UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC

CURSO DE CIÊNCIA DA COMPUTAÇÃO

JULIANA DA LUZ DE CAMPOS

AVALIAÇÃO DE DESEMPENHO DOS ALGORITMOS DE CRIPTOGRAFIA: *DATA ENCRYPTION STANDARD* E ALGORITMO ASSIMÉTRICO (RSA)

CRICIÚMA, JULHO DE 2008

JULIANA DA LUZ DE CAMPOS

AVALIAÇÃO DE DESEMPENHO DOS ALGORITMOS DE CRIPTOGRAFIA:

DATA ENCRYPTION STANDARD E ALGORITMO ASSIMÉTRICO (RSA)

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

Orientador: Prof. MSc. Paulo João Martins.

CRICIÚMA, JULHO DE 2008

JULIANA DA LUZ DE CAMPOS

AVALIAÇÃO DE DESEMPENHO DOS ALGORITMOS DE CRIPTOGRAFIA: *DATA ENCRYPTION STANDARD* E ALGORITMO ASSIMÉTRICO (RSA)

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

_____

Prof[a] MSc. Ana Cláudia Garcia Barbosa

Coordenadora do Curso de Ciência da Computação

Banca Examinadora:

_____

Prof. MSc. Paulo João Martins (UNESC)

Orientador

_____

Prof. MSc. Rogério Antônio Casagrande (UNESC)

_____

Prof. Esp. Arildo Sônego (UNESC)

Dedico essa pesquisa aos meus pais

Jair de Campos e Elizabete da Luz de Campos

pelo estímulo e boa educação que me deram ao longo da minha vida.

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me dado a oportunidade de viver e por nunca ter me abandonado em nenhum momento, mesmo naqueles em que duvidava da sua existência.

Ao meu pai Jair e minha mãe Elizabete pela dedicação e carinho, pelos conselhos e de certa forma pelas broncas, muitas vezes merecidas, por terem investido financeiramente em mim e principalmente pelo exemplo de vida, o qual pretendo copiar e seguir.

Aos meus irmãos Saimon e Higor que não imaginavam o esforço de uma faculdade, mas sempre souberam entender meus rompantes nos momentos de nervosismo e angústia.

Aos meus colegas de classe pelo incentivo, alegria e momentos de descontração. Tais momentos foram fundamentais para reflexão sobre os problemas encontrados no decorrer desta pesquisa.

Aos meus amigos e amigas que muitas vezes receberam um não como resposta a muitos convites, e souberam entender os momentos de irritação, mesmo quando suas intenções eram apenas me distrair.

Ao meu paitrão, por todas as folgas, necessárias para construção desta pesquisa.

A todos que direta ou indiretamente contribuíram para a realização desta conquista, dedico aqui todo o meu agradecimento, carinho e muito obrigada !

“A mente que se abre a uma nova idéia

jamais voltará ao seu tamanho original.”

Albert Einstein

RESUMO

Devido ao grande acesso às informações computacionais, por meio de sistemas de informação e da Internet, existem problemas relacionados à segurança e por isso existiu a necessidade de pensar-se sobre a criptografia que é um dos meios de garantir segurança mais utilizados atualmente. Este trabalho de conclusão de curso apresenta conceitos relacionados à criptografia, e além disso, se propôs a realizar uma análise sobre dois dos mais importantes algoritmos existentes até o momento: o algoritmo simétrico DES e o algoritmo assimétrico RSA. O sistema operacional escolhido para a realização dos testes foi o Windows, por ser o sistema mais utilizado pelos usuários e os softwares de criptografia utilizados foram o JR Cripto e o JR Cripto DES. Esta monografia tem como objetivo principal realizar uma comparação entre os algoritmos DES e RSA, analisando-os em fatores como funções criptográficas, velocidade de processamento, requisitos e tamanho de chaves descrevendo suas vantagens e desvantagens. Além disso, pretende expor pesquisas já realizadas sobre os mesmos.

Palavras Chave: Criptografia, Algoritmo RSA, Algoritmo DES, Segurança de Dados.

ABSTRACT

Due to the large access to computational information, through information systems and the Internet, there are problems related to security and therefore there was the need to think about cryptography, which is one of the most used security systems tools. This course conclusion paper presents concepts related to cryptography, and besides that, it proposed to undertake an analisys about the two most important algorithm nowadays: the symmetrical algorithm DES and the asymmetrical algorithm RSA. The chosen operating system to conduct the tests was the Windows, as it is the most common system among computer users, and the encryption softwares used were the JR Crypto and JR Crypto DES. This monograph main goal is to hold a comparison between the DES and RSA algorithms, analyzing them on factors such as cryptographic functions, processing speed, requirements and key sizes, describing their advantages and disadvantages. It also aims to expose researches already conducted on the same subject.

Keywords: Encryption, RSA algorithm, DES algorithm, Data Security.

LISTA DE SIGLAS

AES	*Advanced Encryption Standard*
ALPOS	Algoritmo Posicional
CSI	*Computer Security Institute*
DES	*Data Encryption Standard*
DSA	*Digital Signature Algorithm*
ECC	*Elliptic Curve Cryptography*
IDEA	*International Data Encryption Algorithm*
MIT	*Massachusetts Institute of Technology*
NIST	*Institute of Standards and Technologies*
NSA	*National Secunty Agency*
RC5	*Rivest Cipher 5*
RSA	*Rivest Shamir Adleman*
SIP	Sistemas de Informações Processuais
XOR	Ou Exclusivo

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO

Nos últimos anos com a ampliação das comunicações via rede de computadores e pela própria *Internet*, a segurança e legitimidade dos dados transmitidos tem se mostrado algo importantíssimo. A cada dia que passa, amplia-se o número de pessoas dedicadas ao estudo e ao desenvolvimento de técnicas que têm por finalidade proteger a informação. A Criptografia é uma das técnicas utilizada para garantir a integridade dos dados, e é tão antiga quanto a própria escrita. Com o aumento de invasores aos bancos de dados e as informações em geral, recentemente a criptografia tornou-se alvo de diversos estudos científicos.

Existem no mercado, vários algoritmos capazes de realizar a criptografia de informações a fim de torná-las seguras. Contudo, desde que esses sistemas de codificações foram criados, o algoritmo *Data Encryption Standard* (DES) tornou-se popular entre seus usuários. Ele é considerado o mais eficiente entre os algoritmos utilizados para o mesmo fim. Entretanto, existe o Algoritmo Assimétrico (RSA), criado na mesma época que o DES e considerado tão eficiente quanto ele, mas não obtendo a mesma aceitação.

Pode-se compreender o fato de que o primeiro algoritmo foi o pioneiro e deste modo, aceito como o definitivo entre os algoritmos. Entretanto, o mercado necessita de outros sistemas alternativos, para não ficar a mercê de um único que possa ser descoberto por pessoas especializadas, comprometendo sua eficácia. Neste contexto, a problemática em questão baseia-se na necessidade de analisar e documentar algoritmos com o intuito de mostrar as vantagens e desvantagens dos mesmos, bem como mostrar o potencial de uso dos algoritmos mais confiáveis do mercado, no caso o DES e o RSA.

1.1 OBJETIVO GERAL

Avaliar o desempenho dos algoritmos criptográficos: Algoritmo Assimétrico (RSA) e *Data Encryption Standard* (DES), visando fazer um estudo comparativo entre os mesmos.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos desta pesquisa são:

a) compreender o processo de tratamento de criptografia;
b) analisar os recursos das ferramentas utilizadas para criação de uma criptografia;
c) avaliar o desempenho do uso do Algoritmo RSA e do Algoritmo DES, realizando uma comparação entre os mesmos, mostrando suas vantagens e desvantagens;

1.3 JUSTIFICATIVA

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 por meio 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.

A privacidade é uma questão fundamental para pessoas e empresas. Inúmeros problemas podem ocorrer se pessoas não autorizadas tiverem acesso a dados

pessoais, como: contracheque, saldo bancário, faturas do cartão de crédito, diagnósticos de saúde e senhas bancárias ou de crédito automático, entre outras. No caso de empresas, os danos podem ser mais sérios, atingindo a organização e os próprios funcionários. Dados estratégicos da empresa, previsão de venda, detalhes técnicos de produtos, resultados de pesquisas e arquivos pessoais são informações valiosas, às quais se alguma empresa concorrente tiver acesso de forma indevida, tal fato poderá acarretar sérios problemas (MORENO; PEREIRA; CHIARAMONTE, 2005).

O guia de Segurança Máxima (2000) destaca que dentre os aspectos de segurança citados acima, uma das principais razões que permitem o ataque externo é a má configuração do sistema. As instituições precisam de um sistema que proteja, controle e monitore seus recursos computacionais e, conseqüentemente, as informações por eles manipuladas.

Em ambientes computacionais, a segurança é um atributo muito complexo e difícil de ser implementado. Projetar e implementar um sistema visando segurança significa analisar um conjunto complexo de situações adversas, o resultado desta análise depende fortemente das escolhas e técnicas utilizadas pelo projetista. Assim, um sistema é considerado seguro se não foi possível, determinar uma maneira de torná-lo inseguro.

Da mesma forma, a medida que a tecnologia de armazenamento e manipulação da informação se torna mais complexa, as oportunidades para que ela seja utilizada por indivíduos não autorizados são cada vez maiores. É neste contexto que a criptografia realizada por meio de algoritmos, tem um papel muito importante. Ela não é capaz de assegurar que um sistema seja considerado 100% seguro, mas dentre todos os sistemas, seria o mais próximo do ideal.

Segundo Righetti (2004, p. 3) “Criptografia é a ciência que consiste na arte da transformação de mensagens numa representação sem significado para qualquer

pessoa exceto para quem saiba qual o processo de reverter à transformação". No uso generalizado em redes de comunicação de dados a criptografia estende-se a diversos domínios, desde a autenticação de utilizadores, a privacidade de comunicações pessoais, ou difundidas, em canais de comunicação poucos seguros, ou de acesso livre.

A criptografia contemporânea não é mais baseada em obscuridade, ou seja, não se utiliza mais a suposição de que qualquer sistema pode ser seguro na medida em que ninguém, exceto seus criadores, tem acesso à metodologia ou aos algoritmos utilizados internamente ao sistema. Para uso moderno, um sistema criptografado tem sua segurança baseada nos algoritmos de cifragem e decifragem, os quais possuem um valor secreto – a chave. O mecanismo deve ser tão seguro que nem mesmo o autor de um algoritmo deve ser capaz de decifrar um texto cifrado sem dispor da chave apropriada (WEBER, 2007).

Os algoritmos de criptografia, portanto, precisam a cada dia ser mais estudados, conhecidos e testados, pois é o um dos tipos de sistema de segurança que poderá assegurar a troca de informação cibernética de maneira a não permitir o acesso a pessoas não autorizadas. Esses algoritmos são mais seguros do que muitos sistemas já existentes, para essa função. Os mais conhecidos e estudados pelas universidades são o DES e o RSA, exatamente aqueles considerados de maior segurança e escolhidos para fazer parte do estudo em questão (RIGHETTI, 2004).

Segundo Moreno, Pereira e Chiaramonte (2005, p. 113) "DES é um dos algoritmos de cifragem mais usados no mundo". É composto de operações simples, como: permutações, substituições, Ou Exclusivo (XOR) e deslocamentos. Já o algoritmo RSA consiste em gerar uma chave pública (geralmente utilizada para cifrar os dados) e uma chave privada (utilizada para decifrar os dados) por meio de números primos grandes, o que dificulta a obtenção de uma chave a partir da outra.

Devido à existência de mais de um padrão de criptografia, seria necessário a comparação desses dois algoritmos com relação a velocidade de cifragem, decifragem e a segurança após cifrados. Sabe-se que é de fundamental importância assegurar o fluxo das informações entre os usuários, sem que ocorra “vazamento” de informações para terceiros.

Essa pesquisa não tem a pretensão de esgotar todas as possibilidades de investigação relacionada aos algoritmos, mas trará o início de uma explanação com valor científico, já que o número de usuários de sistemas que necessitam de segurança é tão grande quanto o número de pessoas que acessam a *Internet* todos os dias. Além disso, o mercado de sistemas de segurança ainda é muito precário e necessita de novas alternativas, para não limitar-se a um ou outro componente, correndo o risco de tornar-se de uma hora para a outra, vulnerável a usuários experientes que consigam burlar um único algoritmo considerado o mais bem sucedido e a prova de falhas.

1.4 ESTRUTURA DO TRABALHO

Este trabalho está estruturado em sete capítulos distribuídos da seguinte forma:

No capítulo 1 está a introdução que descreve o contexto deste trabalho, com a introdução do tema da pesquisa, na qual estão apresentados os seguintes itens: introdução, o objetivo geral e objetivos específicos, justificativa e a estrutura do trabalho.

Já no capítulo 2 são apresentados os conceitos de segurança de dados e segurança eletrônica.

Dando seqüência ao trabalho encontra-se o capítulo 3, onde o mesmo descreve os conceitos básicos sobre criptografia, história e definição, o que são cifras e

chaves, os principais tipos de criptografia e os principais algoritmos criptográficos, principalmente os estudados nessa pesquisa.

O capítulo 4 apresenta os conceitos, técnicas do algoritmo DES.

Assim como o DES são apresentados conceitos, técnicas do algoritmo assimétrico RSA que estão descritos nos capítulo 5.

As pesquisas que utilizam a técnica de Criptografia utilizadas como referência nesse trabalho onde são relatados cinco dos diversos trabalhos de conclusão de cursos, que encontram-se no capítulo 6 desta pesquisa.

O capítulo 7 descreve todo o estudo desenvolvido, pesquisas, testes realizados, objetivos alcançados e ao alcançados, e uma comparação entre o algoritmo DES e o algoritmo RSA.

Na conclusão apresentam-se a visão do acadêmico sobre o trabalho, e sobre as técnicas estudadas, como também as contribuições e recomendações para trabalhos futuros. E, posteriormente, são listadas as referências, a bibliografia consultada e os Anexos.

2 SEGURANÇA DE DADOS

Com a chegada dos computadores pessoais e das redes de computadores, as empresas passaram a ter uma maior preocupação com a segurança de suas informações. A partir do momento em que os sistemas de informação das organizações passaram a ser acessados de uma rede externa, aumentou-se o risco de ocorrerem acessos desautorizados às suas bases de dados, comprometendo assim, a sua segurança. As informações representam um elemento importante para uma organização, pois são fundamentais para o seu sucesso e sobrevivência. Elas são consideradas recursos que estão sob constante risco, pois podem despertar o interesse de ladrões e, até mesmo de empresas concorrentes interessadas em informações estratégicas para proveito próprio. Segurança de dados é um assunto muito complexo, com muitos meandros, algo sem semelhantes pelo simples fato de lidar com um número extremamente grande de variáveis e ainda poder incorrer no aparecimento de variáveis novas e totalmente imprevisíveis (FARIA, 2006).

Segundo Moreno, Pereira e Chiaramonte (2005, p.24) “A *Internet* é um ambiente que viabiliza principalmente a comunicação, a divulgação, a pesquisa e o comércio eletrônico”.

No ano de 1999, existiam mais de 100 milhões de usuários na *Internet* nos Estados Unidos. Em 2003 esse número aumentou para 177 milhões, chegando a 502 milhões em todo o mundo. Também em 1999 o comércio eletrônico surgiu como um novo setor da economia, movimentando mais de U$100 bilhões em vendas, atingindo a marca de U$ 1 trilhão no ano de 2003. Da mesma forma que o índice de vendas cresceu, segundo os autores, o *Computer Secutity Institute* (CSI) constatou um aumento de atividades maliciosas no ambiente cibernético. Diante disso, as empresas em expansão,

cada vez mais se conscientizam a respeito da necessidade de produtos, mecanismos e soluções para a segurança de suas informações (BURNETT; PAINE, 2002).

A segurança eletrônica nunca foi tão discutida como nos tempos atuais, pois no auge do capitalismo e da globalização, as pessoas não desejam ter seu tempo desperdiçado. Infelizmente, os casos de violação de contas bancárias, o acesso à informações sigilosas, invasão e destruição de sistemas são cada vez mais comuns, inviabilizando a utilização e a confiança dos usuários, que ainda preferem fazer suas transações fora da rede de comunicações virtuais (MORENO; PEREIRA; CHIARAMONTE, 2005).

Um meio para se ter a solução desses problemas, é praticamente obrigatório a utilização de técnicas de escrita secreta que garantam a integridade, origem e autenticidade dos conteúdos que trafegam nas redes de computadores. O meio mais adequado de repassar informações entre um emissor e um destinatário sem que haja interferência é por meio de cifras (MORENO; PEREIRA; CHIARAMONTE, 2005).

Uma Cifra implica em um método de escrita ilimitado no seu uso. Este método deve ser capaz de transformar qualquer mensagem em uma forma não compreensível de codificação, chamada criptografia. As cifras criptografadas são obtidas por meio de algoritmos que oferecem um sistema de proteção por meio de permissões, isto é, um algoritmo seguro é capaz de garantir o acesso a informações diferenciadas, de acordo com o nível de permissão, exatamente como ocorre num sistema operacional, contudo, quando um invasor obtém o conteúdo de um arquivo criptografado, este será ilegível. Para ter acesso à informação original, o invasor terá que resolver um problema matemático de difícil solução, diferentemente do sistema operacional que possui técnicas de segurança amplamente conhecidas. Devido a isso a

criptografia é considerada um dos meios mais seguros para o armazenamento e transmissão de dados (MORENO; PEREIRA; CHIARAMONTE, 2005).

3 CRIPTOGRAFIA

A Criptografia é tão antiga quanto a própria escrita. Na história humana sempre houve fórmulas secretas e informações que não podiam cair nas mãos inimigas, principalmente nas guerras mundiais, e com a invenção do computador a criptografia foi crescendo e incorporando algoritmos matemáticos muito complexos (MORENO; PEREIRA; CHIARAMONTE, 2005).

