pelo usuário. Esse certificado tem o mesmo formato de uma assinatura normal, porém inclui um indicador dizendo que o objetivo dele é o de revogar uma chave pública. O

usuário deve então propagar este certificado o mais rapidamente e amplamente possível de forma a habilitar correspondentes em potencial a atualizar as publings.

3.5 VULNERABILIDADES

Nenhum sistema de segurança de dados é impenetrável. O PGP pode ser afetado de diversas formas, o que deve ser analisado, em qualquer sistema é se a mensagem tem maior valor para o atacante do que o valor do ataque. A seguir descreveremos as principais vulnerabilidades que o PGP tem.

3.5.1 Comprometimento da Frase-senha e Chave Privada

Uma das formas mais simples de se quebrar o sistema é escrever a frase-senha em algum lugar onde uma pessoa indevida possa obtê-la, caso ela obtenha a frase-senha e o arquivo onde contenha as chaves privadas, este intruso pode assinar suas mensagens e fazer assinaturas em seu nome. Para que a frase-senha não seja descoberta por uma outra pessoa, o usuário não deve escrever apenas uma palavra na frase-senha, pois alguém poderá utilizar um software que faz busca em um dicionário para descobrir a palavra. O ideal seria criar uma frase que só fazem sentido ao usuário e que contenha espaços e números.

3.5.2 Falsificação da Chave Pública

A falsificação de chave pública é um grande problema para qualquer criptosistema de chave pública. Pode acontecer que alguém queira se identificar como outra pessoa, por isso, quando você for usar uma chave pública, tenha certeza que a mesma não foi

falsificada. A chave pública de um determinado indivíduo só pode ser utilizada se o usuário tiver obtido diretamente do proprietário ou se ela foi assinada por uma pessoa que o usuário tenha confiança.

3.5.3 Arquivos Não Apagados Completamente do Disco

Outro grande problema que pode ocorrer é da forma que o sistema operacional apaga os arquivos. Quando o usuário pede para o sistema operacional apagar um texto plano, o sistema operacional marca os blocos que foram apagados, mas não excluem o conteúdo, somente serão apagados quando estes blocos forem utilizados por um outro arquivo, sendo assim, estes dados “não excluídos” podem ser recuperados por software, do mesmo modo quando ocorre um problema físico no disco de armazenamento, existem técnicos especializados que recuperam estes dados, podendo assim estar também visualizando seu arquivos “não excluídos”. Uma solução que vem sendo adotado é o uso freqüente de desfragmentação no disco de armazenamento, deste modo o sistema operacional reorganiza o disco substituindo os blocos marcados, como apagados por outros.

3.5.4 Vírus e Cavalo de Tróia

Um outro ataque pode acontecer através de um vírus, que infecta o sistema operacional do usuário e assim capturar sua frase-senha e o arquivo de chaves públicas e envia-los ao desenvolvedor do vírus. A solução de defesa para este vírus poderia vir de software antivírus. O PGP não tem defesa contra vírus, então cabe ao usuário utilizar este software, para que não permita estas vulnerabilidades. Outra forma de espionagem envolveria uma cópia do PGP parecida visualmente com a original, mas alterada para que não funcionasse corretamente. Por exemplo, alguém poderia deliberadamente altera-lo para que não funcionasse as assinaturas

corretamente, permitindo a falsificação das mesmas. Esta forma de alteração do código fonte não é tão difícil, pois o mesmo é amplamente divulgado na Internet. A solução para este caso é fazer o download dos lugares oficiais.

3.5.5 Falhas de Segurança Física

O descuido do usuário com seu computador pode permitir que um usuário possa copiar arquivos originais ou mensagens impressas. Um adversário poderá utilizar meios como roubo, vasculhamento, violações físicas de segurança podem compromete-las. Este ataque é de baixo custo comparados aos ataques criptoanalíticos do PGP.

3.5.6 Ataques Tempest

Este tipo de ataque ocorre através da captura de sinais eletromagnéticos vindo do computador do usuário, são utilizado caminhões com equipamentos de auto custo para fazer este tipo de serviço mas mesmo assim o custo deste tipo de ataque é bem mais baixo do que criptoanalíticos. Estes equipamentos conseguem capturar imagens dos monitores e teclas pressionadas pelo usuário. Para evitar este tipo de ataque é utilizado blindagens nos equipamentos e cabos, para que estes sinais não sejam captados. Algumas versões lançadas após a versão 6.0 do PGP permitem que o texto decifrado possa ser exibido no monitor através de níveis reduzidos de emissão de frequência de rádios, este recurso é chamado de visualizador seguro, que dificulta a captação destes sinais.

3.5.7 Exposição de Sistemas Multiusuário

O PGP foi originalmente projetado para rodar em máquinas MSDOS mono-usuários, sob seu controle físico direto. Mas agora existem versões do PGP que também rodam em sistemas multi-usuários como UNIX e VAX/VMS. Nesses sistemas, os riscos de descobrirem sua frase-senha, chaves secretas ou mensagens são maiores. Um intruso esperto o suficiente ou o próprio administrador de rede poderiam ter acesso aos seus arquivos originais, ou talvez utiliza-se de algum programa especial para monitorar constantemente a digitação ou ver o que está aparecendo em sua tela. Os riscos reais de segurança depende de cada situação em particular. Alguns sistemas multi-usuários podem ser considerados seguros porque todos os usuários são confiáveis, ou porque não há interesse suficiente em espionar alguém. De qualquer modo, recomenda-se rodar o PGP de uma máquina isolada, em um sistema mono-usuário, diretamente sob seu controle físico.

3.5.8 Análise de Tráfico

Com a análise de tráfico, o atacante pode até não saber o conteúdo das mensagens cifradas, mas ele vai saber qual é o remetente e o destinatário e qual as horas do dia que são enviadas as mensagens. Para resolver este tipo de problema, precisaria de protocolos de comunicação especialmente desenvolvidos, para reduzir à exposição dessas análises, possivelmente com alguma assistência criptográfica.