3.1 HISTÓRIA E DEFINIÇÃO

Segundo Kahn (1967 apud MORENO; PEREIRA; CHIARAMONTE, 2005) o primeiro exemplo documentado da escrita cifrada relaciona-se aproximadamente ao ano de 1900 a.C, quando o escriba de Khnumhotep II teve a idéia de substituir algumas palavras ou trechos de texto. Caso o documento fosse roubado, o ladrão não encontraria o caminho que o levaria ao tesouro e morreria de fome perdido nas catacumbas da pirâmide.

A palavra criptografia vem do grego *Kryptos* que significa escondido, oculto e *Grafia* que significa escrita, e pode se entendida como um conjunto de métodos e técnicas para cifrar ou codificar informações legíveis por meio de um algoritmo, convertendo um texto original em um texto ilegível, sendo possível mediante o processo inverso recuperar as informações originais. O objetivo principal é garantir que o armazenamento e circulação das mensagens sejam seguros (DELIBERADOR, 2004).

Righetti (2004) afirma que ela existe há muito tempo e que há anos a criptografia deixou de ser uma arte para virar um conjunto de técnicas que tratam da

proteção de dados. Muitas pessoas já utilizaram essa técnica informalmente mesmo sem saber, pois de uma forma ou de outra, queriam manter em segredo o conteúdo de suas mensagens, sem que pessoas indevidas pudessem decifrá-la.

Os criptoanalistas utilizam a análise criptográfica que é o estudo sobre a quebra de sistemas de criptografia, para descobrir e divulgar as fraquezas e falhas dos algoritmos, diferente dos invasores que dificilmente divulgam as possíveis falhas encontradas nos mesmos. Os profissionais que desenvolvem os sistemas e algoritmos de criptografia são chamados de criptógrafos (BURNETT; PAINE, 2002).

Para Terada (2000 apud DELIBERADOR, 2004, p. 9) a criptografia pode ser descrita como:

> Ciência que estuda a transformação de dados de maneira a torná-los incompreensíveis sem o conhecimento apropriado para a sua tradução, tornando os conteúdos secretos, evitando riscos internos e externos que venham a ocorrer durante o trajeto dos dados enviados, que são convertidos em um código que só poderão ser traduzidos por quem possuir a “chave” secreta, enquanto que a criptoanálise executa o processo inverso, sendo a ciência que estuda a decifração, tornando o código compreensível.

A criptografia então pode ser definida como um instrumento que protege os dados enviados em rede local ou *internet*, fazendo com que os mesmos se tornem incompreensíveis as pessoas não autorizadas a ter acesso as informações contidas no arquivo.

Devido ao aumento diário do número de pessoas mal intencionadas na *internet* ou mesmo em sistemas de informações dentro das empresas, a necessidade de se utilizar a criptografia é cada vez maior, pois a mesma é utilizada para codificar dados e mensagens antes de serem enviados a seu destino pelas redes abertas de informações. Mesmo que esses dados sejam interceptados por pessoas não autorizadas, serão ilegíveis ao mesmo. Para que essas mensagens possam ser lidas, é preciso utilizar funções

matemáticas e senhas especiais para codificação, que são chamadas de chaves (DELIBERADOR, 2004).

Mesmo sendo algo difícil de realizar, existem pessoas especializadas em estudos e formas de se descobrir qual o algoritmo empregado na criptografia. Devido a isso, o uso de uma chave torna mais complicado para descriptografar a mensagem interceptada.

Segundo Deliberador (2004, p. 9) apesar de não ser algo simples de se fazer, existem pessoas especializadas em descobrir o algoritmo empregado na criptografia. Desta forma, a utilização de uma chave se torna mais difícil para descriptografar a mensagem supostamente interceptada. Para que se possa entender melhor como funcionada a utilização desta chave, imagine uma porta muito resistente em uma residência. Nesta porta é colocada uma fechadura que qualquer chave possa abrir. Para quem passa do lado de fora da casa, a porta está fechada e a casa aparentemente segura, porém, qualquer pessoa pode abri-la simplesmente, girando a fechadura.

A Figura 1 mostra esquematicamente, como se dá o processo criptográfico na transmissão de dados e/ou informações.

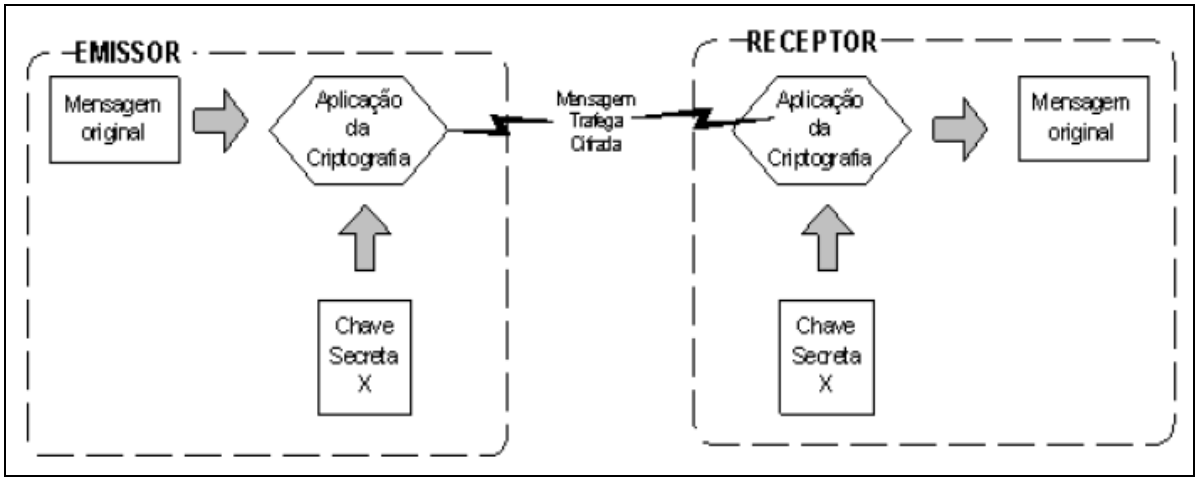

Figura 1. Ilustração de um processo simples de criptografia no envio da mensagem
Fonte: VOLPI, M. (2001, p.7)

Garfinkel e Spafford (1999 apud DELIBERADOR, 2004, p.10) afirmam que as quatro palavras-chave usadas para descrever todas as diferentes funções que a criptografia desempenha nos sistemas modernos de informações são:

a) confidencialidade: embaralha as informações enviadas por canais abertos e armazenadas em servidores, de forma que as pessoas não autorizadas não possam acessar o conteúdo dos dados e utilizá-los de forma incorreta para interferir sobre o todo. Sua finalidade é garantir que somente a origem e o destino tenham conhecimento do exato do teor das informações;

b) identificação: o receptor da mensagem consegue verificar a identidade da pessoa que assinou, e pode ser implementada a partir de senhas ou assinaturas digitais, garantindo assim a identidade de quem está enviando a mensagem;

c) integridade de dados: garante que qualquer receptor, que não seja o destinatário dos dados, fique impedido de modificá-los ou deles se apropriar, pois os dados recebem um tratamento onde são transformados em caracteres ilegíveis, impossibilitando assim que o conteúdo da mensagem seja modificado em trânsito. Sendo assim, a informação que é transmitida de um ponto chega igualmente ao outro ponto;

d) não – repúdio: por meio de procedimentos criptográficos e da assinatura digital, faz com que o autor da mensagem enviada não possa negar falsamente que a tenha enviado. Se uma entidade enviou uma mensagem a outra entidade, não podem negar a autoria da mesma ou o recebendo dela.

O meio mais adequado de repassar informações entre um emissor e um destinatário sem que haja interferência é a utilização de cifras, onde as letras da mensagem original são substituídas, fazendo com que somente as pessoas autorizadas possam ter acesso as informações originais conhecendo o processo de cifragem (MORENO; PEREIRA; CHIARAMONTE, 2005).

3.2 CIFRAS

Deliberador (2004) afirma que uma cifra é uma função matemática ou algoritmo criptográfico, que realiza a transformação entre o texto limpo e o criptograma. A palavra “algoritmo” é utilizada para nomear procedimentos passo a passo, ou seja, é uma lista de instruções que devem ser executadas em uma determinada ordem. Essas listas podem conter perguntas variadas, e dependendo das respostas, descrevem os passos apropriados a serem seguidos.

As mesmas são divididas em substituição e permutação, as quais são descritas a seguir.

3.2.1 Cifras por substituição

De acordo com Righetti (2004) são aquelas em que os caracteres da mensagem são sistematicamente substituídos por outros caracteres. As quais são divididas nas seguintes categorias:

a) cifras mono alfabéticas: são algoritmos que estabelecem um mapeamento único para todos eles, sem desordenar os símbolos dentro da mensagem. A seguir estão descritos esses algoritmos:

- algoritmo de César: de acordo com Moreno, Pereira e Chiaramonte (2005) em 50 a.C, Júlio César alterou letras desviando-as em três posições; A se tornava D, B se tornava E e assim por diante, fazendo com que isso tornasse seu texto original em um texto cifrado. Para que seu método fosse reforçado, César também substituía letras latinas por gregas. Seu código é o único da Antiguidade que ainda é utilizado atualmente, denominando-se código de César. Foi utilizada pelos oficiais sulistas na Guerra de Secessão Americana e pelo exército Russo em 1915, devido a sua simplicidade,

- algoritmo rot 13: substitui uma letra por outra, situada treze posições à frente no alfabeto,

- algoritmo shift-n: as letras são substituídas por outras situadas mais à frente no alfabeto, sendo que o número dos deslocamentos a serem realizados é definido no algoritmo;

b) cifras polialfabéticas: neste algoritmo a substituição é aplicada a cada letra do texto, o qual varia em função da posição que ele ocupada dentro do texto plano;

c) cifras homofônicas: ao empregar um alfabeto de saída com mais símbolos do que o alfabeto de entrada, estes tipos de algoritmos tratam de ocultar as propriedades estatísticas do texto plano, associando várias saídas a um símbolo, impossibilitando assim que um ataque baseado em freqüências ocorra mais vezes na palavra.

3.2.2 Cifras por permutação

Nas cifras por permutação, que também são conhecidas como de transposição, consistem em misturar as letras do texto original de acordo com uma regra reversível qualquer onde são mantidos os mesmos caracteres do texto, apenas rearranjando suas posições (RIGHETTI, 2004).

As cifras de transposição reordenam as letras, mas não as disfarçam, mantendo o conteúdo da mensagem inalterado (TANENBAUM, 1997).

Na Figura 2 pode-se ver um exemplo de cifra de transposição:

M	E	G	A	B	U	C	K
7	4	5	1	2	8	3	6
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

Figura 2. Cifra de transposição de colunas
Fonte: TANENBAUM, A. (1997)

3.3 CHAVES

Na criptografia existem dois tipos de textos. O primeiro chamado de texto puro é a mensagem a ser transmitida, na sua forma original. Esse texto passa por um processo chamado encriptação, que fará com que ele assuma uma nova forma, que se

denomina texto cifrado. Este texto deverá permanecer ininteligível quando for interceptado por um terceiro, pois será a forma transmitida. Para recuperar o texto original ou puro, ao receber o texto cifrado o destinatário usará um processo chamado desencriptação (CARVALHO, 2001).

Segundo Deliberador (2004) as chaves interagem com os algoritmos para cifrar e decifrar as mensagens, tornando-se assim elementos fundamentais. O algoritmo realiza seus passos utilizando a chave para alterar o texto normal e convertê-lo em texto cifrado, e para decifrá-lo tornando em texto legível é necessário inserir a mesma chave. A Figura 3 mostra como esse processo funciona:

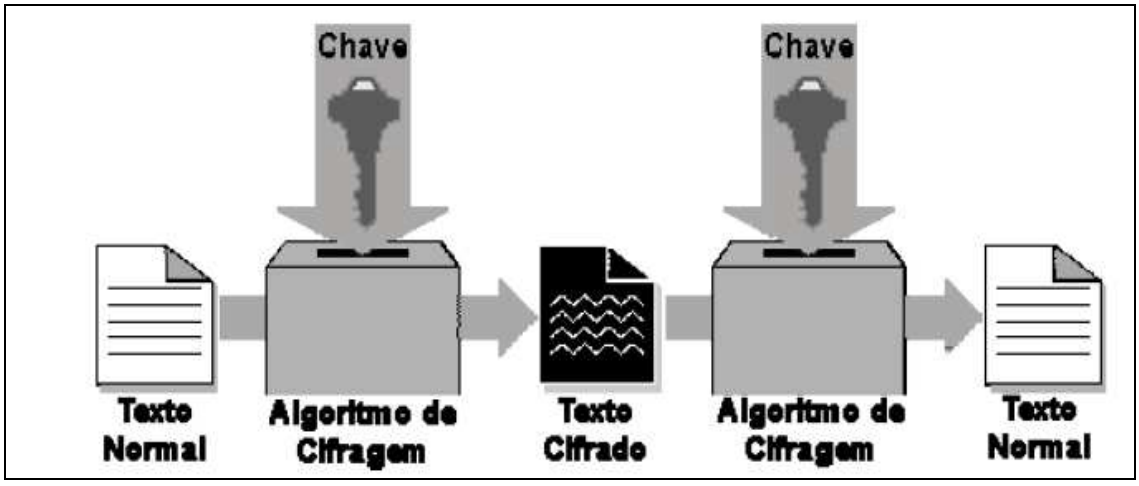

Figura 3. Ilustração de um processo para cifrar e decifrar utilizando chaves
Fonte: GARFINKEL, S.; SPAFFORD, G. (1999, p. 189)

Conforme afirma Burnett e Paine (2002 apud DELIBERADOR, 2004), se for implementada incorretamente, a criptografia não garante a segurança de dados, nem resolverá todos os problemas de segurança, não é a prova de falhas, podendo ser quebrada. Porém, pode-se dificultar muito a quebra de segurança, com a utilização das chaves nos algoritmos de criptografia.

Assim como as senhas de acesso aos bancos e os sistemas de computadores, nas chaves de criptografia o uso da senha correta torna possível ter acesso aos serviços e decifrar as mensagens recebidas, caso a mesma esteja incorreta o acesso será negado, pois o acesso ou não a informação cifrada relaciona-se com o uso das chaves (DELIBERADOR, 2004).

Lynch e Lundquist (1996 apud DELIBERADOR, 2004) afirmam que as chaves na criptografia possuem tamanhos diferentes, assim como as senhas, pois seu grau de segurança está relacionado à sua extensão. Quanto maior for a senha ou chave de um usuário, maior será o grau de confidencialidade da mensagem ou do serviço.

Na criptografia, as chaves são longas seqüências de *bits*. Uma chave de três dígitos oferecerá oito ($2^3 = 8$) possíveis valores para a mesma. Como a chave é uma *string* que pode ser alterada, o algoritmo pode ser conhecido, fazendo com que vários especialistas tentem decodificar o sistema. Se depois de alguns anos, esses especialistas e estudiosos não conseguirem quebrar o algoritmo de criptografia, significa que o mesmo é seguro (PICONI, 2004).

Entre os anos de 1933 e 1945, o matemático Marian Rejewski baseou-se em textos cifrados e uma lista de chaves obtidas por um espião, para quebrar o sistema e aperfeiçoar a máquina Enigma que havia sido criada por Arthur Scherbius, até a mesma se transformar na ferramenta de criptografia mais importante da Alemanha nazista (MORENO; PEREIRA; CHIARAMONTE, 2005).

Em relação ao uso das chaves, existem dois tipos de criptografia, a simétrica e a assimétrica. Na criptografia simétrica é utilizada a mesma chave, tanto para cifrar a mensagem quanto para decifrar, já na assimétrica são utilizadas chaves diferentes para esses procedimentos (FUZITAKI, 2004).

3.4 CRIPTOGRAFIA SIMÉTRICA

Moreno, Pereira e Chiaramonte (2005) afirmam que na criptografia simétrica o remetente e o destinatário usam a mesma chave para cifrar e decifrar mensagens, ou seja, para esses dois processos é utilizada apenas uma única chave.

Nos algoritmos simétricos, ocorre o chamado de problema de distribuição de chaves, pois a chave tem que ser enviada para todos os usuários autorizados, antes que as mensagens sejam trocadas, resultando assim um atraso de tempo e possibilitando que a chave chegue a pessoas não autorizadas e até mesmo mal intencionadas.

Por isso esse processo de criptografia é pequeno e rápido, mas com a desvantagem de que não só o transmissor conhece a chave mas também o receptor (PICONI, 2004).

A chave privada funciona muito bem quando o usuário que encripta a mensagem é o mesmo que irá desencriptar (MORAES, 2004.)

3.4.1 Vantagens

Segundo Fuzitaki (2004) a criptografia simétrica possui algumas vantagens as quais estão relacionadas a seguir:

a) os cifradores de simétricos possuem altas taxas de codificação, onde algumas implementações de software podem obter taxas de encriptação de *megabytes* por segundo, enquanto as implementações de hardware podem obter taxas centenas de *megabytes* por segundo;
b) as chaves são relativamente curtas;

c) podem ser utilizados para construção de diversos mecanismos, entre eles, funções hash, esquemas eficientes de assinaturas digitais, geradores de números randômicos, entre outros;
d) com transformações simples que são fáceis de analisar, podem ser utilizados para produzir cifradores poderosos;
e) a encriptação de chave simétrica possui uma longa história, onde muito do seu conhecimento se deve ao desenvolvimento do algoritmo DES no início da década de 70.

3.4.2 Desvantagens

Assim como existem vantagens na criptografia simétrica, também existem as desvantagens, as quais serão listas a seguir:

a) a chave deve sempre permanecer secreta, quando há comunicação entre dois elementos;
b) como em uma grande rede há um grande número de pares de chaves que precisam ser gerenciadas, é preciso que exista um terceiro confiável que possa ser confiável para o transmissor e para o receptor;
c) para que os dados sejam secretos é preciso que a chave seja mudada freqüentemente, talvez até em cada seção de comunicação;
d) os mecanismos de assinatura digital exigem grandes chaves.

3.4.3 Principais Algoritmos Simétricos

Os principais exemplos de algoritmos de criptografia simétricos são: DES, Triple – DES, IDEA, Skipjack, RC2, RC4, RC5, RC6, AES.

3.4.3.1 Triple DES

O Triple DES é uma simples variação do DES, utilizando-o em três ciframentos sucessivos, podendo empregar um versão com duas ou com três chaves diferentes. É seguro, porém muito lento para ser um algoritmo padrão (MORENO; PEREIRA; CHIARAMONTE, 2005).

Segundo Deliberador (2004) como foram construídas várias máquinas que quebravam as chaves do DES em menos de 24 horas, tentaram melhorar o algoritmo DES, chegando a solução que seria utilizar o algoritmo três vezes, resultando assim no Tripel DES. Este algoritmo utiliza três chaves de 56 *bits* cada, fazendo com que pessoas leigas no assunto imaginem que como o DES pode ser quebrado em 24 horas, então o Triple DES poderia ser quebrado em 72 horas. Mas isso só é possível se o mesmo souber que a chave quebrou, pois o Triple DES só permite que você saiba que quebrou a primeira chave, se você quebrar as três chaves.

Burnett e Paine (2002) afirmam que o Triple DES apresenta dois problemas, um deles é que descobriram formas de diminuir o espaço de chave de 168 *bits* para 108 *bits* reduzindo a quantidade de tentativas necessárias para um ataque de força bruta. Já o segundo problema é a velocidade, pois ele é três vezes mais lento, aumentando a quantidade de processamento necessária para realização das tarefas.

3.4.3.2 IDEA

O *International Data Encryption Algorithm* (IDEA) foi criado em 1991 por James Massey e Xuejia Lai e possui patente da suíça ASCOM Systec. Utiliza uma chave de 128 *bits* e tanto o texto puro como o criptografado, utilizam 64 *bits*, e foi projetado para ser eficiente em implementações de softwares, mas não foi desenvolvido para substituir o DES, mesmo seguindo as mesmas linhas dele.

O IDEA é utilizado principalmente no mercado financeiro, mas ainda não pode ser considerado como forte, devido ao seu pouco tempo de vida, mas aparenta ser robusto, devido a sua chave de 128 *bits* (DELIBERADOR, 2004).

3.4.3.3 RC2, RC4, RC5 e RC6

RC2 é um algoritmo de chave privada de tamanho variável, projetado por Ron Rivest, voltado para criptografia de e-mail corporativo. Proporciona uma criptografia em alto volume aliada a performance. É uma cifra de bloco muito similar ao DES, mas sua performance é duas vezes mais rápida que o mesmo (TERADA, 2000).

O algoritmo RC4 é uma cifra de corrente que permite que a chave possa variar de tamanho de 1 a 2048 *bits*. Sua vantagem sobre o DES é que a sua performance é 10 vezes mais rápida que o mesmo. Mas tanto o RC2 quanto o RC4 possuem códigos fáceis de serem quebrados, e não se tem informações precisas sobre a sua segurança com chaves extensas (DELIBERADOR, 2004).

O algoritmo RC5 também é chamado de *"Ron's Code"*, foi desenvolvido em 1994, e é bastante conhecido pela sua velocidade e simplicidade. Permite que o usuário defina o tamanho do bloco de dados, o tamanho da chave e o número de iterações necessárias para garantir a segurança e a performance do mesmo. Em poucas

linhas de código é possível implementar os procedimentos de cifragem e decifragem de maneira eficiente. O RC6 é um algoritmo simples que possui chave de 8 a 1024 *bits* podendo ser implementado facilmente de forma compacta tanto em *software* como em *hardware* (DELIBERADOR, 2004).

3.4.3.4 AES

Em 1997 o *National Institute of Standards in Technology* (NIST), anunciou um plano para definir um algoritmo que iria substituir o DES, pois esse não poderia mais garantir a segurança desejada por muito tempo, devido a evolução das máquinas (DELIBERADOR, 2004).

O NIST especificou que o *Advanced Encryption Standard* (AES), substituto do DES, deveria ter alguns requisitos funtamentais (MORENO, PEREIRA E CHIARAMONTE, 2005):

a) segurança forte: o algoritmo projetado deveria suportar ataques futuros, sem nenhuma fraqueza algorítmica;
b) projeto simples: facilitar a análise e certificação matemática da segurança oferecida pelo algoritmo;
c) desempenho: razoavelmente bom em uma variedade de plataformas;
d) não serem patenteados: os algoritmos devem ser de domínio público e estar disponíveis mundialmente;
e) tamanho: não utilizar muita memória.

O AES usa um número variável de tamanho de chave e tamanho de bloco, e seu código é bem enxuto não dependendo de nenhum outro tipo de componente criptográfico, fazendo com que seu nível de segurança seja maior (DELIBERADOR,

2004).

Várias pessoas físicas e jurídicas desenvolveram seus algoritmos e no dia 2 de outubro de 2000, o NIST anunciou que o grande vencedor seria o algoritmo Rijndael, desenvolvido por Vicent Rijmen e Joan Daemen (BURNETT; PAINE, 2002).

3.4.3.5 Skipjack

O Governo norte-americano patrocinou e propôs este algoritmo, por meio da Agência Nacional de Segurança (NSA), na década de 1980. Por seu projeto ter sido secreto, ele foi aguardado com entusiasmo pelo público, e em 1998 foi liberado para uso individual. Ele cifra bloco de 64 *bits* utilizando chave de 80 *bits*, dividindo a mensagem em quatro partes, estas sendo alternadamente permutadas por uma quantidade de vezes fixa nas suas regras (MORENO; PEREIRA; CHIARAMONTE, 2005).

3.5 CRIPTOGRAFIA ASSIMÉTRICA

Já na criptografia assimétrica, existem duas chaves, uma pública utilizada para criptografar e uma privada utilizada para descriptografar uma mensagem. Esse tipo de criptografia foi elaborado somente na década de 70, sendo o responsável pelo maior avanço na história da criptografia (FUZITAKI, 2004).

Por volta do ano de 1976 Whitfield Diffie e Martin Hellman, inventaram a criptografia de chaves públicas, a fim de resolver o problema de distribuição de chaves, pois utiliza chaves públicas e privadas. A chave pública é divulgada, enquanto a chave privada é mantida em segredo. Para mandar uma mensagem

privada, o transmissor cifra a mensagem usando a chave pública do destinatário pretendido, que deverá usar a sua respectiva chave privada para conseguir recuperar a mensagem original (MORENO, PEREIRA; CHIARAMONTE, 2004)

Esses algoritmos eliminam o problema dos algoritmos simétricos onde é necessário que as partes comunicantes troquem informações sigilosas, sendo que com esse algoritmo todas as comunicações irão envolver somente a chave pública (RIGHETTI, 2004).

3.5.1 Vantagens

Segundo Fuzitaki (2004) a criptografia assimétrica apresenta algumas vantagens que serão relacionadas a seguir:

a) nesse tipo de riptografia somente a chave privada deve ser mantida em segredo;
b) o par de chaves pública e privada pode permanecer inalterado por um período de tempo considerável, dependendo claro, do modo de uso dos usuários;
c) a chave que descreve a verificação pública é muito menor do que na criptografia simétrica, e são utilizadas por vários mecanismos de assinatura digital;
d) o número de chaves necessárias para uma rede extensa, pode ser consideravelmente menor que na criptografia simétrica.

3.5.2 Desvantagens

A criptografia assimétrica possui algumas desvantagens as quais estão relacionadas a seguir (FUZITAKI, 2004):

a) as taxas de codificação são mais lentas que na criptografia simétrica;
b) os tamanhos das chaves para criptografia assimétrica e para assinaturas digitais são relativamente muito maiores do que na simétrica;
c) não foi provado que os esquemas de chave pública são seguros;
d) criptografia de chave pública não tem uma extensa história, sendo descoberta apenas na metade da década de 70.

3.5.3 Principais algoritmos Assimétricos

Os principais exemplos de algoritmos de criptografia assimétricos são: RSA, DSA e DH, os quais são descritos abaixo:

3.5.3.1 DSA

O DSA é um algoritmo de assinatura digital que utiliza função de hash, e que recebeu diversas críticas como, tamanho de chave foi considerado pequeno tendo 512 *bits*, fatores políticos de distribuição do algoritmo relacionado a patentes e tempo de estudo insuficiente para garantir uma segurança (MORENO; PEREIRA; CHIARAMONTE, 2005).

3.5.3.2 DH

Foi desenvolvido por Whitfield Diffie e Martin Hellman, e é um algoritmo que não criptografa dados, somente gera um segredo. Sendo assim as duas partes geram um segredo e então utilizam para criar uma chave de sessão que será utilizada num algoritmo simétrico (DELIBERADOR, 2004).

4 ALGORITMO SIMÉTRICO DES

Desde que os sistemas de codificações foram criados, o algoritmo simétrico DES tornou-se popular entre seus usuários.

Em 1972 o *National Bureau of Standards* (NBS) concluiu um estudo sobre as necessidades de segurança de informação do governo norte-americano, publicando em 1973 um noticiário onde solicitava propostas de algoritmos de criptografia, para proteger os dados do governo americano durante as transmissões e armazenamentos, explicando a importância da cifragem dos dados. Como não obteve respostas o algoritmo foi proposto em 1974 pela IBM, com o nome de LUCIFER, mas que já havia sido desenvolvido no início de 1970 (MORENO; PEREIRA; CHIARAMONTE, 2005).

Esse algoritmo foi adotado como padrão em 1977 pelo NBS com ajuda do NIST e após algumas modificações, como a diminuição do tamanho da chave de 128 para 56 *bits*, este algoritmo passou a chamar-se DES (MORAES, 2004).

Segundo Righetti (2004) o algoritmo deveria respeitar algumas especificações:

a) alto nível de segurança;
b) estar totalmente documentado e ser de fácil entendimento;
c) a segurança do algoritmo deve se encontrar na chave e não depender do sigilo do algoritmo;
d) ser disponível para todos os usuários;
e) ser adaptável para uso em diversas aplicações;
f) deve ser economicamente implementável em dispositivos eletrônicos;
g) ser eficiente;
h) habilitado para validação;

i) ser exportável.

Quando o algoritmo DES foi criado, foi estipulado que a cada 5 anos ele seria revisto, sendo assim em 1987, o NSA comunicou que o DES deveria deixar de ser utilizado, pois com a evolução dos computadores ele seria comprometido devido ao seu tamanho de chave (DELIBERADOR, 2004).

O DES é um codificador composto que cifra blocos de 64 *bits* em blocos do mesmo tamanho, para isso se utiliza de uma chave composta por 56 *bits*, com 8 *bits* de paridade totalizando 64 *bits*. Os blocos que constroem o algoritmo são compostos de permutações e substituições (RIGHETTI, 2004).

O DES calcula primeiro 16 novas sub-chaves de 48 *bits* a partir da chave original de 56 *bits*, onde esse cálculo é feito por meio de 16 iterações idênticas em que cada iteração expande a metade da mensagem em 32 *bits* para direita e para esquerda, obtendo 48 *bits*. Após a 16ª iteração esses metades se juntam e é realizado uma permutação inversa à primeira. Nessas iterações os *bits* da chave deslocam-se a esquerda, e são combinados por um XOR, com os dados da direita após estes terem sido expandidos e permutados, e logo após são substituídos por 32 *bits* e novamente permutados (FUZITAKI, 2004).

Este processo de cifragem é apresentado na Figura 4.

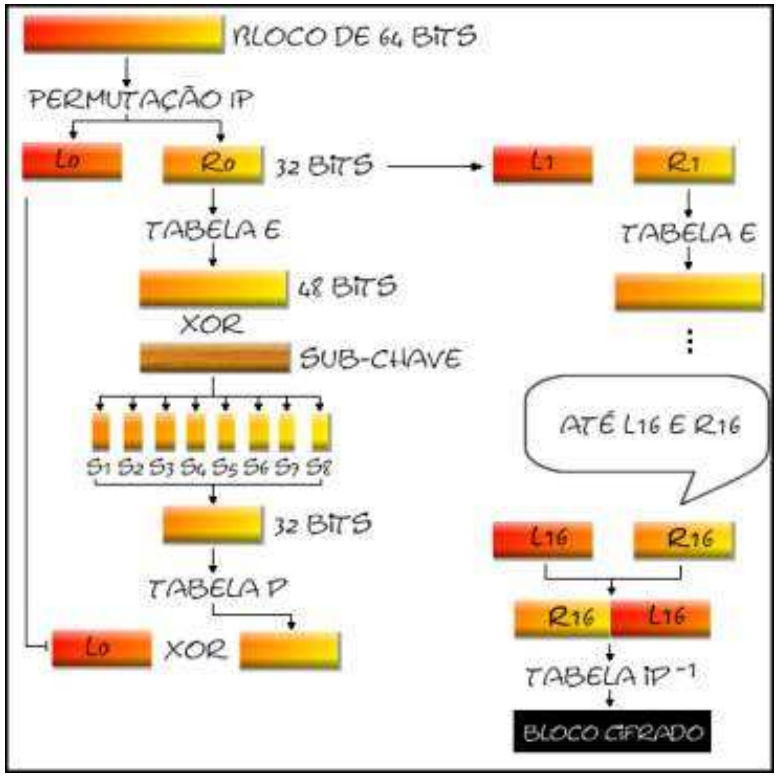

Figura 4. Processo de cifrar a mensagem no DES
Fonte: ARAÚJO, R.(2004, p. 21)

4.1 OBTER 16 SUB-CHAVES DE 48 *BITS* CADA

Para serem formadas as sub-chaves a serem utilizadas por cada iteração do algoritmo, as chaves são armazenadas com tamanho de 64 *bits*, sendo que cada oitavo bit é ignorado ou será utilizado para verificar a paridade da chave. Desta forma o tamanho da chave é reduzido para 56 *bits*. Para cada iteração do processamento principal, é gerada uma sub-chave, onde a partir da chave original são geradas 16 sub-chaves (MORENO; PEREIRA; CHIARAMONTE, 2005).

A Figura 5 representa a seqüência de passos para geração de sub-chaves.

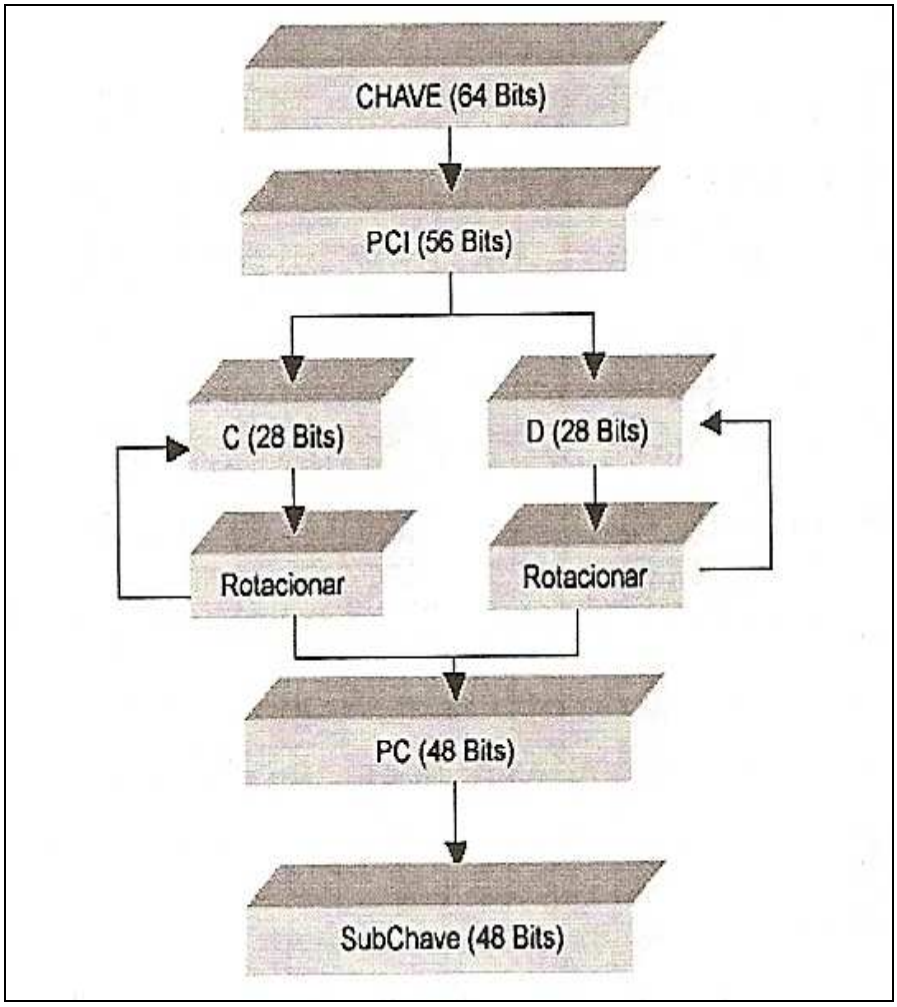

Figura 5. Processo de geração de sub-chaves do DES
Fonte: MORENO, E. PEREIRA, F. e CHIARAMONTE, R. (2005, p. 115)

Na Figura 5 pode observar que a chave original sofre uma permutação de compressão inicial de acordo com a Quadro 1:

Quadro 1. Permutação de compressão inicial

57	49	41	33	25	17	09	01	58	50	42	34	26	18
10	02	59	51	43	35	27	19	11	03	60	52	44	36
63	55	47	39	31	23	15	07	62	54	46	38	30	22
14	06	61	53	45	37	29	21	13	05	28	20	12	04

Observando-se o quadro 1, verifica-se que a primeira entrada é o número 57, que representa o 57º *bit* da chave original e que torna-se o primeiro *bit* da chave permutada, já o segundo bit da chave permutada representa o 49º bit da chave original.

O último *bit* da chave permutada é o 4º *bit* da chave original, aparecendo apenas 56 *bits* da chave original na chave permutada (MORENO; PEREIRA; CHIARAMONTE, 2005).