3.5.9 Criptoanálise

Este é o mais caros dos métodos de ataque contra o PGP, para que possa fazer uma criptoanálise e descobrir a chave secreta de uma mensagem, o espião teria que ter um acesso a um supercomputador. Eles poderiam quebrar sua chave pública usando

algum novo método secreto de quebra de defesa. Mas a academia civil dos Estados Unidos têm intensivamente atacado a criptografia por chave pública sem sucesso em 1978. Talvez o governo americano tenha algum método confidencial para quebrar o algoritmo de criptografia convencional utilizado no PGP (IDEA). Mas algum otimista se justifica, pois os projetistas do algoritmo IDEA estão entre os melhores da Europa. Ele foi extensivamente analisado e revisado por alguns dos melhores cripto-analistas encontrado no mundo. Além disso, mesmo que estes algoritmos tivessem alguma fraqueza desconhecida, o PGP comprime o texto original antes de criptografá-lo, o que deveria reduzir essas fraquezas quaisquer que fossem. Concluimos que os gastos computacional para quebrar uma mensagem provavelmente seria maior do que o valor da própria mensagem.

4 INTEROPERABILIDADE (PGP v.s GNUPG)

Nesta capítulo será mostrado um relato do teste efetuado entre PGP e o GnuPG, com objetivo de determinar o grau de interoperabilidade entre o PGP e o GnuPG.

4.1 GNUPG

O GnuPG (GNU Privacy Guard) teve sua elaboração formada após o alemão Werner Koch participar de uma palestra feita por Richard Stallman em Achen, Alemanha 1997. Na ocasião Stallman convidava programadores europeus a desenvolverem um projeto GNU com um software relacionado à criptografia. Em abril daquele mesmo ano a patente do algoritmo Diffe-Hellman expirou-se. Após algumas semanas Werner se viu explorando um analisador PGP de pacotes de dados apenas por distração e pouco tempo depois já estava enviando mensagens com o PGP versão 2.0. Werner então entrou em contato com a FSF (Free Software Foundation) e comunicou-lhes que estaria começando a trabalhar em uma implementação livre do PGP, tornando-se futuramente conhecido como GnuPG (Pimenta, 2001).

O GnuPG é um software livre GNU desenvolvido sobre a licença GPL, este criptosistema foi desenvolvido sobre o padrão OpenPGP criado pela Network Associate Inc. descrito na RFC 2440 e não utiliza-se de algoritmos de criptografia patenteados. Desse modo qualquer pessoa poderá utilizar este software livre sem pagar por royalties. O GnuPG esta disponível em diversas plataforma: Mac, Unix, Linux, xBSD, Windows e OS/2 no site <http://www.gnupg.org>. Hoje na versão 1.2.3 o GnuPG suporta as seguintes algoritmos: ElGamal, DSA, RSA, AES, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160 e TIGER.

4.2 RELATO DO TESTE EFETUADO

Os testes foram efetuados no dia 02/09/2003 através dos usuários rgsonline@ig.com.br e edson@yanaga.com.br. Para este teste foi utilizado o sistema PGP pelo usuário (rgsonline@ig.com.br) na versão 7.0.3 disponível no site <http://www.pgpi.org> sobre o sistema operacional Windows 98 e o sistema GnuPG pelo usuário (edson@yanaga.com.br) na versão 1.2.3 disponível no site <http://www.gnupg.org> sobre o sistema operacional Red Hat Linux 9.

Em um primeiro instante foram distribuídos as chaves públicas de ambos os usuários (rgsonline@ig.com.br e edson@yanaga.com.br) através de e-mail, de forma que rgsonline@ig.com.br mandou sua chave pública para edson@yanaga.com.br e o mesmo enviou sua chave pública criptografada com a chave pública de rgsonline@ig.com.br havia tinha enviado. A chave do usuário rgsonline@ig.com.br tem as seguintes configurações (Algoritmo de chave pública: DS/HSS com chave de tamanho 2048 e o algoritmo para manter a privacidade: Twofish). Já a do usuário edson@yanaga.com.br tem as seguintes propriedades (Algoritmo de chave pública: RSA com chave de tamanho 2048 e o algoritmo para manter a privacidade: AES-256). Observamos que através deste tipo de distribuição de chaves públicas, poderia ocorrer interceptação por terceiros. Ou mesmo poderia haver a falsificação de uma mensagem, enviando uma chave pública falsificada para um ou para ambos usuários. Excluindo esta possibilidade, consideramos que ambos os usuários estejam com as chaves públicas corretas.

Apesar de o PGP e o GnuPG fornecerem a função de fingerprint para verificação das chaves públicas dos usuários, neste testes estes recursos não foram utilizados. O fingerprint deve ser utilizado sempre, pois é um recurso que facilita para ao usuário certificar-se que está importando a chave pública correta.

No segundo momento o usuário `rgsonline@ig.com.br` assinou a chave pública do usuário `edson@yanaga.com.br` fazendo assim a certificação da chave pública na Teia de Confiança do PGP.

Após ambos usuários terem feitos os processos anteriores, começaram a compartilhar mensagens. Nos testes realizados o sistema PGP utilizado pelo usuário `rgsonline@ig.com.br` conseguiu descriptografar as mensagens criptografadas pelo sistema GnuPG, também o mesmo ocorreu com o usuário `edson@yanaga.com.br`. Não houve nenhum problema durante os teste efetuados com as configurações dos pares de chaves citados no começo.