Logo após a permutação de compressão inicial, é feito um desmembramento do vetor resultante de 56 *bits* em duas partes iguais de 28 *bits*, sendo que a parte C ficará com os 28 *bits* menos significativos e a parte D com os 28 *bits* mais significativos. Logo em diante realiza-se uma rotação à esquerda das duas partes conforme o Quadro 2 a seguir:

Quadro 2. Vetor de rotação de chave

1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Logo após cada deslocamento é realizado a permutação de compressão, que concatena C e D formando uma sub-chave de 48 *bits*, gerando ao final 16 sub-chaves derivadas da chave original conforme mostra o Quadro 3:

Quadro 3. Permutação de compressão

14	17	11	24	01	05	03	28
15	06	21	10	23	19	12	04
26	08	16	07	27	20	13	02
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

4.2 OBTER BLOCOS CODIFICADOS DE CADA 64 BITS DA MENSAGEM

Ao começar é separado do texto original um bloco de 64 *bits*, e se esse bloco for menor do que 64 *bits* ele é preenchido com zeros. A permutação inicial não influi na segurança do DES, e tem o propósito de facilitar o uso do mesmo (RIGHETTI, 2004).

Quadro 4. Permutação de inicial

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

A permutação de expansão é realizada sobre 32 *bits*, onde o vetor resultante será de 48 *bits* representados no Quadro 5 a seguir:

Quadro 5. Permutação de expansão

32	01	02	03	04	05	04	05	06	07	08	09
08	09	10	11	12	13	12	13	14	14	16	17
16	17	18	19	20	21	20	21	22	23	24	25
24	25	26	27	28	29	28	29	30	31	32	01

A Xor com sub-chave realiza as operações de XOR entre o vetor resultante da permutação de expansão juntamente com a sub-chave da iteração que está sendo executada (DELIBERADOR, 2004).

As S-BOXs também chamadas de S-BOXES são oito quadros que são denominados de caixas de substituição, onde substituem 8 blocos de 6 *bits* que são criados ao dividir os 48 *bits* do XOR. São formadas por 4 linhas e 16 colunas, onde cada grupo atuará sobre uma S-BOX obedecendo a ordem em que os primeiros 6 *bits* são substituídos por algum valor da S-BOX 1, e assim por diante (MORENO; PEREIRA; CHIARAMONTE, 2005).

Os quadros abaixo representam as S-BOX:

Quadro 6. S-BOX 1

14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

Quadro 7. S-BOX 2

15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

Quadro 8. S-BOX 3

10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08
13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07
01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

Quadro 9. S-BOX 4

07	13	14	03	00	06	09	10	01	02	08	05	11	12	04	15
13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

Quadro 10. S-BOX 5

02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
15	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

Quadro 11. S-BOX 6

12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
04	03	02	12	09	05	15	10	11	14	01	07	05	00	08	13

Quadro 12. S-BOX 7

04	11	02	14	15	00	08	13	03	12	09	07	05	10	06	01
13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

Quadro 13. S-BOX 8

13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
01	15	13	08	10	03	07	04	12	05	06	11	00	14	09	02
07	11	04	01	09	12	14	02	00	06	10	13	15	03	05	08
02	01	14	07	04	10	08	13	15	12	09	00	03	05	06	11

Segundo Moreno, Pereira e Chiaramonte (2005) o P-BOX é a concatenação das S-BOX, onde aplica-se a permutação comum apresentada no Quadro 14 a seguir:

Quadro 14. P-BOX

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

Após serem feitas as 16 iterações, realiza-se uma permutação final, o qual não influencia muito na segurança, pois não tem relação com a chave criptográfica (MORENO, PEREIRA E CHIARAMONTE, 2005).

Quadro 15. Permutação de final

40	08	48	16	56	24	64	32
39	07	47	15	55	23	63	31
38	06	46	14	54	22	62	30
37	05	45	13	53	21	61	29
36	04	44	12	52	20	60	28
35	03	43	11	51	19	59	27
34	02	42	10	50	18	58	26
33	01	41	09	49	17	57	25

Após todas as etapas acima listadas, as sub-chaves são criadas, e serão utilizadas pelo algoritmo para que seja feita a cifragem e decifragem do arquivo.

Pode-se observar melhor esses passos na figura 7 a seguir:

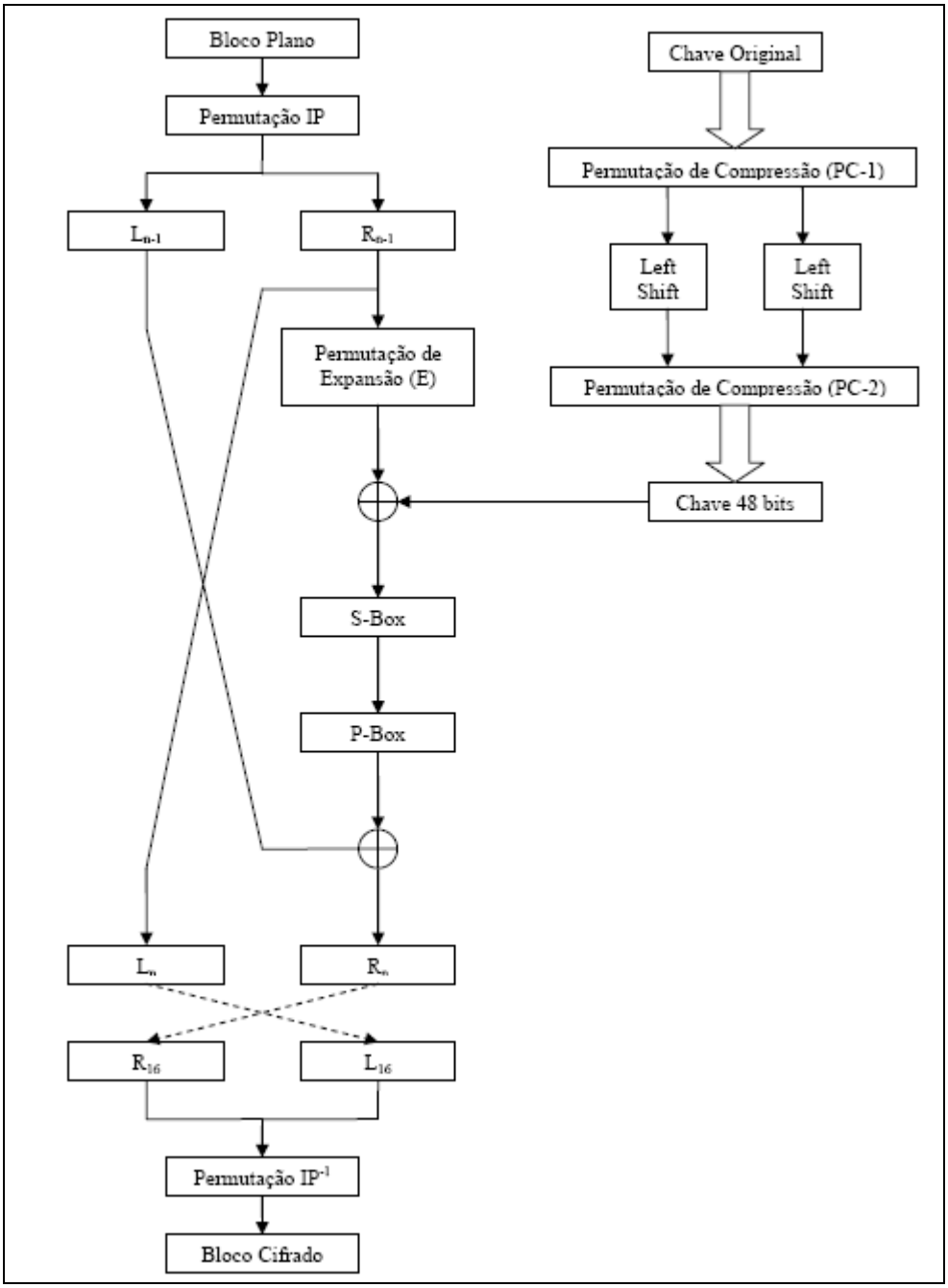

Figura 6. Ciclo DES de codificação de blocos de mensagem
Fonte: RIGHETTI, F. (2004, p. 18)

Observando-se todas as etapas do algoritmo DES, pode-se verificar como é difícil para que o algoritmo seja criado, e possa criptografar o arquivo com segurança, o mesmo acontece com o algoritmo RSA.

5 ALGORITMO ASSIMÉTRICO RSA

Por volta do Ano de 1977, os professores Ron Rivest e Adi Shamir do *Massachusetts Institute of Technology* (MIT) e o professor Leonard Adleman, criaram o algoritmo assimétrico RSA, que possui este nome devido a seus inventores, que também são fundadores da empresa RSA Data Security. Até o ano de 2005, ele era considerado o algoritmo de chave pública mais amplamente utilizado, além de ser uma das mais poderosas formas de criptografia de chave pública conhecidas (MORENO; PEREIRA; CHIARAMONTE, 2005).

De acordo com Moreno, Pereira e Chiaramonte (2005) “o sistema consiste em gerar uma chave pública (geralmente utilizada para cifrar os dados) e uma chave privada (utilizada para decifrar os dados) por meio de números primos grandes, o que dificulta a obtenção de uma chave a partir de outra.”

Para a geração das chaves e para uma maior segurança dos dados, é preciso que os números primos que serão escolhidos sejam grandes, sendo que geralmente os mesmos têm 512 *bits* de comprimento e se combinados formam chaves de 1.024 *bits*. Em aplicações bancárias, essas chaves chegam a ter 2.048 *bits*. Com os passar dos anos, é provável que esse número aumente devido ao avanço dos sistemas computacionais que são capazes de fatorar chaves maiores em pouco tempo (MORENO; PEREIRA; CHIARAMONTE, 2005).

A chave pública pode ser conhecida por todos os usuários, diferentemente da chave privada que deve ser mantida em segredo, sendo essa a principal vantagem da criptografia assimétrica. Somente a chave privada do usuário que criou as duas chaves, pública e privada, poderá decifrar o texto criptografado com a chave pública que foi disponibilizada a outro usuário. Ou seja, mesmo que a mensagem original chegue a um

desconhecido, ela não poderá ser decifrada sem a chave privada (MORENO; PEREIRA; CHIARAMONTE, 2005).

É necessário seguir as seguintes etapas para a geração das chaves pública e privada no algoritmo RSA:

1. Escolhem-se dois números primos grandes (*p* e *q*), sendo que sejam diferentes;
2. Gera-se um número *n* por meio da multiplicação dos números escolhidos anteriormente (n = p.q), sendo que os fatores *p* e *q* devem ser mantidos em segredo;
3. Escolhe-se um número *d*, tal que *d* é menor que *n*. Ou seja, d é relativamente primo a n. Neste caso, d = (p - 1).(q - 1);
4. Escolhe-se um número *e* tal que (ed-1) seja divisível por (p-1).(q-1). Para realizar esse cálculo é necessário o algoritmo de Euclides estendido. Os valores *e* e *d* são chamados de expoentes público e privado. O par (n, e) é a chave pública e pode ser publicado e o par (n, d) é a chave privada que deve ser mantida em segredo.

Com esse algoritmo para poder cifrar uma mensagem, realiza-se o seguinte cálculo:

$$C = T^e \text{ (MOD n)}$$

Onde:

a) C como mensagem cifrada;

b) T é o texto original;

c) *e* e *n* são dados a partir da chave pública (n, e).

Utiliza-se a chave privada (n, d) por meio do seguinte cálculo, para decifrar a mensagem C:

$$T = C^d \text{ (MOD n)}$$

O exemplo utilizado nesta pesquisa foi extraído do livro Criptografia em Software e Hardware**,** onde de acordo com Moreno, Pereira e Chiaramonte (2005) a mensagem utilizada para que o algoritmo RSA realize a criptografia foi " ITS ALL GREEK TO ME" utilizada pelos autores do RSA no artigo "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems" (RIVEST et al., 1978), com chaves diferentes.

Após realizar a etapa de criação de chaves, chamada de etapa 1, que foi descrita anteriormente, escolhe-se dois números primos aleatórios, no caso foram escolhidos para esse exemplo os seguintes valores: $p = 53$ e $q = 61$.

Seguindo a etapa 2 tem-se $n = p.q$ que resulta em $n = 53.61 = 3233$.

Na etapa 3 escolhe-se um número primo aleatório maior que *p* e *q*, sendo que o número *d* seja menor que n e relativamente primo a (*p*-1).(*q*-1), sendo escolhido o $d = 193$ para este exemplo.

Na etapa 4 é preciso escolher um número *e* tal que (*ed*-1) seja divisível por *(p-1).(q-1)*, sendo que $e = 97$.

Os valores importantes no algoritmo RSA para esse exemplo são:

p = 53 *q = 61* *n = 3233* *d = 193* *e = 97*

Já no artigo, as chaves que foram utilizadas são:

p = 47 *q = 59* *n = 2773* *d = 157* *e = 17*

Para representar as letras maiúsculas do alfabeto numericamente adotou-se o Quadro 15 a seguir:

Quadro 16. Representação numérica das letras do alfabeto

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

Sendo assim a frase “ITS ALL GREEK TO ME" seria representada numericamente como:

09 20 19 00 01 12 12 00 07 18 05 05 11 00 20 15 00 13 05 00

Como o valor máximo que pode aparecer na frase original é o equivalente à seqüência “ZZ” = 2626 e o valor de *n* nesse exemplo é 3233, a mensagem pode ser criptografada em duas letras, pois a seqüência é menor que *n*. A mensagem agrupada em duas letras ficará da seguinte forma:

0920 1900 0112 1200 0718 0505 1100 2015 0013 0500

O tabela 1 a seguir, representa o cálculo de cifragem mencionado acima:

Tabela 1. Representação do cálculo de cifragem

T	***T^e (MOD n)***	***C***
0920	0920^{97} MOD 3233	2546
1900	1900^{97} MOD 3233	1728
0112	0112^{97} MOD 3233	0514
1200	1200^{97} MOD 3233	0210
0718	0718^{97} MOD 3233	2304
0505	0505^{97} MOD 3233	0153
1100	1100^{97} MOD 3233	2922
2015	2015^{97} MOD 3233	2068
0013	0013^{97} MOD 3233	1477
0500	0500^{97} MOD 3233	2726

A mensagem cifrada fica:

2546 1728 0514 0210 2304 0153 2922 2068 1477 2726

Realiza-se o cálculo a seguir, para decifrar cada bloco:

$$T = C^d \text{ (MOD n)}$$

Onde:

a) T é a mensagem original

b) C, a mensagem cifrada.

c) *d* e *n* são dados a partir da chave pública (n, d).

A seguir o tabela 2, representa o cálculo de decifragem de blocos acima mencionado:

Tabela 2. Representação do cálculo de decifragem

C	***C^d (MOD n),***	***T***
2546	2546^{193} MOD 3233	0920
1728	1728^{193} MOD 3233	1900
0514	0514^{193} MOD 3233	0112
0210	0210^{193} MOD 3233	1200
2304	2304^{193} MOD 3233	0718
0153	0153^{193} MOD 3233	0505
2922	2922^{193} MOD 3233	1100
2068	2068^{193} MOD 3233	2015
1477	1477^{193} MOD 3233	0013
2726	2726^{193} MOD 3233	0500

A mensagem decifrada fica igual mensagem original que foi representada numericamente e utilizada no começo deste exemplo:

0920 1900 0112 1200 0718 0505 1100 2015 0013 0500

Pode-se verificar que o algoritmo RSA tem o entendimento mais fácil dos processos de cifragem e decifragem, do que o algoritmo DES.

6 PESQUISAS QUE UTILIZAM A TÉCNICA DE CRIPTOGRAFIA

Este capítulo tem por finalidade mostrar alguns trabalhos que estudaram diversas técnicas de criptografia, entre outros.

6.1 AVALIAÇÃO DE DESEMPENHO DE ALGORITMOS DE CRIPTOGRAFIA

Trabalho de Conclusão de Curso desenvolvido na Universidade Estadual de Londrina, no Centro de Ciências Exatas e Departamento de Ciência da Computação no Ano de 2004, por Gerson Yoshio Fuzitaki.

"Este trabalho possui como objetivo a análise de desempenho dos principais algoritmos de criptografia. Para isto, primeiro se descrevem os conceitos envolvidos na criptografia e algumas aplicações. Em seguida são enumeradas as principais diferenças entre a criptografia assimétrica e a simétrica. Durante o decorrer do texto, são detalhados os algoritmos RSA e DES e mostra idéia básica do DSA e da *Elliptic Curve Cryptography* (ECC).

Por fim, as principais diferenças de segurança e desempenho entre diferentes algoritmos são explicitadas por meio de tabelas. Nelas é possível perceber que a ECC pode possuir desempenho superior ao RSA. Com isso, torna-se evidente que a ECC pode vir a se tornar o novo padrão da criptografia assimétrica num futuro breve" (FUZITAKI, 2004).

6.2 O IMPACTO DO USO DA CRIPTOGRAFIA NO THROUGHPUT DE REDES LOCAIS: UM ESTUDO DE CASO USANDO O ALGORITMO TRIPLE-DES

Monografia desenvolvida no Curso de Ciência da Computação da Faculdade de Ciências Aplicadas de Cascavel – FACIAP, no ano de 2004 por Fabiano Reese Righetti.

"Este trabalho realiza uma análise comparativa entre o envio de dados não criptografados e dados criptografados em redes locais com o intuito de medir o impacto causado no *throughput*. O algoritmo de criptografia simétrica escolhido para esta análise foi o Triple-DES por se tratar de um algoritmo seguro, simples, amplamente utilizado e relativamente rápido. Foi adotado um ambiente de simulação utilizando-se de dois microcomputadores interligados por meio de uma rede 10/100 BaseT, um sistema para geração de datagramas em texto puro e criptografados desenvolvido na linguagem de programação C, ferramenta para análise do tráfego na rede (TCPDUMP) e o sistema operacional GNU/Linux.

Os testes realizados buscaram simular um ambiente real sendo a rede isolada e todas as características dos computadores mantidas durante a geração dos dois tipos de datagramas. Com base nos resultados alcançados pode-se notar uma elevada diferença no volume de pacotes enviados nas duas formas, cerca de 60%. Ficando claro que por mais que o algoritmo seja simples e relativamente rápido o *throughput* é afetado, sendo de elevada importância fazer um estudo de qual algoritmo adotar em um projeto e analisar formas de aumentar a segurança sem degradar o desempenho do mesmo"(RIGHETTI, 2004).

6.3 UM COMPONENTE COMPUTACIONAL PARA AUXILIAR O DESENVOLVIMENTO DE UMA ASSINATURA DIGITAL NO SISTEMA DE INFORMAÇÕES PROCESSUAIS

Dissertação de Mestrado desenvolvido na Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia de Produção, no Ano de 2004, por Paulo de Tarso Deliberador.

“Este trabalho apresenta conceitos relacionados à assinatura digital, e a necessidade de garantir a confiabilidade e autenticidade de informações em ambientes computacionais. O intuito da assinatura digital é fazer com que os documentos digitais tenham o mesmo valor legal do que um documento escrito e assinado de maneira comum, com a finalidade de garantir a autoria e autenticar a origem dos dados contidos no documento.

Esta dissertação tem como objetivo apresentar um componente computacional para auxiliar programadores e analistas de sistemas, facilitando a aplicação da técnica de assinatura digital na integração do Sistema de Informações Processuais (SIP), desenvolvido pelo DesignLAB para aplicação na Justiça Trabalhista Brasileira em parceria com o Departamento de Expressão Gráfica da Universidade Federal de Santa Catarina.

Com o componente computacional proposto, pretende-se facilitar a implementação da assinatura digital, aperfeiçoando a segurança do sistema atual, garantindo a integridade e autenticação informatizada dos documentos criados ou alterados pelos usuários do sistema” (DELIBERADOR, 2004).

6.4 INTRODUÇÃO A CRIPTOGRAFIA

Artigo desenvolvido na Universidade Católica de Pelotas-RS na Escola de Informática, por Eduardo N. F. Bastos.

"Este artigo introduz conceitos básicos de criptografia e em especial do método criptográfico RSA. Tem o objetivo de explicar o funcionamento dos diferentes tipos de criptografia mais conhecidos e seus modos de operação, fornecendo uma boa base teórica e prática destes métodos largamente utilizados na computação" (BASTOS).

6.5 ESTUDO DA TÉCNICA DE CRIPTOGRAFIA ALGORITMO POSICIONAL – ALPOS NA SEGURANÇA DOS DADOS DE UM BANCO DE DADOS

Monografia desenvolvida nas Faculdades Santo Agostinho como – FASA e Faculdade de Ciências Exatas e Tecnológicas – FACET no curso de sistema de informação no ano de 2006 por Fabiano Otávio de Faria.

"Este trabalho tem como objetivo o estudo da técnica de criptografia Algoritmo Posicional (ALPOS), como alternativa para prover segurança adicional aos dados armazenados em um banco de dados qualquer, por meio da criptografia dos dados. Para tanto, foi feito um estudo acerca das técnicas de criptografia DES e ALPOS que foi a técnica utilizada no desenvolvimento deste trabalho, sendo que foram propostos algoritmos para criptografar e descriptografar dados de um banco de dados, baseados no ALPOS. Por fim, é apresentado um relatório com os resultados obtidos e as respectivas conclusões" (FARIA, 2006).

7 COMPARAÇÃO ENTRE OS ALGORITMOS DE CRIPTOGRAFIA DES E RSA

Devido ao grande número de usuários que necessitam de segurança um dos objetivos dessa pesquisa é de se realizar a comparação dos dois principais algoritmos estudados, no caso o DES e o RSA. Contudo este capítulo não tem a pretensão de esgotar todas as possibilidades de investigação relacionadas aos mesmos, mas traz o início de uma explanação do que foi estudado, e os resultados dos testes aplicados no decorrer da pesquisa.

O desenvolvimento desta pesquisa foi iniciado com um levantamento bibliográfico com auxilio de livros, e muitos artigos científicos, como também, monografias e trabalhos de conclusão de curso. Portanto foram estudados os requisitos de criptografia, os principais algoritmos empregados, sua história, os algoritmos RSA e DES, as vantagens e desvantagens de cada um, entre muitos outros assuntos relacionados á mesma.

A partir dos estudos realizados compreendeu-se como o processo do tratamento da criptografia e as ferramentas utilizadas para a criação da mesma, são importantes em todo o processo de cifragem e decifragem, tornando mais fácil o entendimento do funcionamento dos algoritmos estudados.

Sendo o tamanho das chaves e o tempo de cifragem e decifragem os aspectos utilizados para fazer a comparação entre os algoritmos escolhidos, foram utilizados dois softwares de criptografia, um utilizando o algoritmo DES e o outro utilizando o algoritmo RSA, que serão explicados mais adiante. Para essas comparações entre os mesmos, foram utilizados arquivos de diversos tipos de tamanhos e extensões e também chaves de diversos tamanhos.

Nos testes realizados foi utilizado notebook com a configuração representada na tabela 3 abaixo:

Tabela 3. Configuração dos Computadores utilizados nos testes

Item	*Configuração*
CPU	Intel Celeron M Processador 420 (1.6 Ghz)
Memória	896 MB de Ram
HD	120 GB
S.O.	Windows XP spk 2

O ambiente escolhido para utilização e pesquisa de softwares, foi o sistema operacional Windows, devido ser de utilização maior pelos usuários, do que os demais sistemas operacionais existentes. Alguns softwares utilizando as técnicas estudadas foram encontrados, como por exemplo o TrueCrypt 5.1, AB File Encrytor, Cryptopp entre outros, mas por serem de difícil entendimento ou mesmo por não possuírem um manual de ajuda, ou até mesmo por não realizarem os processos necessários, dificultavam os estudos. Assim como esses não foram os softwares escolhidos para essa pesquisa, não foi considerado necessário fazer um estudo maior sobre os mesmos e documentá-los. Sendo assim, foram escolhidos dois softwares de língua portuguesa, de fácil entendimento ao usuário.

Os softwares utilizados em todos os testes foram o JR. CRIPTO DES versão 1.0 e JR. CRIPTO versão 1.0 criados por Flávio de Souza no ano de 2008 na Cidade de Criciúma em Santa Catarina e implementados na linguagem delphi, sendo que os mesmos utilizam os algoritmos RSA e DES respectivamente.

Tanto no JR Cripto como no JR Cripto Des o tempo é disparado quando o botão criptografar ou descriptografar é acionado, logo após ele associa a chave digitada e termina a contagem logo que as funções são encerradas, mostrando o tempo final.

A seguir encontra-se uma parte do código fonte, onde a função realiza o processo da criptografia com o algoritmo RSA e com o algoritmo DES, e encontra-se nos softwares utilizados:

Função que realiza o processo de Criptografia com o Algoritmo DES

```
// criptografar e descriptografar

CriptografaButton   .Enabled := False;
DescriptografaButton.Enabled := False;

// encriptar fica TRUE se pressionar o botão criptografar(CriptografaButton) e FALSE
se for o descriptografar(DescriptografaButton)

Encriptar := TComponent(Sender).Tag = 1;

// salvamos o momento atual
T  := Now;

  try

// atualiza chaves de criptografia com base na palavra

    GenerateLMDKey(Key64 , SizeOf(Key64 ), PalavraEdit.Text);
    GenerateLMDKey(Key128, SizeOf(Key128), PalavraEdit.Text);

// dependendo do tipo de criptografia escolhido criptamos o arquivo:

    case TipoCombo.ItemIndex of
      0: DESEncryptFile         (EntradaEdit.Text, SaidaEdit.Text, Key64 , Encriptar);
      1: DESEncryptFileCBC      (EntradaEdit.Text, SaidaEdit.Text, Key64 , Encriptar);
      2: TripleDESEncryptFile   (EntradaEdit.Text, SaidaEdit.Text, Key128, Encriptar);
      3: TripleDESEncryptFileCBC(EntradaEdit.Text, SaidaEdit.Text, Key128,
Encriptar);
    end;
  except
    on E:Exception do
      ShowMessage('Ocorreu um erro ao tentar criptografar. Mensagem original: ' +
e.Message);
  end;

 // essa função mostra o tempo que levou:
  TempoLabel.Caption := 'Tempo: ' + FormatDateTime('NN"m":SS"s",ZZZ', Now - T);

  CriptografaButton   .Enabled := True;
  DescriptografaButton.Enabled := True;
  ShowMessage('Processo concluído.');
end;
```

Função que realiza o processo de Criptografia com o Algoritmo RSA

```
// criptografar e descriptografar
CriptografaButton   .Enabled := False;
DescriptografaButton.Enabled := False;

// encriptar fica TRUE se pressionar o botão CriptografaButton e FALSE se for o
DescriptografaButton
Encriptar := TComponent(Sender).Tag = 1;

 // salvamos o momento atual
 TempoLabel.Caption := 'Contando tempo...';
 T  := Now;
 try
    if PrivateKey <> nil then
      RSAComp.PrivateKey.Assign(PrivateKey);

    if PublicKey <> nil then
      RSAComp.PublicKey .Assign(PublicKey);

 // função que criptografa o arquivo:
    if Encriptar then
    begin
      MostrarStatus('Criptografando... aguarde, pode levar alguns minutos...');
      RSAComp.EncryptFile(EntradaEdit.Text, SaidaEdit.Text);
    end
    else

// função que descriptografa o arquivo:
    begin
      MostrarStatus('Descriptografando... aguarde, pode levar alguns minutos...');
      RSAComp.DecryptFile(EntradaEdit.Text, SaidaEdit.Text);
    end;
  except
    on E:Exception do
      ShowMessage('Ocorreu um erro ao tentar criptografar/descriptografar. Mensagem
original: ' + e.Message);
  end;

  // limpa a msg de status
  MostrarStatus('');

  // mostra o tempo que levou:
  TempoLabel.Caption := 'Tempo: ' + FormatDateTime('NN"m":SS"s",ZZZ', Now - T);

  CriptografaButton   .Enabled := True;
  DescriptografaButton.Enabled := True;
  ShowMessage('Processo concluído.');
end;
```

O funcionamento dos softwares é simples, sendo fáceis de serem utilizados.

O JR. Cripto DES, contém um campo chamado TIPO onde se escolhe o algoritmo que se quer utilizar, um campo chamado PALAVRA CHAVE onde digita-se a palavra chave que será utilizada para criptografar e descriptografar o arquivo, um campo chamado ARQUIVO DE ENTRADA onde se coloca o arquivo original que se quer criptografar ou o arquivo criptografado para descriptografar e um campo chamado ARQUIVO DE SAÍDA onde se coloca o nome do arquivo que será salvo quando se clicar os botões criptografar ou descriptografar.

Ao iniciar o software escolhe-se o algoritmo que será utilizado na operação, no caso o DES, digita-se a palavra-chave e após escolhido o arquivo original, que encontra-se no arquivo de entrada, para que seja cifrado, coloca-se o nome e o caminho que será salvo o arquivo no campo arquivo de saída, e clica-se no botão criptografar. Feito isso o software calcula o tempo utilizado para a ação e salva o arquivo, impedindo que o usuário tente acessá-lo sem a utilização de softwares e chaves, o qual não conseguirá acesso, pois o arquivo estará cifrado.

Para descriptografar o usuário utiliza a mesma palavra chave e o mesmo algoritmo e no campo arquivo de entrada coloca o arquivo salvo que encontra-se cifrado, escolhe o nome e o caminho que será salvo o arquivo decifrado no campo arquivo de saída e clica-se no botão descriptografar. Após feito essas ações, o software novamente calcula o tempo utilizado para esse processo e salva o arquivo, permitindo que o usuário acesse sem utilização de softwares e agora podendo visualizar o arquivo como o original.

Caso tente-se utilizar uma palavra chave para descriptografar diferente da chave que foi utilizada para criptografar o software não mostra uma mensagem que diz que a palavra chave está incorreta, ele faz o processo normal, mas se o usuário tentar

acessar o arquivo não conseguirá, pois ele continuará cifrado. Para visualização do software, a figura 7 a seguir, mostra a interface do mesmo:

Figura 7. Interface do Programa JR Cripto DES

É possível observar na figura acima que, o software é de fácil entendimento e utilização, permitindo que qualquer pessoa possa fazer testes e utilizá-lo da forma que escolher.

O software JR Cripto que utiliza o algoritmo assimétrico RSA, tem o funcionamento diferente e um pouco mais complexo, mas também fácil de ser utilizado.

Esse software contém os seguintes campos: CHAVE PUBLICA onde o usuário digita uma senha que será disponibilizada para o outro usuário que receber o arquivo criptografado, CHAVE PRIVADA na qual o usuário digita a chave que somente ele saberá, TAMANHO DA CHAVE onde é possível escolher o tamanho da senha que o programa irá gerar, PRIMO BASE é o número primo no qual o software utilizará para criação das chaves públicas e privadas, CHAVE PUBLICA e CHAVE PRIVADA nesses campos o software preenche automaticamente com a chave pública e privada que o mesmo gerou, ARQUIVO DE ENTRADA no qual se coloca o arquivo original que se quer criptografar ou o arquivo criptografado para descriptografar e um campo chamado ARQUIVO DE SAÍDA onde se coloca o nome do arquivo que será

salvo quando se clicar os botões criptografar ou descriptografar igualmente como no JR Cripto DES.

Ao iniciar o software, é preciso seguir alguns passos para geração das chaves, criptografar e descriptografar um arquivo, os quais estão relacionados a seguir.

Para a criação de chaves o usuário digita uma chave pública e uma privada, sendo que a chave pública será disponibilizada para o usuário que receber o arquivo criptografado e a chave privada, como o nome já diz, somente o autor do arquivo criptografado saberá. A partir daí escolhe-se o tamanho da chave, e um número primo que o software irá utilizar para multiplicação e geração das mesmas. Então pressiona-se o botão "Gerar Chave", para que o software automaticamente possa gerar as chaves pública e privada que irão aparecer nos campos CHAVE PÚBLICA E CHAVE PRIVADA. Ao aparecer as chaves criadas, clica-se em salvar em cada campo para que salve a chave privada e pública em arquivos separados.

Após criadas as chaves, é preciso que seja enviado a chave pública gerada e também a que o usuário digitou para a pessoa que irá receber o arquivo criptografado. Ao receber a chave o usuário digita a mesma, e abre o arquivo de chave pública que foi salvo, seleciona o arquivo de entrada que será criptografado e digita um nome no arquivo de saída o qual será salvo o arquivo e pressiona o botão "Criptografar". Feito isso o software calcula o tempo utilizado para a ação e salva.

Ao receber o arquivo cifrado, o usuário terá que digitar a chave privada e abrir o arquivo de chave privada criado pelo software, salvo quando gerou as chaves, para descriptografar o arquivo recebido. No arquivo de entrada o usuário colocará o arquivo cifrado e no arquivo de saída irá digitar o nome que o arquivo decifrado receberá, e então pressiona o botão "Descriptografar". Feito essas ações, o software novamente calcula o tempo utilizado para esse processo e salva o arquivo, permitindo

que o usuário acesse sem utilização de softwares e agora podendo visualizar o arquivo como o original.

Caso o usuário tente utilizar uma palavra chave para descriptografar diferente da chave que foi utilizada para criptografar, ou utilizar um arquivo público ou privado diferente do que foi criado, o software mostra uma mensagem dizendo que a chave está incorreta e não deixará descriptografar o arquivo. Para visualização do software as figuras 8 e 9 a seguir, mostram a interface do mesmo:

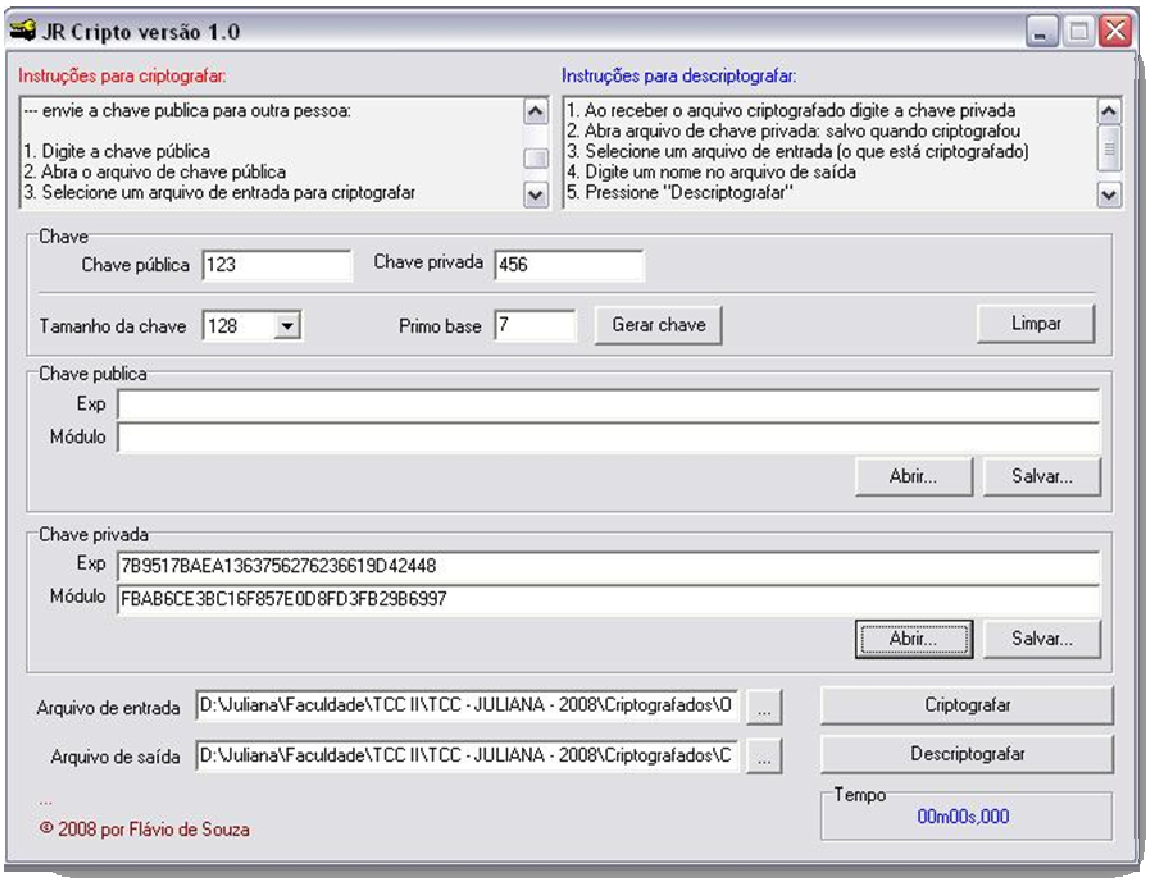

Figura 8. Interface do Programa JR Cripto utilizando a Chave Privada

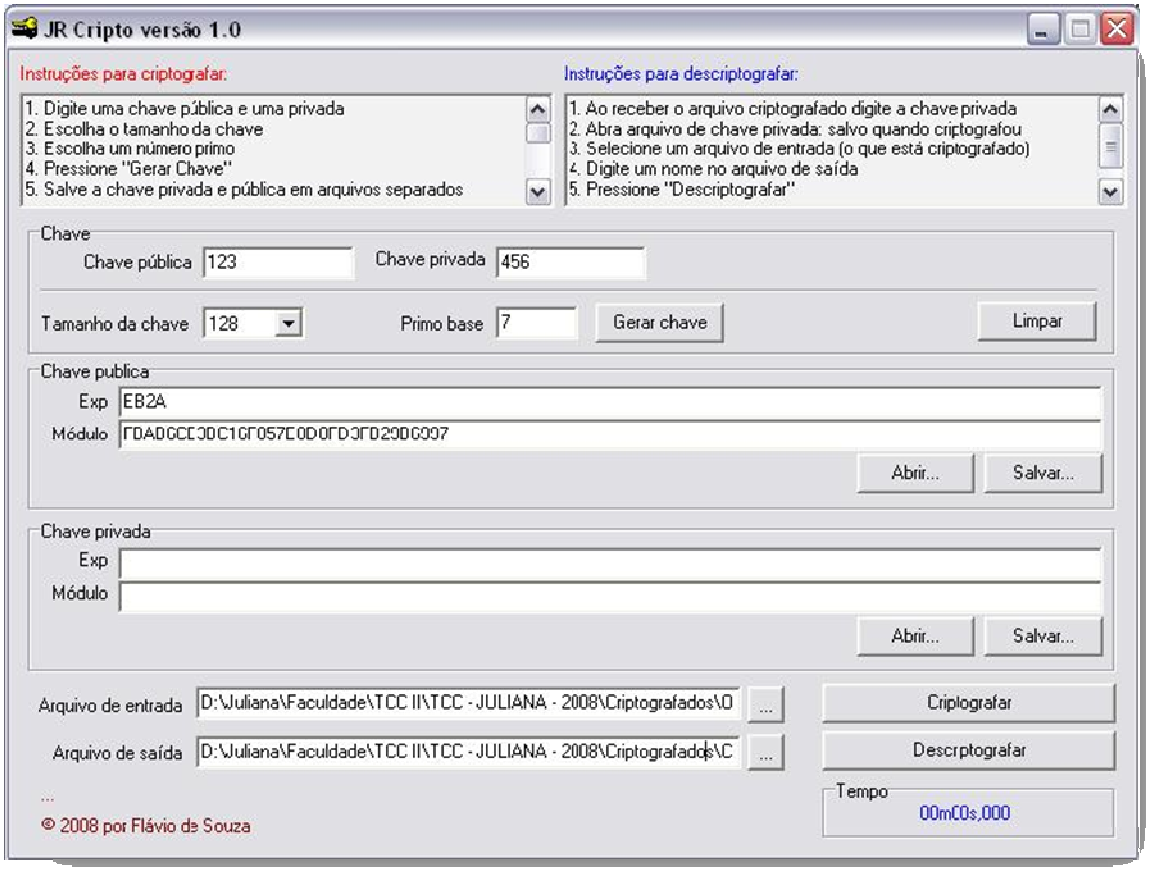

Figura 9. Interface do Programa JR Cripto utilizando a Chave Pública

Após escolhidos os arquivos que seriam utilizados nos testes, sendo eles de vários tamanhos e de várias extensões, iniciou-se a bateria de testes com os dois programas. Com cada um deles foram repetidos os testes 30 vezes, mas apenas os 10 que apresentam mais diferenças, estão representados em forma de quadros nos anexos ao final da pesquisa. Após serem realizados, foi feito uma média de tempo de cifragem e decifragem utilizando os 10 testes escolhidos com cada um dos algoritmos, que podem ser analisados nas tabelas 4 e 5 a seguir.

Tabela 4. Média de Tempo de Criptografia com o Algoritmo DES

Nr. Consulta	*Extensão do Arquivo*	*Tamanho KB*	*Cifragem*	*Decifragem*
01	MPEG	42.958	01m 31s 084ms	01m 27s 114ms
02	MP3	19.332	00m 40s 042ms	00m 39s 993ms
03	EXE	15.365	00m 40s 749ms	00m 40s 238ms
04	PPT	7.559	00m 15s 933ms	00m 15s 718ms
05	MP3	5.355	00m 10s 999ms	00m 11s 479ms
06	JPEG	4.667	00m 09s 644ms	00m 09s 926ms
07	MPEG	3.580	00m 07s 444ms	00m 07s 845ms
08	PDF	2.181	00m 04s 525ms	00m 04s 541ms
09	JPEG	1.969	00m 04s 064ms	00m 03s 973ms
10	DOC	949	00m 01s 966ms	00m 01s 975ms
11	DOC	587	00m 01s 273ms	00m 01s 297ms
12	PPS	441	00m 00s 925ms	00m 00s 961ms
13	PDF	293	00m 00s 583ms	00m 00s 588ms
14	XLS	48	00m 00s 100ms	00m 00s 098ms
15	XLS	20	00m 00s 041ms	00m 00s 047ms
16	JPEG	16	00m 00s 034ms	00m 00s 046ms
17	TXT	8	00m 00s 017ms	00m 00s 025ms

Tabela 5. Média de Tempo de Criptografia com o Algoritmo RSA

Nr. Consulta	*Extensão do Arquivo*	*Tamanho KB*	*Cifragem*	*Decifragem*
01	MPEG	42.958	2h 13m 67s 511ms	1h 22m 55s 557ms
02	MP3	19.332	57m 47s 908ms	1h 15m 43s 686ms
03	EXE	15.365	47m 01s 807ms	1h 27m 55s 624ms
04	PPT	7.559	23m 43s 177ms	1h 18m 49s 498ms
05	MP3	5.355	16m 19s 083ms	1h 13m 20s 829ms
06	JPEG	4.667	14m 16s 848ms	1h 05m 54s 879ms
07	MPEG	3.580	10m 32s 329ms	60m 36s 377ms
08	PDF	2.181	06m 55s 451ms	59m 43s 401ms
09	JPEG	1.969	06m 15s 963ms	54m 17s 361ms
10	DOC	949	02m 58s 724ms	24m 14s 332ms
11	DOC	587	01m 51s 627ms	15m 50s 382ms
12	PPS	441	01m 24s 055ms	12m 45s 404ms
13	PDF	293	00m 57s 635ms	07m 54s 047ms
14	XLS	48	00m 09s 386ms	01m 14s 388ms
15	XLS	20	00m 03s 801ms	00m 30s 185ms
16	JPEG	16	00m 02s 822ms	00m 24s 022ms
17	TXT	8	00m 01s 403ms	00m 12s 032ms

Nas tabelas os tempos de cifragem, e decifragem são representados em hora(h), minuto(m), segundo(s) e miléssimo de segundo(ms). O número da consulta representa apenas o número do arquivo utilizado. Na extensão do arquivo estão listados todas as extensões utilizadas e seus tamanhos representados em Kb.

Para que se possa visualizar melhor a diferença de tempo entre os testes feitos com os dois algoritmos foram criadas duas tabelas, onde uma mostra a cifragem com os dois algoritmos e a outra mostra a decifragem. Essas diferenças de tempo são mostradas nas tabelas 6 e 7 a seguir:

Tabela 6. Comparativo de Cifragem dos Algoritmos DES e RSA

Nr. Consulta	***Extensão do Arquivo***	***Tamanho KB***	***Cifragem DES***	***Cifragem RSA***
01	MPEG	42.958	01m 31s 084ms	2h 13m 67s 511ms
02	MP3	19.332	00m 40s 042ms	57m 47s 908ms
03	EXE	15.365	00m 40s 749ms	47m 01s 807ms
04	PPT	7.559	00m 15s 933ms	23m 43s 177ms
05	MP3	5.355	00m 10s 999ms	16m 19s 083ms
06	JPEG	4.667	00m 09s 644ms	14m 16s 848ms
07	MPEG	3.580	00m 07s 444ms	10m 32s 329ms
08	PDF	2.181	00m 04s 525ms	06m 55s 451ms
09	JPEG	1.969	00m 04s 064ms	06m 15s 963ms
10	DOC	949	00m 01s 966ms	02m 58s 724ms
11	DOC	587	00m 01s 273ms	01m 51s 627ms
12	PPS	441	00m 00s 925ms	01m 24s 055ms
13	PDF	293	00m 00s 583ms	00m 57s 635ms
14	XLS	48	00m 00s 100ms	00m 09s 386ms
15	XLS	20	00m 00s 041ms	00m 03s 801ms
16	JPEG	16	00m 00s 034ms	00m 02s 822ms
17	TXT	8	00m 00s 017ms	00m 01s 403ms

Tabela 7. Comparativo de Decifragem dos Algoritmos DES e RSA

Nr. Consulta	*Extensão do Arquivo*	*Tamanho KB*	*Decifragem DES*	*Decifragem RSA*
01	MPEG	42.958	01m 27s 114ms	1h 22m 55s 557ms
02	MP3	19.332	00m 39s 993ms	1h 15m 43s 686ms
03	EXE	15.365	00m 40s 238ms	1h 27m 55s 624ms
04	PPT	7.559	00m 15s 718ms	1h 18m 49s 498ms
05	MP3	5.355	00m 11s 479ms	1h 13m 20s 829ms
06	JPEG	4.667	00m 09s 926ms	1h 05m 54s 879ms
07	MPEG	3.580	00m 07s 845ms	60m 36s 377ms
08	PDF	2.181	00m 04s 541ms	59m 43s 401ms
09	JPEG	1.969	00m 03s 973ms	54m 17s 361ms
10	DOC	949	00m 01s 975ms	24m 14s 332ms
11	DOC	587	00m 01s 297ms	15m 50s 382ms
12	PPS	441	00m 00s 961ms	12m 45s 404ms
13	PDF	293	00m 00s 588ms	07m 54s 047ms
14	XLS	48	00m 00s 098ms	01m 14s 388ms
15	XLS	20	00m 00s 047ms	00m 30s 185ms
16	JPEG	16	00m 00s 046ms	00m 24s 022ms
17	TXT	8	00m 00s 025ms	00m 12s 032ms

Nos testes realizados com o algoritmo DES, pode-se observar que não existe uma diferença significante nos tempos de cifragem e decifragem, sendo que o tempo gasto em cada operação é retornado pelo programa após a conclusão das mesmas.

Já nos testes feitos com o algoritmo RSA, observa-se que a diferença de tempo entre a cifragem e a decifragem é muito grande, chegando muitas vezes a travar o programa, e tendo que se reiniciar a operação novamente até que o programa consiga retornar o tempo exato para a realização de cada operação.

Também foram feitos testes com os dois algoritmos, com tamanhos de chaves diferentes para que fosse possível notar se o tamanho da chave iria influenciar na diferença de tempo. Com o algoritmo DES utilizando-se chaves de até 100 caracteres, sendo eles números e letras, notou-se que, os tempos foram praticamente iguais, com diferenças de no máximo 5 milisegundos. Com o RSA utilizando-se chaves de tamanho 128, 256, 512, 768 e 1024 bits, a diferença máxima foi de 25 milisegundos. Devido a

isso os quadros que serviram para cálculo da média foram os 10 que continham mais diferenças, e que podem ser visualizados nos anexos, como também pode-se visualizar alguns quadros com tamanhos de chaves diferentes.

Pode-se verificar nas tabelas 6 e 7 com a diferença dos tempos de cada processo realizado, que o algoritmo DES chegou a ser muito mais rápido que o algoritmo RSA, mostrando-se claramente mais eficiente quanto a agilidade.

Isso deixa claro que o algoritmo DES além de ser mais fácil de ser utilizado, pois necessita apenas de uma chave tanto para criptografar quanto para descriptografar um arquivo, também é mais rápido em relação ao algoritmo RSA, devendo ser utilizado com mais freqüência por aplicações que necessitam de rapidez.

Mas como os arquivos criptografados pelos dois algoritmos não permitiram que fossem acessados seus conteúdos sem as chaves criadas, não pode-se atribuir a afirmação de que o algoritmo DES seja mais eficiente quanto à segurança em relação ao algoritmo RSA, e vice-versa. Com relação ao impacto dos arquivos criptografados na rede, verificou-se não haver uma necessidade de realizar testes, pois os arquivos de saída ficam com tamanhos iguais aos arquivos originais, fazendo com que o tempo que esses arquivos utilizam para serem enviados de um computador ao outro, é o mesmo tempo dos arquivos originais.

Com os testes realizados pode-se confirmar o que os autores, Moreno, Pereira e Chiaramonte ref ao livro “Criptografia em Software e Hardware”, já afirmavam nas páginas 127 e 155. Os mesmos afirmavam que o algoritmo DES é bastante explorado, tendo sua implementação atingindo um bom nível de desempenho, e o algoritmo RSA é o mais lento de todos os algoritmos que os autores implementaram. Também afirmam que os algoritmos simétricos são mais rápidos em realizar o processo de criptografar e descriptografar do que os algoritmos assimétricos.

CONCLUSÃO

Tanto os algoritmos assimétricos, quanto os algoritmos simétricos estão justamente no mercado para auxiliarem as organizações e as pessoas a protegerem seus arquivos comerciais e pessoais, de pessoas mal intencionadas. A criptografia veio para ajudar nessa proteção, que hoje em dia é muito procurada por todos.

Como objeto de estudo dessa pesquisa foi realizado a comparação do desempenho do algoritmo simétrico DES *versus* o algoritmo assimétrico RSA, e como pode-se observar nos resultados obtidos, o algoritmo DES é muito mais rápido do que o algoritmo RSA, sendo que o primeiro utiliza apenas uma chave e o segundo utiliza duas chaves, pode-se concluir que as chaves e os seus tamanhos não influenciam significantemente no desempenham dos algoritmos, e sim na sua segurança.

Então conclui-se que mesmo com a escolha de um algoritmo como padrão de criptografia, os demais também tem sua importância, e todos eles apresentam falhas que precisam ser cada dia mais estudadas para que os algoritmos não fiquem a mercê de pessoas mal intencionadas.

Conclui-se então que, sempre que um sistema precisar de um método de segurança de informações, e esse método for a criptografia, é necessário fazer um estudo sobre qual algoritmo utilizar, se será um algoritmo simétrico ou assimétrico, para saber qual deles irá satisfazer as necessidades do sistema.

Como teve-se a oportunidade de analisar as diversas características da criptografia, e até mesmo dos algoritmos utilizados, pode-se perceber que por a criptografia ser um fator importante para a segurança das informações, ainda tem-se uma série de testes que podem ser realizados para contribuir com as conclusões chegadas nesse trabalho. Como sugestão de trabalhos futuros tem-se:

a) realizar testes em rede com os dois algoritmos, enviando dados em formato de imagem, vídeo, e tamanhos grandes, para que se consiga medir o desempenho com arquivos mais complexos;
b) comparar com outros tipos de algoritmos, tanto simétricos como assimétricos, para que se possa verificar se o DES e o RSA são realmente os mais eficazes.
c) realizar um estudo crescente sobre criptografia, pois é cada vez maior o número de pessoas e organizações que utilizam essa técnica, incluindo estudos sobre novos algoritmos de criptografia, pois com o passar dos anos, os algoritmos apresentados nessa pesquisa vão tornando-se vulneráveis a ataques de quebra de segurança.
d) realizar a mesma comparação utilizando-se outros sistemas operacionais, como o Linux por exemplo, e também outros tipos de softwares.

REFERÊNCIAS

BASTOS, Eduardo N. F. Introdução a Criptografia, Artigo desenvolvido na Universidade Católica de Pelotas-RS

BURNETT, S; PAINE S. **Criptografia e Segurança.** O guia oficial RSA. Rio de Janeiro: Campus, 2002.

CARVALHO, Daniel Balparda de**. Segurança de Dados com Criptografia**. Rio de Janeiro: Book Express, 2001.

DELIBERADOR, Paulo de Tarso. **Um Componente Computacional para Auxiliar o Desempenho de uma Assinatura Digital no Sistema de Informações Processuais.** 2004. 186 f. Dissertação (Pós-Graduação em Engenharia de Produção) – Universidade federal de Santa Catarina, Florianópolis - SC.

FARIA, Fabiano Otávio. **Estudo da Técnica de Criptografia Algoritmo Posicional – Alpos na Segurança dos Dados de um Banco de Dados**. 2006. 56 f. Monografia (Curso de Sistema de Informação) - Faculdades Santo Agostinho – FASA e Faculdade de Ciências Exatas e Tecnológicas – FACET, Montes Claros – MG.

FUZITAKI, Gerson Yoshio. **Avaliação do Desempenho de Algoritmos de Criptografia.** 2004. 85 f. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Centro de Ciências Exatas, Universidade Estadual de Londrina, Londrina – PR.

LYNCH, Daniel C., LUNDQUIST, Leslie. **Dinheiro Digital: o comércio na Internet.** Tradução por: Follow-up Traduções e Assessoria de Informática. Rio de Janeiro, Campus, 1996.

MARGI, Cíntia Borges. **Um Mecanismo para Distribuição Segura de Vídeo MPEG.** 2000. 127 f. Dissertação (Escola Politécnica) – Universidade de São Paulo, São Paulo.