4.3 CONSIDERAÇÕES FINAIS

Existe uma restrição que deve ser vista antes de transmitir mensagens entre o sistema PGP e o GnuPG. De acordo arquivo FAQ que vem incluso no criptosistema GnuPG não são suportados mensagens criptografadas pelo PGP versão 2 por utilizar-se do IDEA, um algoritmo é patenteado. Isto foge da filosofia do GnuPG de se utilizar somente algoritmos não patenteados. Caso haja a necessidade de utilizar o PGP versão 2 junto com o GnuPG, estará disponível no endereço `ftp://ftp.gnupg.dk/pub/contrib-dk/` uma biblioteca com versões para o sistema operacional Linux e Windows que possibilitaram o GnuPG de trabalhar com algoritmo IDEA. Vale lembrar que existem países que proíbem a utilização deste algoritmo sem o pagamento royalties.

5 UTILIZAÇÃO PRÁTICA DO PGP

O material apresentado neste capítulo é baseado na guia do usuário do PGP referente a versão 7.0 (Network Associates Inc, 2001). O objetivo deste capítulo é capacitar o usuário do correio eletrônico a utilizar de forma básica o PGP de modo que possa configurar a ferramenta, e estar transmitindo e recebendo mensagens de forma seguras. O PGP a ser apresentado é a versão 7.0.3 que trabalha sobre as plataformas Windows 95, 98, ME, NT e 2000. Disponível no endereço [Http://www.pgpi.org](http://www.pgpi.org).

5.1 CRIANDO UM PAR DE CHAVES

Para que o usuário possa enviar ou assinar uma mensagem de e-mail, será preciso criar um par de chaves. Para criar o par de chaves siga os seguintes passos:

1.º Passo: Executar o PGP, clique em Iniciar – Programas – PGP – PGPKeys.

Deste modo será aberto o gerenciador de chaves (PGPKeys) como mostra na figura a baixo. As principais funcionalidades do PGP serão obtidas através deste programa.

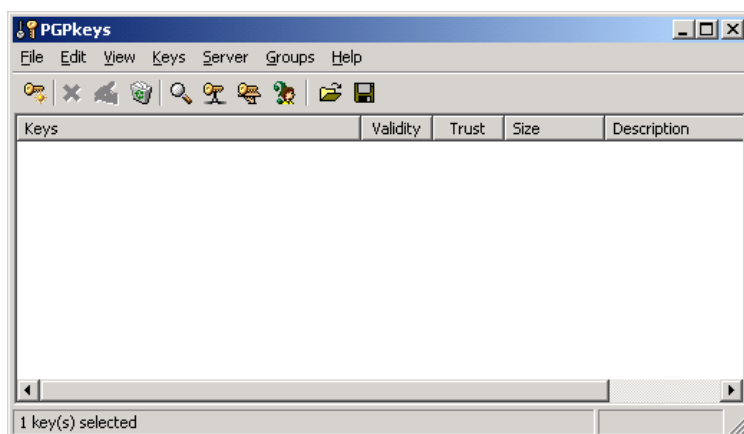


Figura 8 PGPKeys

2.º Passo: Criar o par de chaves, clique no menu Keys – News Keys...

Após este passo será aberto um assistente para a geração do par de chaves.



Figura 9 Key Gereneration Wizard (Welcome)

Para um melhor aproveitamento do PGP é recomendável criar o par de chaves através do modo detalhado, acessando através do botão Expert. Deste modo o usuário poderá escolher suas configurações.

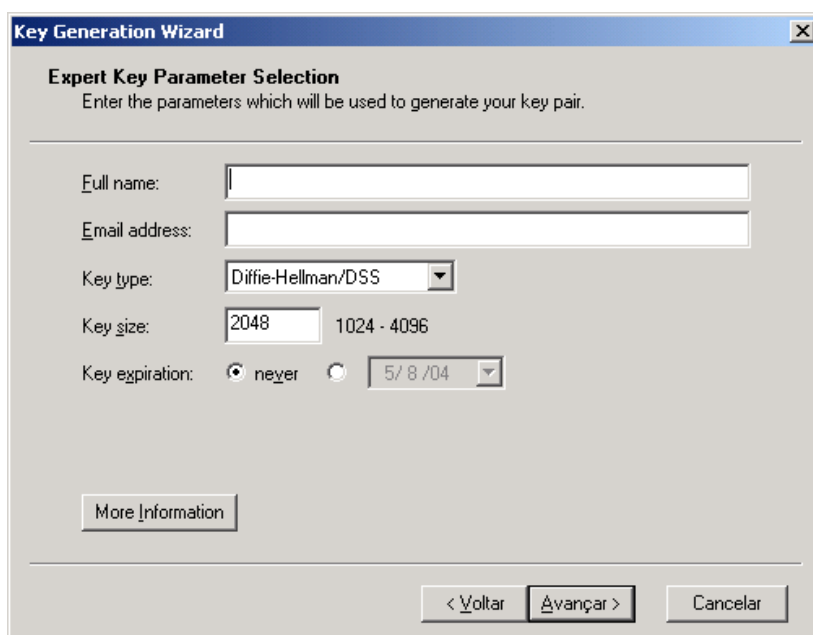


Figura 10 Key Gereneration Wizard (Expert Key Paramet Selection)

3.º Passo: O usuário deve informar no campo Full name o seu nome completo, E-mail address o endereço de e-mail utilizado para o envio da mensagem. Já o campo key type é referente ao tipo de algoritmo que o usuário irá utilizar, o PGP fornece três tipos de algoritmos de criptografia: O Diffe-Hellman/DSS, RSA e o RSA legacy. Estes algoritmos foram descritos na seção 3.2, o RSA legacy é utilizado quando o usuário trabalha com mensagens de versões antigas, esta opção foi incluída a partir da versão 5.0. Existe também o campo Key size que define o tamanho da chave a ser utilizada por estes algoritmos. Deve analisar bem o tamanho da chave a ser utilizada, pois quanto maior for o tamanho da chave, mais tempo levará para criptografar uma mensagem. O campo Key expiration pode ser utilizado quando usuário quiser determinar uma data de validade da chave, após esta data atendida a chave será invalidada. Então depois de configurado o usuário deverá clicar no botão Avançar.