MORAES, Rosane França. **Construção de um ambiente WEB com ferramentas para estudo de algoritmos de criptografia por meio do Matlab**. Rio de Janeiro, 2004.
Departamento de Eletrônica – Escola de Engenharia. UFRJ – Universidade Federal do Rio de Janeiro.

MORENO, Edward David; PEREIRA, Fábio Dacêncio; CHIARAMONTE, Rodolfo Barros. **Criptografia em Software e Hardware**. São Paulo: Novatec, 2005**.**

PICONI, Andressa Cristiani; SUZUKI, Elcio Keniti; CAMARGO, Juliana Aparecida. **Algoritmo RSA**. Campinas, 2004. Especialização em Redes de Computadores – Instituto de Computação, UNICAMP – Universidade de Campinas.

RIGHETTI, Fabiano Reese. **O Impacto do Uso da Criptografia no Throughput de Redes Locais:** um estudo de caso usando o algoritmo triple-des. Monografia. 2004. 49 f. (Especialização em Ciências da Computação) Faculdade de Ciências Aplicadas de Cascavel – FACIAP.

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

TANENBAUM, A. S. **Redes de Computadores**. Rio de Janeiro: Campus, 4ª ed., 1997.

TERADA, Routo. **Segurança de Dados:** Criptografia em Redes de Computador. São Paulo: Edgard Blucher, 2000.

VOLPI, Marlon M. **Assinatura Digital:** Aspectos Técnicos, Práticos e Legais. Rio de Janeiro: Axcel Books do Brasil, 2001.

SOUZA, FLÁVIO DE. Jr Cripto e Jr Cripto Des: softwares de criptografia. Criciúma, 2008

BIBLIOGRAFIA COMPLEMENTAR

BARRETO, Paulo. **Escola Politécnica da Universidade de São Paulo,** 2003. Disponível em: <http://planeta.terra.com.br/informatica/paulobarreto/> Acesso em: 15 Mai. 2007.

BERNSTEIN, Terry, BRIMANI, Ansh B. SHULTZ, Eugene, SIEGEL, Carol A. **Segurança na Internet.** Rio de Janeiro: Campus, 1997.

CHIARAMONTE, Rodolfo Barros; MORENO, Edward David. **Criptografia Posicional em Hardware (VHDL e FPGAs)**. São Paulo: Revista REIC-SBC, 2002**.**

COUTINHO, S,C. **Números Inteiros e Criptografia RSA**. Rio de Janeiro: IMPA/SBM, 2000.

DINIZ, Davi Monteiro. Monteiro. **Documentos Eletrônicos, Assinaturas Digitais:** da qualificação jurídica dos arquivos digitais como documentos**.** São Paulo: LTr, 1999.

GARFINKEL, Simson; SPAFFORD, Gene. **Comércio e Segurança na Web.** São Paulo: Market Press, 1999.

LIMA, Marcelo Ferreira. **Assinatura Digital**: Solução DELPHI & CAPICOM. Florianópolis: Visual Books, 2005.

MACHADO, Giancarlo Bianchin. **Implementação de Assinatura Digital na Evolução Médica de Prontuário Eletrônico do Paciente:** Um Estudo de Caso no Hospital Regional de Araranguá. 2006. 71 f. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade do Extremo Sul Catarinense, Criciúma.

MORENO, Edward David. **Projeto, Desempenho e Aplicações de Sistemas Digitais em Circuitos Programáveis (FPGAS)**. Marília: Bless, 2003**.**

REZENDE, P. A. Dourado. **Departamento da Ciência da Computação da Universidade de Brasília**. 2002. Disponível em: <http://www.cic.unb.br/docentes/pedro/trabs/leis.htm> Acesso em: 20 de Mai. 2007.

SCHNEIER, Bruce. **Segurança.com –** Segredos e mentiras sobre a proteção na vida digital. Rio de Janeiro: Campus, 2001.

TRINTA, F. A. Mota; MACEDO, Rodrigo Cavalcanti. **Um Estudo Sobre Criptografia e Assinatura Digital,** 1998. Disponível em: <http://www.di.ufpe.br/~flash/ais98/cripto/criptografia.htm>. Acesso em 15 de Mai 2007.

UNO, D. N.; FALEIROS, A. C. **Princípios de Criptografia Quântica**. [S.l.], 2003. Disponível em: <http://www.bibl.ita.br/ixencita/artigos/FundDanielNobuo.pdf>. Acesso em 18 de Mai. 2007.

WADLOW, Thomas A.**. Segurança de Redes: Projeto e Gerenciamento de Redes Seguras**. Rio de Janeiro: Campus, 2001.

WEBER, Raul Fernando. **Criptografia Contemporânea**. Porto Alegre: Instituto de informática. Ufrgs. Disponível em: <http://www.inf.ufsc.br/~mauro/curso/cripto.doc>. Acesso em: 04 Abr. 2007.

ANEXOS

Em anexo encontram-se as tabelas com as tomadas de tempo de cifragem e decifragem utilizados para a realização da pesquisa. A nomenclatura dos testes, por exemplo: Teste 1, Teste 2 e assim por diante, significa o número do teste realizado com o algoritmo referente.

Teste 1 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1m 18s 594ms	1m 16s 125ms
02	MP3	19.332	00m 35s 328ms	00m 33s 968ms
03	EXE	15.365	00m 45s 953ms	00m 48s 828ms
04	PPT	7.559	00m 13s 703ms	00m 13s 641ms
05	MP3	5.355	00m 09s 859ms	00m 10s 656ms
06	JPEG	4.667	00m 09s 156ms	00m 08s 828ms
07	MPEG	3.580	00m 06s 375ms	00m 06s 672ms
08	PDF	2.181	00m 03s 937ms	00m 04s 250ms
09	JPEG	1.969	00m 03s 703ms	00m 03s 453ms
10	DOC	949	00m 01s 641ms	00m 02s 016ms
11	DOC	587	00m 01s 172ms	00m 01s 062ms
12	PPS	441	00m 00s 828ms	00m 00s 860ms
13	PDF	293	00m 00s 532ms	00m 00s 547ms
14	XLS	48	00m 00s 078ms	00m 00s 093ms
15	XLS	20	00m 00s 032ms	00m 00s 031ms
16	JPEG	16	00m 00s 015ms	00m 00s 125ms
17	TXT	8	00m 00s 016ms	00m 00s 047ms

Quadro 17. Teste 1 Realizado com o algoritmo DES

Teste 2 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 58s 219ms	01m 32s 046ms
02	MP3	19.332	00m 38s 687ms	00m 44s 125ms
03	EXE	15.365	00m 38s 000ms	00m 42s 937ms
04	PPT	7.559	00m 15s 812ms	00m 17s 765ms
05	MP3	5.355	00m 11s 016ms	00m 14s 937ms
06	JPEG	4.667	00m 09s 485ms	00m 10s 469ms
07	MPEG	3.580	00m 07s 328ms	00m 08s 500ms
08	PDF	2.181	00m 04s 437ms	00m 04s 984ms
09	JPEG	1.969	00m 04s 015ms	00m 04s 407ms
10	DOC	949	00m 01s 937ms	00m 02s 141ms
11	DOC	587	00m 01s 203ms	00m 01s 422ms
12	PPS	441	00m 00s 906ms	00m 01s 000ms
13	PDF	293	00m 00s 578ms	00m 00s 578ms
14	XLS	48	00m 00s 110ms	00m 00s 093ms
15	XLS	20	00m 00s 047ms	00m 00s 047ms
16	JPEG	16	00m 00s 032ms	00m 00s 032ms
17	TXT	8	00m 00s 016ms	00m 00s 015ms

Quadro 18. Teste 2 Realizado com o algoritmo DES

Teste 3 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 31s 297ms	01m 29s 344ms
02	MP3	19.332	00m 40s 907ms	00m 40s 953ms
03	EXE	15.365	00m 40s 672ms	00m 40s 953ms
04	PPT	7.559	00m 16s 828ms	00m 16s 625ms
05	MP3	5.355	00m 11s 391ms	00m 11s 766ms
06	JPEG	4.667	00m 09s 969ms	00m 10s 438ms
07	MPEG	3.580	00m 07s 828ms	00m 07s 875ms
08	PDF	2.181	00m 04s 547ms	00m 04s 766ms
09	JPEG	1.969	00m 04s 188ms	00m 04s 344ms
10	DOC	949	00m 02s 079ms	00m 02s 141ms
11	DOC	587	00m 01s 375ms	00m 01s 328ms
12	PPS	441	00m 00s 875ms	00m 01s 016ms
13	PDF	293	00m 00s 656ms	00m 00s 594ms
14	XLS	48	00m 00s 094ms	00m 00s 078ms
15	XLS	20	00m 00s 046ms	00m 00s 047ms
16	JPEG	16	00m 00s 032ms	00m 00s 032ms
17	TXT	8	00m 00s 016ms	00m 00s 047ms

Quadro 19. Teste 3 Realizado com o algoritmo DES

Teste 4 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 33s984ms	01m 26s 531ms
02	MP3	19.332	00m 41s 766ms	00m 38s 407ms
03	EXE	15.365	00m 41s 719ms	00m 38s 282ms
04	PPT	7.559	00m 16s 875ms	00m 15s 109ms
05	MP3	5.355	00m 11s 704ms	00m 10s 750ms
06	JPEG	4.667	00m 10s 578ms	00m 09s 719ms
07	MPEG	3.580	00m 07s 625ms	00m 07s 812ms
08	PDF	2.181	00m 04s 657ms	00m 04s 625ms
09	JPEG	1.969	00m 04s 187ms	00m 04s 062ms
10	DOC	949	00m 02s 032ms	00m 01s 953ms
11	DOC	587	00m 01s 266ms	00m 01s 235ms
12	PPS	441	00m 01s 032ms	00m 01s 062ms
13	PDF	293	00m 00s 563ms	00m 00s 578ms
14	XLS	48	00m 00s 109ms	00m 00s 093ms
15	XLS	20	00m 00s 047ms	00m 00s 062ms
16	JPEG	16	00m 00s 031ms	00m 00s 047ms
17	TXT	8	00m 00s 015ms	00m 00s 031ms

Quadro 20. Teste 4 Realizado com o algoritmo DES

Teste 5 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 30s 438ms	01m 37s 578ms
02	MP3	19.332	00m 40s 531ms	00m 41s 688ms
03	EXE	15.365	00m 42s 484ms	00m 41s 344ms
04	PPT	7.559	00m 17s 609ms	00m 16s 922ms
05	MP3	5.355	00m 11s 359ms	00m 11s 922ms
06	JPEG	4.667	00m 09s 766ms	00m 13s 046ms
07	MPEG	3.580	00m 07s 594ms	00m 11s 719ms
08	PDF	2.181	00m 04s 765ms	00m 04s 672ms
09	JPEG	1.969	00m 04s 172ms	00m 04s 063ms
10	DOC	949	00m 02s 000ms	00m 01s 906ms
11	DOC	587	00m 01s 296ms	00m 01s 250ms
12	PPS	441	00m 00s 953ms	00m 00s 984ms
13	PDF	293	00m 00s 594ms	00m 00s 563ms
14	XLS	48	00m 00s 093ms	00m 00s 109ms
15	XLS	20	00m 00s 032ms	00m 00s 047ms
16	JPEG	16	00m 00s 047ms	00m 00s 046ms
17	TXT	8	00m 00s 015ms	00m 00s 016ms

Quadro 21. Teste 5 Realizado com o algoritmo DES

Teste 6 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 36s 375ms	01m 28s 578ms
02	MP3	19.332	00m 41s 952ms	00m 39s 812ms
03	EXE	15.365	00m 42s 985ms	00m 38s 719ms
04	PPT	7.559	00m 17s 078ms	00m 15s 516ms
05	MP3	5.355	00m 11s 813ms	00m 10s 844ms
06	JPEG	4.667	00m 09s 953ms	00m 09s 281ms
07	MPEG	3.580	00m 07s 890ms	00m 07s 204ms
08	PDF	2.181	00m 04s 734ms	00m 04s 453ms
09	JPEG	1.969	00m 04s 266ms	00m 03s 922ms
10	DOC	949	00m 02s 078ms	00m 01s 969ms
11	DOC	587	00m 01s 344ms	00m 01s 156ms
12	PPS	441	00m 01s 000ms	00m 00s 953ms
13	PDF	293	00m 00s 562ms	00m 00s 578ms
14	XLS	48	00m 00s 110ms	00m 00s 094ms
15	XLS	20	00m 00s 047ms	00m 00s 031ms
16	JPEG	16	00m 00s 047ms	00m 00s 031ms
17	TXT	8	00m 00s 031ms	00m 00s 031ms

Quadro 22. Teste 6 Realizado com o algoritmo DES

Teste 7 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 25s 859ms	01m 25s 672ms
02	MP3	19.332	00m 39s 047ms	00m 40s 250ms
03	EXE	15.365	00m 38s 969ms	00m 37s 859ms
04	PPT	7.559	00m 15s 047ms	00m 15s 422ms
05	MP3	5.355	00m 11s 094ms	00m 10s 484ms
06	JPEG	4.667	00m 09s 375ms	00m 09s 375ms
07	MPEG	3.580	00m 08s 203ms	00m 07s 375ms
08	PDF	2.181	00m 05s 000ms	00m 04s 422ms
09	JPEG	1.969	00m 04s 422ms	00m 03s 922ms
10	DOC	949	00m 02s 328ms	00m 01s 907ms
11	DOC	587	00m 01s 344ms	00m 01s 984ms
12	PPS	441	00m 00s 968ms	00m 00s 969ms
13	PDF	293	00m 00s 578ms	00m 00s 594ms
14	XLS	48	00m 00s 094 ms	00m 00s 110ms
15	XLS	20	00m 00s 047ms	00m 00s 047ms
16	JPEG	16	00m 00s 031ms	00m 00s 047ms
17	TXT	8	00m 00s 016ms	00m 00s 015ms

Quadro 23. Teste 7 Realizado com o algoritmo DES

Teste 8 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 26s 563ms	01m 26s 390ms
02	MP3	19.332	00m 41s 515ms	00m 40s 252ms
03	EXE	15.365	00m 39s 344ms	00m 37s 859ms
04	PPT	7.559	00m 15s 313ms	00m 15s 444ms
05	MP3	5.355	00m 10s 765ms	00m 11s 047ms
06	JPEG	4.667	00m 09s 484ms	00m 09s 375ms
07	MPEG	3.580	00m 07s 453ms	00m 07s 093ms
08	PDF	2.181	00m 04s 406ms	00m 04s 456ms
09	JPEG	1.969	00m 03s 953ms	00m 03s 922ms
10	DOC	949	00m 01s 875ms	00m 01s 917ms
11	DOC	587	00m 01s 234ms	00m 01s 250ms
12	PPS	441	00m 00s 937ms	00m 00s 953ms
13	PDF	293	00m 00s 562ms	00m 00s 609ms
14	XLS	48	00m 00s 109 ms	00m 00s 094ms
15	XLS	20	00m 00s 047ms	00m 00s 063ms
16	JPEG	16	00m 00s 031ms	00m 00s 031ms
17	TXT	8	00m 00s 015ms	00m 00s 015ms

Quadro 24. Teste 8 Realizado com o algoritmo DES

Teste 9 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 24s 734ms	01m 24s 469ms
02	MP3	19.332	00m 40s 344ms	00m 40s 240ms
03	EXE	15.365	00m 38s 157ms	00m 37s 850ms
04	PPT	7.559	00m 14s 812ms	00m 15s 308ms
05	MP3	5.355	00m 10s 515ms	00m 11s 813ms
06	JPEG	4.667	00m 09s 250ms	00m 09s 376ms
07	MPEG	3.580	00m 07s 140ms	00m 07s 125ms
08	PDF	2.181	00m 04s 406ms	00m 04s 399ms
09	JPEG	1.969	00m 03s 860ms	00m 03s 800ms
10	DOC	949	00m 01s 843ms	00m 01s 890ms
11	DOC	587	00m 01s 297ms	00m 01s 141ms
12	PPS	441	00m 00s 875ms	00m 00s 938ms
13	PDF	293	00m 00s 641ms	00m 00s 657ms
14	XLS	48	00m 00s 110 ms	00m 00s 094ms
15	XLS	20	00m 00s 032ms	00m 00s 047ms
16	JPEG	16	00m 00s 031ms	00m 00s 032ms
17	TXT	8	00m 00s 015ms	00m 00s 016ms

Quadro 25. Teste 9 Realizado com o algoritmo DES

Teste 10 - Realizado com o Algoritmo Des				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 24s 781ms	01m 24s 406ms
02	MP3	19.332	00m 40s 343ms	00m 40s 231ms
03	EXE	15.365	00m 39s 203ms	00m 37s 750ms
04	PPT	7.559	00m 16s 250ms	00m 15s 423ms
05	MP3	5.355	00m 10s 469ms	00m 10s 570ms
06	JPEG	4.667	00m 09s 421ms	00m 09s 356ms
07	MPEG	3.580	00m 07s 000ms	00m 07s 078ms
08	PDF	2.181	00m 04s 360ms	00m 04s 378ms
09	JPEG	1.969	00m 03s 875ms	00m 03s 831ms
10	DOC	949	00m 01s 844ms	00m 01s 910ms
11	DOC	587	00m 01s 203ms	00m 01s 141ms
12	PPS	441	00m 00s 875ms	00m 00s 875ms
13	PDF	293	00m 00s 563ms	00m 00s 578ms
14	XLS	48	00m 00s 094 ms	00m 00s 125ms
15	XLS	20	00m 00s 031ms	00m 00s 047ms
16	JPEG	16	00m 00s 047ms	00m 00s 032ms
17	TXT	8	00m 00s 016ms	00m 00s 016ms