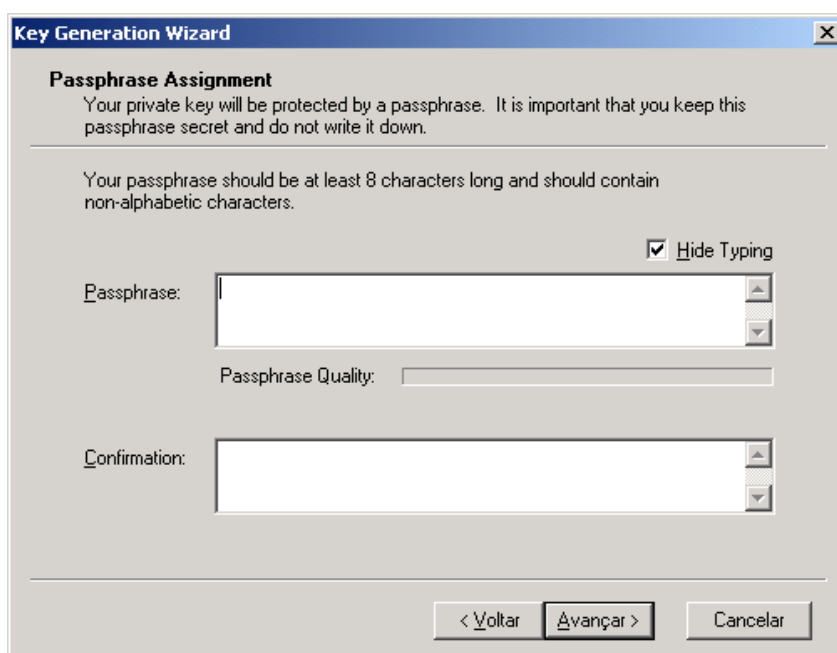


Figura 11 Key Generation Wizard (Passphrase Assignment)

4.º Passo: O usuário deverá escolher uma frase-senha na área passphrase e repetir o mesmo conteúdo na área confirmation. Sugerimos que quando o usuário for criar sua frase-senha ela não coloque apenas uma palavra, mas monte uma frase com espaços e se possível números. Existe uma barra de progresso nesta tela chamada de passphrase quality, que mostra para o usuário o nível de segurança que a frase-senha

tem, comparado se um usuário poderá deduzí-la ou decodificá-la através de tentativas por software. Após definido a frase-senha o usuário deverá clicar no botão avançar e então o PGP gera as chaves.

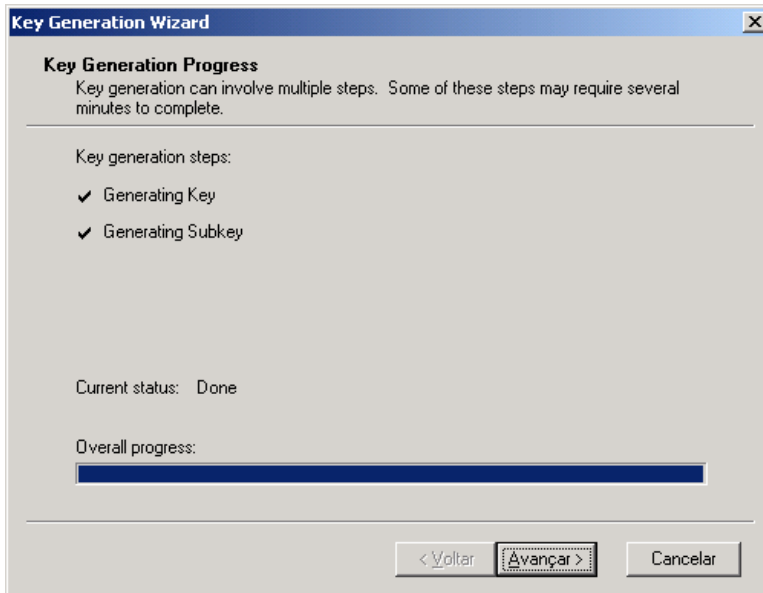


Figura 12 Key Generation Wizard (Key Generation Progress)

Durante a geração das chaves será mostrada a figura acima, após o termino o usuário terá que clicar em avançar.

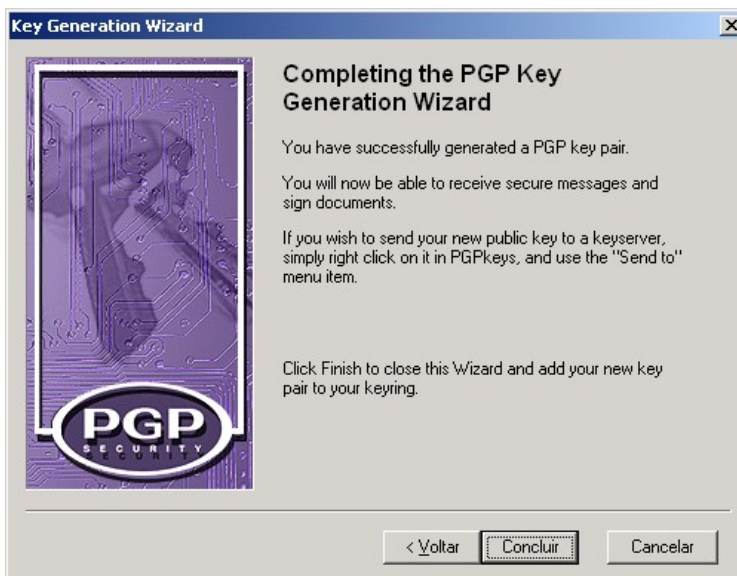


Figura 13 Key Generation Wizard (Completing)

E então o PGP mostrará uma mensagem de conclusão e o usuário deverá clicar em Concluir.