Quadro 26. Teste 10 Realizado com o algoritmo DES

Teste 1 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 594ms	1h 22m 45s 209ms
02	MP3	19.332	57m 48s 375ms	1h 15m 30s 469ms
03	EXE	15.365	47m 01s 266ms	87m 57s 654ms
04	PPT	7.559	23m 01s 859ms	78m 59s 769ms
05	MP3	5.355	16m 18s 953ms	73m 20s 469ms
06	JPEG	4.667	14m 16s 953ms	65m 56s 987ms
07	MPEG	3.580	10m 32s 047ms	60m 30s 267ms
08	PDF	2.181	06m 55s 203ms	59m 47s 469ms
09	JPEG	1.969	06m 16s 235ms	54m 13s 265ms
10	DOC	949	02m 58s 860ms	24m 13s 250ms
11	DOC	587	01m 51s 250ms	15m 50s 078ms
12	PPS	441	01m 23s 969ms	12m 46s 328ms
13	PDF	293	00m 57s 578ms	07m 53s 860ms
14	XLS	48	00m 09s 391ms	01m 14s 593ms
15	XLS	20	00m 03s 703ms	00m 30s 078ms
16	JPEG	16	00m 02s 969ms	00m 23s 719ms
17	TXT	8	00m 01s 390ms	00m 11s 890ms

Quadro 27. Teste 1 Realizado com o algoritmo RSA

Teste 2 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 643ms	1h 22m 56s 300ms
02	MP3	19.332	57m 48s 300ms	1h 15m 30s 513ms
03	EXE	15.365	47m 01s 379ms	87m 58s 245ms
04	PPT	7.559	23m 01s 678ms	78m 56s 098ms
05	MP3	5.355	16m 18s 234ms	73m 20s 786ms
06	JPEG	4.667	14m 16s 978ms	65m 46s 398ms
07	MPEG	3.580	10m 32s 235ms	60m 31s 387ms
08	PDF	2.181	06m 55s 267ms	59m 47s 570ms
09	JPEG	1.969	06m 16s 245ms	54m 14s 200ms
10	DOC	949	02m 58s 867ms	24m 13s 100ms
11	DOC	587	01m 51s 247ms	15m 50s 918ms
12	PPS	441	01m 23s 960ms	12m 45s 789ms
13	PDF	293	00m 57s 579ms	07m 53s 860ms
14	XLS	48	00m 09s 387ms	01m 14s 593ms
15	XLS	20	00m 03s 763ms	00m 30s 079ms
16	JPEG	16	00m 02s 965ms	00m 23s 710ms
17	TXT	8	00m 01s 392ms	00m 11s 892ms

Quadro 28. Teste 2 Realizado com o algoritmo RSA

Teste 3 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 11m 41s 100ms	1h 23m 45s 216ms
02	MP3	19.332	57m 49s 659ms	1h 16m 31s 470ms
03	EXE	15.365	47m 05s 200ms	87m 56s 655ms
04	PPT	7.559	23m 03s 274ms	78m 60s 777ms
05	MP3	5.355	16m 15s 856ms	73m 21s 450ms
06	JPEG	4.667	14m 16s 750ms	65m 56s 999ms
07	MPEG	3.580	10m 33s 567ms	60m 30s 289ms
08	PDF	2.181	06m 56s 203ms	59m 46s 678ms
09	JPEG	1.969	06m 16s 602ms	54m 13s 345ms
10	DOC	949	02m 59s 871ms	24m 14s 251ms
11	DOC	587	01m 51s 938ms	15m 51s 079ms
12	PPS	441	01m 22s 754ms	12m 47s 309ms
13	PDF	293	00m 58s 234ms	07m 54s 799ms
14	XLS	48	00m 09s 398ms	01m 13s 576ms
15	XLS	20	00m 04s 567ms	00m 30s 079ms
16	JPEG	16	00m 01s 969ms	00m 24s 719ms
17	TXT	8	00m 01s 587ms	00m 12s 356ms

Quadro 29. Teste 3 Realizado com o algoritmo RSA

Teste 4 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 654ms	1h 22m 45s 345ms
02	MP3	19.332	57m 48s 397ms	1h 15m 30s 768ms
03	EXE	15.365	47m 01s 277ms	87m 57s 644ms
04	PPT	7.559	23m 01s 999ms	78m 59s 770ms
05	MP3	5.355	16m 18s 945ms	73m 20s 312ms
06	JPEG	4.667	14m 16s 958ms	65m 56s 989ms
07	MPEG	3.580	10m 32s 049ms	60m 30s 234ms
08	PDF	2.181	06m 55s 213ms	59m 47s 478ms
09	JPEG	1.969	06m 16s 240ms	54m 13s 245ms
10	DOC	949	02m 58s 889ms	24m 13s 248ms
11	DOC	587	01m 51s 256ms	15m 50s 120ms
12	PPS	441	01m 23s 999ms	12m 46s 456ms
13	PDF	293	00m 57s 588ms	07m 53s 889ms
14	XLS	48	00m 09s 395ms	01m 14s 488ms
15	XLS	20	00m 03s 701ms	00m 30s 090ms
16	JPEG	16	00m 02s 970ms	00m 23s 725ms
17	TXT	8	00m 01s 392ms	00m 11s 889ms

Quadro 30. Teste 4 Realizado com o algoritmo RSA

Teste 5 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 678ms	1h 22m 45s 567ms
02	MP3	19.332	57m 48s 389ms	1h 15m 30s 789ms
03	EXE	15.365	47m 01s 378ms	87m 57s 823ms
04	PPT	7.559	23m 01s 799ms	78m 59s 700ms
05	MP3	5.355	16m 13s 978ms	73m 20s 469ms
06	JPEG	4.667	14m 16s 958ms	65m 56s 888ms
07	MPEG	3.580	10m 32s 039ms	60m 30s 300ms
08	PDF	2.181	06m 55s 450ms	59m 47s 460ms
09	JPEG	1.969	06m 16s 345ms	54m 13s 278ms
10	DOC	949	02m 58s 868ms	24m 13s 199ms
11	DOC	587	01m 51s 247ms	15m 50s 079ms
12	PPS	441	01m 23s 978ms	12m 46s 368ms
13	PDF	293	00m 57s 534ms	07m 53s 839ms
14	XLS	48	00m 09s 410ms	01m 14s 567ms
15	XLS	20	00m 03s 699ms	00m 30s 099ms
16	JPEG	16	00m 02s 978ms	00m 23s 756ms
17	TXT	8	00m 01s 401ms	00m 11s 901ms

Quadro 31. Teste 5 Realizado com o algoritmo RSA

Teste 6 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 601ms	1h 22m 46s 104ms
02	MP3	19.332	57m 48s 399ms	1h 15m 30s 678ms
03	EXE	15.365	47m 01s 301ms	87m 57s 600ms
04	PPT	7.559	24m 01s 567ms	78m 59s 801ms
05	MP3	5.355	16m 19s 001ms	73m 20s 9 00ms
06	JPEG	4.667	14m 17s 023ms	65m 56s 670ms
07	MPEG	3.580	10m 32s 076ms	60m 30s 234ms
08	PDF	2.181	06m 55s 240ms	59m 47s 489ms
09	JPEG	1.969	06m 15s 999ms	54m 13s 235ms
10	DOC	949	02m 58s 398ms	24m 13s 225ms
11	DOC	587	01m 51s 267ms	15m 50s 124ms
12	PPS	441	01m 23s 000ms	12m 46s 401ms
13	PDF	293	00m 57s 589ms	07m 53s 780ms
14	XLS	48	00m 09s 401ms	01m 14s 578ms
15	XLS	20	00m 03s 734ms	00m 30s 090ms
16	JPEG	16	00m 02s 975ms	00m 23s 756ms
17	TXT	8	00m 01s 395ms	00m 11s 826ms

Quadro 32. Teste 6 Realizado com o algoritmo RSA

Teste 7 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 25m 20s 014ms	1h 22m 45s 210ms
02	MP3	19.332	57m 41s 234ms	1h 15m 30s 470ms
03	EXE	15.365	47m 01s 378ms	87m 57s 655ms
04	PPT	7.559	24m 06s 989ms	78m 59s 770ms
05	MP3	5.355	16m 18s 999ms	73m 20s 489ms
06	JPEG	4.667	14m 17s 001ms	65m 56s 902ms
07	MPEG	3.580	10m 32s 134ms	60m 30s 254ms
08	PDF	2.181	06m 55s 309ms	59m 47s 473ms
09	JPEG	1.969	06m 16s 267ms	54m 13s 254ms
10	DOC	949	02m 58s 901ms	24m 13s 290ms
11	DOC	587	01m 51s 301ms	15m 50s 189ms
12	PPS	441	01m 23s 989ms	12m 46s 403ms
13	PDF	293	00m 57s 610ms	07m 53s 859ms
14	XLS	48	00m 09s 403ms	01m 14s 605ms
15	XLS	20	00m 03s 825ms	00m 30s 111ms
16	JPEG	16	00m 02s 989ms	00m 23s 700ms
17	TXT	8	00m 01s 390ms	00m 11s 887ms

Quadro 33. Teste 7 Realizado com o algoritmo RSA

Teste 8 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 593ms	1h 22m 45s 208ms
02	MP3	19.332	57m 48s 374ms	1h 15m 30s 468ms
03	EXE	15.365	47m 01s 267ms	87m 57s 655ms
04	PPT	7.559	23m 01s 858ms	78m 59s 768ms
05	MP3	5.355	16m 18s 955ms	73m 20s 468ms
06	JPEG	4.667	14m 16s 954ms	65m 56s 986ms
07	MPEG	3.580	10m 32s 046ms	60m 30s 268ms
08	PDF	2.181	06m 55s 206ms	59m 47s 467ms
09	JPEG	1.969	06m 16s 234ms	54m 13s 263ms
10	DOC	949	02m 58s 861ms	24m 13s 254ms
11	DOC	587	01m 51s 253ms	15m 50s 076ms
12	PPS	441	01m 23s 967ms	12m 46s 326ms
13	PDF	293	00m 57s 579ms	07m 53s 864ms
14	XLS	48	00m 09s 392ms	01m 14s 591ms
15	XLS	20	00m 03s 704ms	00m 30s 070ms
16	JPEG	16	00m 02s 968ms	00m 23s 711ms
17	TXT	8	00m 01s 395ms	00m 11s 894ms

Quadro 34. Teste 8 Realizado com o algoritmo RSA

Teste 9 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 52s 593ms	1h 22m 35s 203ms
02	MP3	19.332	57m 49s 373ms	1h 15m 60s 466ms
03	EXE	15.365	47m 02s 261ms	87m 37s 657ms
04	PPT	7.559	23m 02s 854ms	77m 59s 761ms
05	MP3	5.355	16m 28s 955ms	73m 22s 462ms
06	JPEG	4.667	14m 15s 956ms	65m 46s 984ms
07	MPEG	3.580	10m 33s 048ms	60m 90s 265ms
08	PDF	2.181	06m 56s 202ms	59m 07s 460ms
09	JPEG	1.969	06m 13s 232ms	54m 53s 264ms
10	DOC	949	02m 57s 865ms	24m 23s 252ms
11	DOC	587	01m 54s 259ms	15m 51s 079ms
12	PPS	441	01m 26s 967ms	12m 36s 329ms
13	PDF	293	00m 57s 479ms	07m 54s 861ms
14	XLS	48	00m 09s 294ms	01m 13s 595ms
15	XLS	20	00m 03s 607ms	00m 31s 079ms
16	JPEG	16	00m 02s 468ms	00m 25s 710ms
17	TXT	8	00m 01s 297ms	00m 12s 895ms

Quadro 35. Teste 9 Realizado com o algoritmo RSA

Teste 10 - Realizado com o Algoritmo RSA				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 643ms	1h 22m 46s 209ms
02	MP3	19.332	57m 48s 575ms	1h 15m 30s 768ms
03	EXE	15.365	47m 01s 366ms	87m 57s 654ms
04	PPT	7.559	25m 07s 897ms	78m 59s 769ms
05	MP3	5.355	16m 18s 953ms	73m 20s 489ms
06	JPEG	4.667	14m 16s 953ms	65m 56s 987ms
07	MPEG	3.580	10m 32s 047ms	60m 30s 267ms
08	PDF	2.181	06m 55s 213ms	59m 47s 469ms
09	JPEG	1.969	06m 16s 235ms	54m 13s 265ms
10	DOC	949	02m 57s 860ms	24m 13s 250ms
11	DOC	587	01m 51s 250ms	15m 50s 076ms
12	PPS	441	01m 23s 969ms	12m 46s 328ms
13	PDF	293	00m 57s 578ms	07m 53s 860ms
14	XLS	48	00m 09s 391ms	01m 14s 693ms
15	XLS	20	00m 03s 705ms	00m 30s 079ms
16	JPEG	16	00m 02s 967ms	00m 23s 718ms
17	TXT	8	00m 01s 392ms	00m 11s 894ms

Quadro 36. Teste 10 Realizado com o algoritmo RSA

Teste 11 - Realizado com o Algoritmo RSA com tamanho de chave 256				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 670ms	1h 22m 46s 230ms
02	MP3	19.332	57m 48s 590ms	1h 15m 30s 775ms
03	EXE	15.365	47m 01s 359ms	87m 57s 665ms
04	PPT	7.559	25m 07s 892ms	78m 59s 770ms
05	MP3	5.355	16m 18s 970ms	73m 20s 491ms
06	JPEG	4.667	14m 16s 963ms	65m 56s 989ms
07	MPEG	3.580	10m 32s 040ms	60m 30s 274ms
08	PDF	2.181	06m 55s 230ms	59m 47s 475ms
09	JPEG	1.969	06m 16s 210ms	54m 13s 269ms
10	DOC	949	02m 57s 865ms	24m 13s 275ms
11	DOC	587	01m 51s 275ms	15m 50s 075ms
12	PPS	441	01m 23s 916ms	12m 46s 329ms
13	PDF	293	00m 57s 590ms	07m 53s 863ms
14	XLS	48	00m 09s 392ms	01m 14s 696ms
15	XLS	20	00m 03s 707ms	00m 30s 078ms
16	JPEG	16	00m 02s 966ms	00m 23s 714ms
17	TXT	8	00m 01s 391ms	00m 11s 896ms

Quadro 37. Teste 11 Realizado com o algoritmo RSA com chave de 256

Teste 12 - Realizado com o Algoritmo RSA com tamanho de chave 768				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 640ms	1h 22m 46s 235ms
02	MP3	19.332	57m 48s 571ms	1h 15m 30s 774ms
03	EXE	15.365	47m 01s 366ms	87m 57s 659ms
04	PPT	7.559	25m 07s 895ms	78m 59s 765ms
05	MP3	5.355	16m 18s 954ms	73m 20s 495ms
06	JPEG	4.667	14m 16s 952ms	65m 56s 990ms
07	MPEG	3.580	10m 32s 049ms	60m 30s 282ms
08	PDF	2.181	06m 55s 211ms	59m 47s 470ms
09	JPEG	1.969	06m 16s 234ms	54m 13s 298ms
10	DOC	949	02m 57s 865ms	24m 13s 265ms
11	DOC	587	01m 51s 256ms	15m 50s 085ms
12	PPS	441	01m 23s 974ms	12m 46s 338ms
13	PDF	293	00m 57s 576ms	07m 53s 870ms
14	XLS	48	00m 09s 395ms	01m 14s 698ms
15	XLS	20	00m 03s 710ms	00m 30s 085ms
16	JPEG	16	00m 02s 975ms	00m 23s 720ms
17	TXT	8	00m 01s 395ms	00m 11s 899ms

Quadro 38. Teste 10 Realizado com o algoritmo RSA com chave de 768

Teste 13 - Realizado com o Algoritmo RSA com tamanho de chave 1024				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	1h 12m 51s 665ms	1h 22m 46s 255ms
02	MP3	19.332	57m 48s 595ms	1h 15m 30s 785ms
03	EXE	15.365	47m 01s 378ms	87m 57s 660ms
04	PPT	7.559	25m 07s 890ms	78m 59s 735ms
05	MP3	5.355	16m 18s 975ms	73m 20s 460ms
06	JPEG	4.667	14m 16s 968ms	65m 56s 985ms
07	MPEG	3.580	10m 32s 110ms	60m 30s 289ms
08	PDF	2.181	06m 55s 233ms	59m 47s 478ms
09	JPEG	1.969	06m 16s 255ms	54m 13s 305ms
10	DOC	949	02m 57s 880ms	24m 13s 279ms
11	DOC	587	01m 51s 270ms	15m 50s 096ms
12	PPS	441	01m 23s 999ms	12m 46s 346ms
13	PDF	293	00m 57s 556ms	07m 53s 885ms
14	XLS	48	00m 09s 385ms	01m 14s 705ms
15	XLS	20	00m 03s 725ms	00m 30s 100ms
16	JPEG	16	00m 02s 995ms	00m 23s 736ms
17	TXT	8	00m 01s 400ms	00m 11s 906ms

Quadro 39. Teste 13 Realizado com o algoritmo RSA com chave de 1024

Teste 11 - Realizado com o Algoritmo DES com tamanho de chave 100				
Nr. consultas	**Extensão do Arquivo**	**Tamanho KB**	**Cifragem**	**Decifragem**
01	MPEG	42.958	01m 24s 786ms	01m 24s 407ms
02	MP3	19.332	00m 40s 340ms	00m 40s 232ms
03	EXE	15.365	00m 39s 208ms	00m 37s 755ms
04	PPT	7.559	00m 16s 255ms	00m 15s 426ms
05	MP3	5.355	00m 10s 466ms	00m 10s 573ms
06	JPEG	4.667	00m 09s 424ms	00m 09s 357ms
07	MPEG	3.580	00m 07s 005ms	00m 07s 079ms
08	PDF	2.181	00m 04s 363ms	00m 04s 379ms
09	JPEG	1.969	00m 03s 878ms	00m 03s 832ms
10	DOC	949	00m 01s 849ms	00m 01s 911ms
11	DOC	587	00m 01s 201ms	00m 01s 143ms
12	PPS	441	00m 00s 876ms	00m 00s 880ms
13	PDF	293	00m 00s 564ms	00m 00s 579ms
14	XLS	48	00m 00s 095ms	00m 00s 122ms
15	XLS	20	00m 00s 036ms	00m 00s 048ms
16	JPEG	16	00m 00s 048ms	00m 00s 032ms
17	TXT	8	00m 00s 019ms	00m 00s 017ms

Quadro 40. Teste 11 Realizado com o algoritmo DES com chave de 100