5.2 CONFIGURANDO O ALGORITMO SIMÉTRICO

O usuário pode no PGP definir qual o algoritmo simétrico deverá ser utilizada quando for cifrar uma mensagem. Para escolher o algoritmo deverá estar com PGKey aberto, clicar em Edit - Options, na palheta Advanced e na opção Preferred algorithm, deve se escolher o algoritmo. A figura a baixo mostra

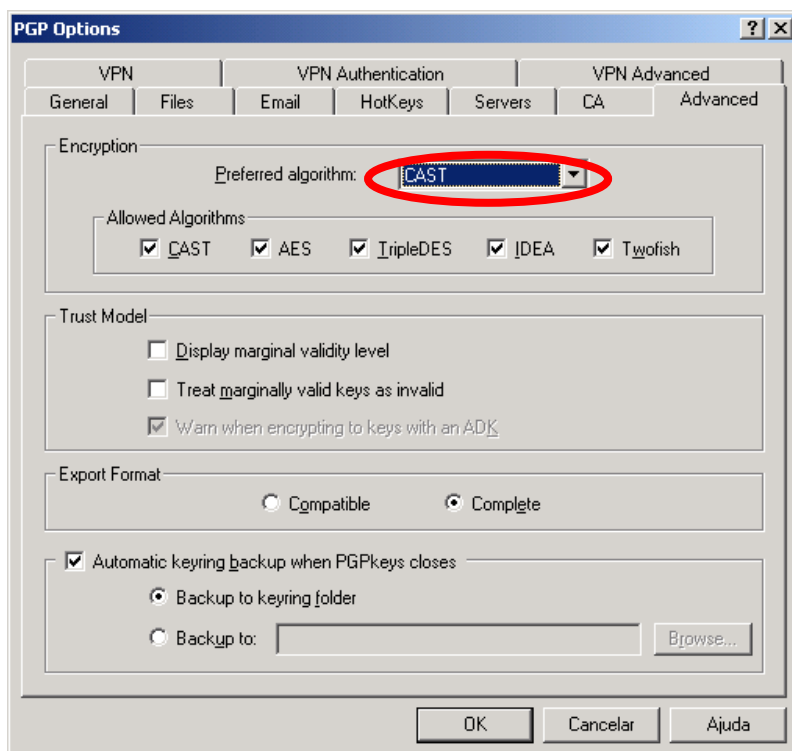


Figura 14 Configuração do PGP (Escolhendo um Algoritmo Simétrico)

5.3 DISPONIBILIZANDO CHAVE PÚBLICA ATRAVÉS DE UM SERVIDOR

Para que o usuário possa receber uma mensagem cifrada, sua chave pública tem que ser distribuída pelo maior número de pessoas possíveis, muitas pessoas utilizam de servidores públicos de chaves para distribuir suas chaves públicas. No PGP para o usuário enviar sua chave pública a um servidor deve fazer os seguinte passos:

1.º Passo: Caso não tenha nenhum servidor público de chaves cadastrado no PGPKeys, o usuário deve entrar no menu Edit – Options entrar na palheta Servers, como na figura abaixo.

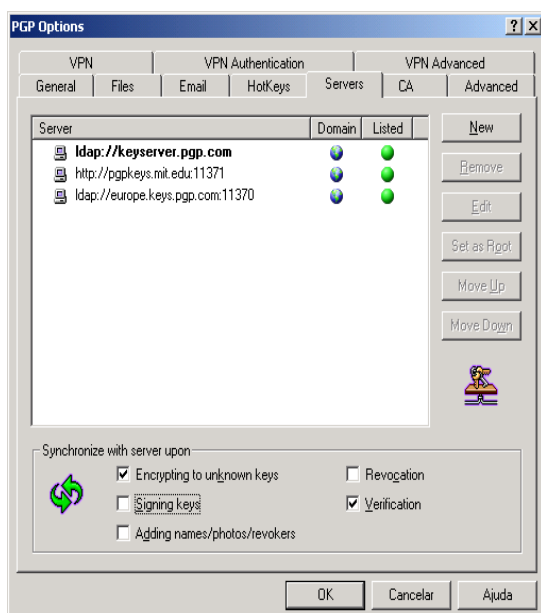


Figura 15 Configuração do PGP (Servidor Público de Chaves)

Clicar em New

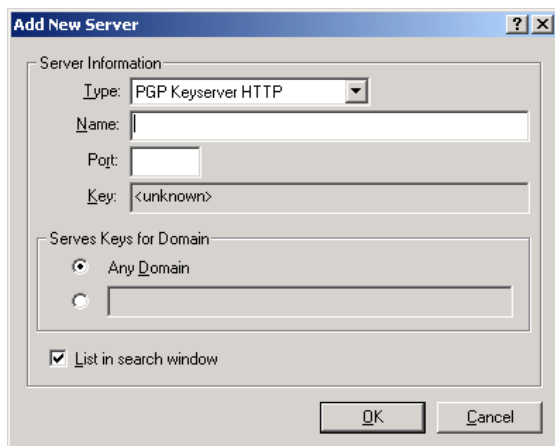


Figura 16 Adicionando um novo servidor público de chaves

O usuário terá que informar o tipo de servidor no campo Type, o endereço do servidor de chaves no campo Name e a porta que este serviço trabalha no servidor e então o usuário deverá clicar em Ok, como mostra na figura acima.

2.º Passo: O usuário deve selecionar a chave, no PGPKeys que deseja enviar ao servidor e então clicar no menu Server - Send To e escolher qual o servidor que ele deseja enviá-lo. Quando este processo for feito o computador do usuário deverá obrigatoriamente estar conectado a Internet.

5.4 EXPORTAR A CHAVE PÚBLICA EM TEXTO ASCII

As vezes o usuário quer disponibilizar sua chave pública através de disquetes ou um outro meio magnético, o PGP fornece esta opção de exportação de chave em arquivo texto. Para fazer isto o usuário terá que estar com o PGPKeys aberto, então ele deverá selecionar a chave a ser exportada clicar no meu Keys – Export e então selecionar o local onde será gravada a chave pública. Uma chave pública em formato texto pode ser observado na figura abaixo.

```

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GPGfreeware 7.0.3 for non-commercial use <http://www.pgp.com>

mQGfBD8wduARBADUAQT+r+hXRYTj6TJcJrGS6QM8QH2hi0kWEKAKZrWGZOGXqIEV
/JDDshsot8agb5Jwk0chj19FIqqkjKED15TyZz6iQUiGk5wWYJgbTv3bes0dE8LO
PawMBze+1xvAj4bqubwR+aY/Z1qC3R5EviXa4AgPMo7KujZiZn1LDMRhhQcG/1fM
7Rp029Budggg3wTAgDwkfX8D/2D6kGrFU31hzpIm25jgUXS1EvxIvvd+BFnJBvdL
hfjyrGZajv08oyycP0RvQB4KwenXdkVFLF/11v+psP4NxmZTzj5GoxR3EYbu96w/
GJxk1j2MtubGu7/OagVfPNF+uIPrfjpcX1/KF1FHb5Poxp91YAwpPG1I1pg5H4UP
kekAA/90j8CGDbmrT6BOAAR eHZmyowaK1NpM/c9KFOZaaFVZzqd3TFO+NwF1PM1U
TCBE7Enw1YxultH2AEY+0zgt88e4G+F5U3MJSIz61c0Qo82MupkmwN7P1AZReq/L
rw+tdwplvtKxTdkkvfj4QT/yufdrGx8pFTUhlazRJsLCE50zvLQqum9kcm1nbyBH
dwvkZXMgZUGU291emEGPGd1bg1yYUB5Ywhvby5jb20+iQBYBBARAgAYBQI/MHVA
CASDCQgHAGEkAhkBBRS0AAAAA0JEPx01E1DaAc8Aa0A0MwNkrkgwUvjgXbn1Pm
D3U1/zq7AKDKKroKAHdpBmA8PgrBdaSwuogMurkCDQq/MHVBEAgA9kJxtwh/CBdy
orrwqULzBej5UxE5T7bXbr1LOcDaAadwoxTjp0BV89AHxstDqZst90xkhn4DIO9
ZekX1KHTUPj1wv/cdlJPPt2N286Z4veSwc39Uk50T8XdrydXucwYc58ywb/Ffm7
/ZFexwGq01uejac1cjrUGvC/RgBYk+X0iP1Ytknbz5C0neSRBzZrM2w4DUudD3yI
sxx8wy209vPJI8BD8KvBGI2ou1wMuF040zT9fBdxQ6MdGGzemyEstSr/POGxKUAY
EY18hKckctagxAMZyAcPESqVDNmwn6vQC1CbAkbtCD1mpF1Bn5x8vY1LIhkmuguI
XSNV6TILowACAgf9Eaw+Uj0F6AWRVZK5Cpr71yxM7HHVf/XMDyJgXnb2vm29tA1N
+PRpnEh5qUsMIq4r7AkJN1y/yg37RI4goteyZ9ZPvrGUNov3no7MFx110Mv6FI1k
TOuCLB7Yp5Gpp8v7U15CwbweQoxDy8ygbwIwjIuhwKXotCBjxbawb3xvctcoSI
18EcpwxMILvd98zrjYbH359eche3DeMmdkzCCKb78YT6bk9gnALQx+ZDnNBLA4X3
K/F8GmQ+s1tpH5oveAc+Dmi0PxjYos008dBGTC7y11oc98qUsyj0Pv2m7ua7vHTI
1Fvs1a5ijq1U0wbb15SctTn7wJ4o//X/rENV+4kATAQYEQIADAUCPzB1QQubDAAA
AAAKCRD8TtRNQ2GHPow7AKDbAIowN5bFMUUF2Xlrjm1w8kwcFwcfQQL5ptjqZHTU
Rwr781g2J+6wcoU=
=THuz
-----END PGP PUBLIC KEY BLOCK-----

```

Figura 17 Formato de uma chave pública no PGPKeys

5.5 ADICIONANDO CHAVES PÚBLICAS

Para adicionar uma chave pública ao chaveiro do usuário, deve importa a chave através no menu Keys – Import, como na figura abaixo.

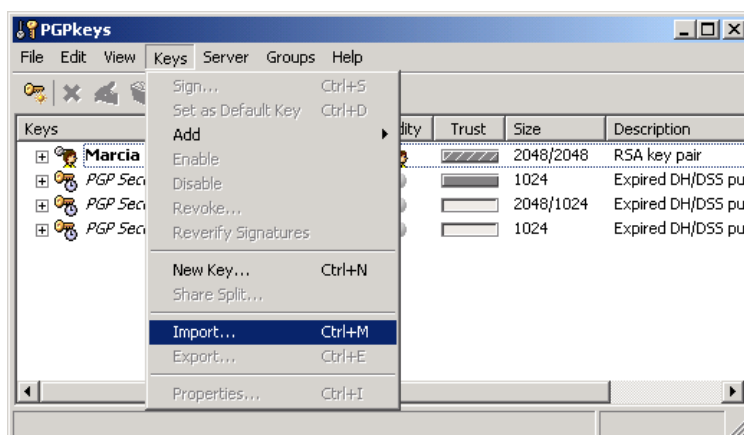


Figura 18 Menu de Importação de chave pública no PGPKeys

Então abrirá uma janela onde o usuário deve localizar a chave pública de um outro usuário, este arquivo pode ter a extensão .txt ou .asc. Quando o usuário abrir o

arquivo será mostrado a tela a baixo, onde conterà as chaves públicas á serem importadas.



Figura 19 Seleção de Chave Pública

Para importar, o usuário deve selecioná-las e depois clicar no botão Import. Como o PGP trabalha como uma teia de confiança o usuário após importar a chave deve fazer a assinatura sobre chave importada. Para assinar uma chave o usuário terá que selecionar a chave pública e depois clicar com o botão direito sobre ela, e então clicar sobre a opção Sign.

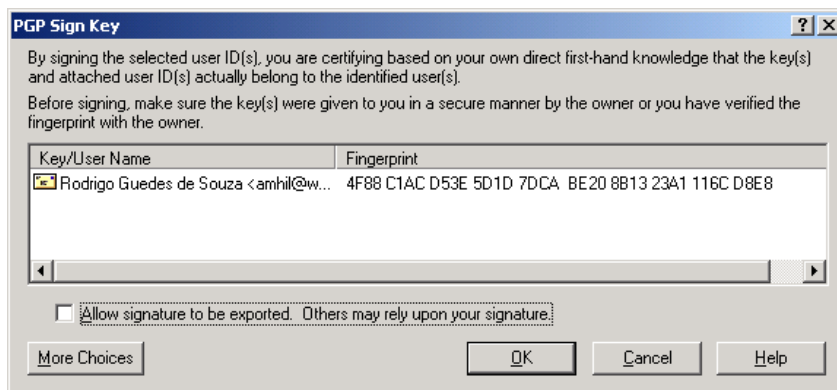


Figura 20 Assinatura de Chave

Será apresentado o nome do usuário e o fingerprint da chave pública, antes de assinar a chave o usuário deve ter absoluta certeza que a chave é realmente de quem diz ser, para acabar com este problema o usuário pode estar entrando em contado com o dono da chave por telefone, por exemplo, e podendo confirmar o fingerprint da chave. Então, confirmado o usuário poderá clicar no botão Ok. Deste modo a chave importada será assinada pelo usuário confirmando realmente sua validade.

5.6 UTILIZANDO PGP NO OUTLOOK 2000

Nesta seção será mostrado, como utilizar o PGP integrado a um programa de correio eletrônico, adotando-se o Microsoft Outlook 2000 (versão 9.0.0.2814) para a explicação. O PGP utiliza plug-ins para fazer a integração com o software, no Microsoft Outlook percebe-se que após a instalação, criou-se um menu chamado PGP, onde contem os principais recursos do PGP.

5.6.1 Enviando uma Mensagem

Para criptografar uma mensagem e enviar, deve seguir os seguintes passos:

- 1.º Passo: Digitar o texto da mensagem.
- 2.º Passo: Depois de criado, o texto deve ser criptografado com a chave pública do destinatário e assinado com a chave privada do remetente, clicando no menu PGP – Encrypt and Sign Now.

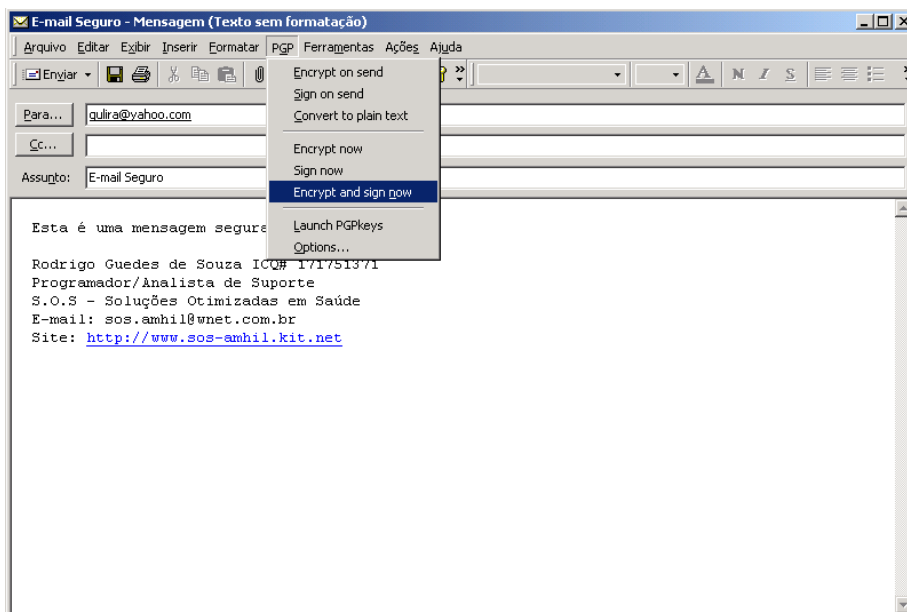


Figura 21 Criptografando e Verificando uma Assinatura de uma Mensagem recebida no Outlook 2000

3.º Passo: Depois que o usuário clicar em encrypt and sign now, será pedido a frase-senha que deverá ser informado. Se a frase-senha estiver correta o PGP criptografa a mensagem. O PGP permite outras opções, como por exemplo: só criptografar a mensagem e não assinar.

4.º Passo: Depois de criptografar o texto, o usuário terá que fazer a rotina padrão do outlook para enviar a mensagem, clicando no botão enviar.

5.6.2 Lendo uma Mensagem Criptografada

Para decifrar uma mensagem , deve seguir os seguintes passos:

- 1.º Passo: Deve abrir o outlook, clicar no botão enviar/receber mensagem.
- 2.º Passo: O usuário deverá clicar duas vezes na mensagem de e-mail para abrir em uma nova janela.
- 3.º Passo: Clicar no menu PGP – Decrypt / Verify, para decifrar a mensagem e verificar se foi realmente enviado por quem diz ser.

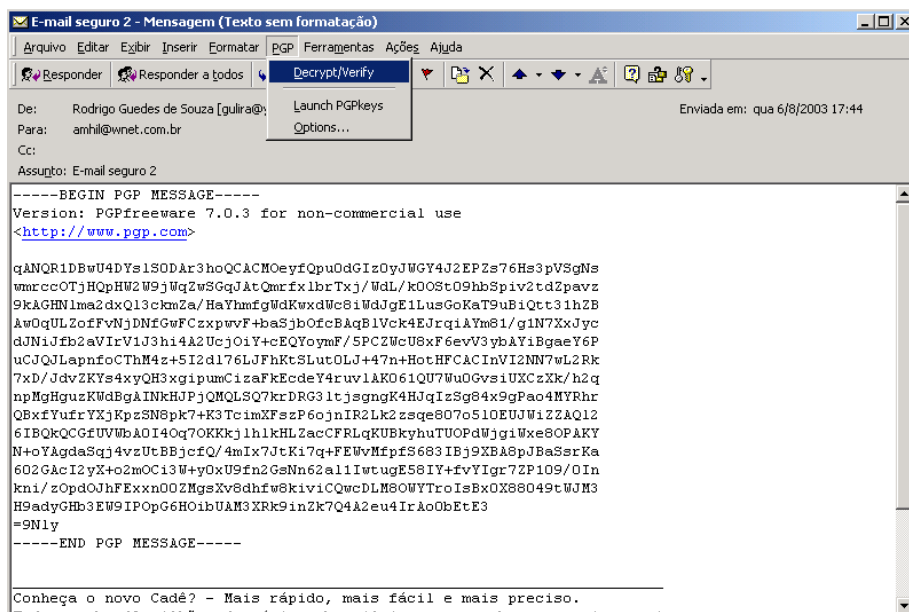


Figura 22 Decryptografando e Verificando uma assinatura de uma Mensagem Recebida no Outlook 2000

4.º Passo: Depois de feito o 3.º passo, o PGP pedirá ao usuário digitar a frase-senha para decryptografar a mensagem, e então a mensagem em texto plano (original) será apresentada na mesma janela.

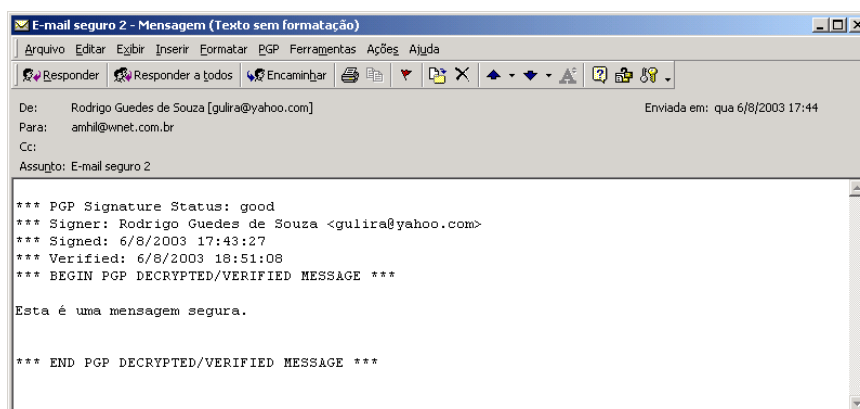


Figura 23 Mensagem em Formato Original

6 CONCLUSÃO

Com este estudo percebeu-se que o correio eletrônico não foi projetado para manter a segurança nas mensagens a serem enviadas ou recebidas. Na época em que foi desenvolvida esta aplicação, a necessidade era voltada ao desempenho do envio de mensagens e a definição de um padrão de formato, e não era visto como importante o fator de segurança. Tal falta levou a sérias conseqüências passar do tempo, quando a utilização do correio eletrônico virou febre mundial.

Muitas pessoas, principalmente da área militar, foram responsáveis por habilitar e evoluir a criptografia, para ser uma metodologia utilizada como subsídios às propriedades de segurança da informação.

Como mostrado no trabalho, o PGP é uma ferramenta que foi desenvolvida para fornecer privacidade às mensagens no correio eletrônico, até contra os governos, utilizando-se de criptografia pesada. Tais idéias que foram julgadas por atos que contrariavam leis.

O PGP tem uma ótima funcionalidade, pode ser implantado por qualquer pessoa e em quase todas as plataformas. Mas o único problema que ocorre é o fato esta tecnologia não ser divulgada.

Através deste trabalho, espera-se poder ensinar os usuários do correio eletrônico e estarem utilizando o PGP, para implantação de segurança no ambiente de trabalho.

Este trabalho poderá servir como guia para profissionais da área, como também utilizados em trabalhos futuros com objetivos educacionais.

REFERÊNCIAS

BURNETT, Steve; PAINE, Stephen. **Criptografia e Segurança. O Guia Oficial RSA.** Rio de Janeiro, Editora Campus, 2002. 367p. ISBN 85-352-1009-1.

CAVALHEIRO, Toni. Criptografia mais segura do que nunca. **Revista PC Master**, São Paulo, n.70, Março 2003.

GONÇALVES, Leandro S.; RIBEIRO, Vinicius G. **Um Estudo Comparativo entre Algoritmos de criptografia DES – Lúcifer (1997) e AES – Rijndael (2000).** 2001, Universidade Luterana do Brasil, Canoas, RS. Disponível por www em http://www.pesquisa.lasalle.tche.br/vinicius_comparativo_entre_algoritmos.pdf. Acesso em: 12 de Junho de 2003.

NETWORK ASSOCIATES INC. **PGP User's Guide – Version 6.0.** Santa Clara, 1998. 177p.

NETWORK ASSOCIATES INC. **PGP User's Guide – Version 7.0.** Santa Clara, 2001. 246p.

PIMENTA, Regina (ed.). Criptografia em 1.º lugar. **Revista do Linux**, Curitiba, n. 16, p. 16 – 19, Abr. 2001.

SADLER, Will. **Usando E-mail na Internet.** 1.Ed. Rio de Janeiro, Editora Campus, 1996. 728p. ISBN 85-352-0069-x.

RIBEIRO, Vinicius G. **Um Estudo Comparativo entre os Algoritmos Finalistas do AES.** 2001, Universidade Luterana do Brasil, Canoas, RS. Disponível por www em Http://www.pesquisa.lasalle.tche.br/vinicius_comparativo.pdf. Acesso em: 12 de Junho de 2003.

TANENBAUM, Andrew S. **Redes de Computadores.** 4. Ed. Rio de Janeiro, Editora Campus, 1997. 923p. ISBN 85-352-01572.

