UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

"PROTEÇÃO CRIPTOGRÁFICA NA REDE DE COMPUTADORES DA POLÍCIA MILITAR DE PERNAMBUCO, UM ESTUDO DE CASO."

por

KÁTIA GARCIA PINTO

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como parte dos requisitos para a obtenção do grau de Mestre em Engenharia Elétrica.

ORIENTADOR: PROF. DR. VALDEMAR C. DA ROCHA Jr., Ph.D.

Recife, Agosto de 2000.

Universidade Federal de Pernambuco

Pós-Graduação em Engenharia Elétrica

PARECER DA COMISSÃO EXAMINADORA DE DEFESA DE DISSERTAÇÃO DE MESTRADO DE

KÁTIA GARCIA PINTO

TÍTULO

"PROTEÇÃO CRIPTOGRÁFICA NA REDE DE COMPUTADORES DA POLÍCIA MILITAR DE PERNAMBUCO, UM ESTUDO DE CASO"

A comissão examinadora composta pelos professores: VALDEMAR CARDOSO DA ROCHA JÚNIOR, DES/UFPE, RAFAEL DUEIRE LINS, DES/UFPE e MARCELO SAMPAIO DE ALENCAR, DE/UFPB, sob a presidência do primeiro, consideram a candidata **KÁTIA GARCIA PINTO APROVADA**.

Recife, 18 de agosto de 2000

Valdemar C. Rocha Jr.

VALDEMAR CARDOSO DA ROCHA JÚNIOR

Rafael Dueire Lins

RAFAEL DUEIRE LINS

Marcelo Sampaio de Alencar

MARCELO SAMPAIO DE ALENCAR

À minha amada e adorada mãe

Ceci

"Mestre não é quem sempre ensina, mas quem de repente aprende."

(Guimarães Rosa)

Aos amores da minha vida

Louyse e Roberto

AGRADECIMENTOS

Inicialmente, agradeço a Deus, pois, sem Ele, nada disso teria sido possível.

Agradeço com muito amor à minha querida mãe, Ceci, por todos os sacrifícios que fez com muita dedicação para que eu chegasse ao final dos meus estudos e conclusão deste trabalho, nos momentos de grande desânimo e cansaço, era nela que eu encontrava forças para continuar.

Agradeço ao meu querido e amado marido, Roberto pela compreensão e amor constantes, com permanente dedicação, apoiando e incentivando nos momentos mais difíceis, e a minha amada filha, Louyse, por existir.

Agradeço a todos os professores do Departamento de Eletrônica e Sistemas, da Universidade Federal de Pernambuco, em especial ao Prof. Dr. Valdemar Cardoso da Rocha Júnior, orientador da dissertação, pela grande oportunidade do seu acompanhamento no meu trabalho com muita paciência, confiança (acreditou que eu conseguiria terminar, apesar de todos os acidentes de percurso no decorrer do curso!), incentivo permanente, amizade e críticas construtivas ao desenvolvimento deste trabalho, ao Prof. Ricardo Campello, por sua dedicação nas infinitas aulas com muito apoio e fé no nosso potencial adormecido.

Agradeço de todo coração, a Andréa Tenório, por toda dedicação e carinho com que me tratou, nela tudo começou, com grande incentivo, paciência nos inúmeros pedidos de documentos, mas sempre atendida de pronto! Muito obrigada querida amiga!

Gostaria de expressar o meu mais sincero agradecimento ao Prof. Luiz Carlos Soares da Silva e ao Analista de Redes Bartolomeu Gustavo da Silva, pelas infinitas horas de discussão que me levou a ter um melhor conhecimento sobre redes de computadores, pelo acompanhamento, pela orientação e pelas sugestões valiosas ao desenvolvimento deste trabalho.

Agradeço de forma muito especial às minhas amigas do mestrado. Consegui encontrar neste Departamento muito mais que amizade, formamos uma verdadeira família, o companheirismo e incentivo de Maria de Lourdes Alcoforado, Luciana Espínola e Simone, sem dúvida foram as grandes alavancas para obtenção do êxito em muitas disciplinas, além de tornar o nosso espaço de estudo prazeroso.

Agradeço também a todas as pessoas que, de alguma forma, contribuíram, direta ou indiretamente, na conclusão deste trabalho e que, porventura, não tenham sido citadas; a elas, o meu carinho e reconhecimento.

Não poderíamos deixar de agradecer aos meus colegas de trabalho da Polícia Militar, em especial ao Ten. Marcone Feliciano de Moura, pela amizade e companheirismo demonstrados para que eu conseguisse terminar o mestrado.

Resumo da Dissertação apresentada à UFPE como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica.

"PROTEÇÃO CRIPTOGRÁFICA NA REDE DE COMPUTADORES DA POLÍCIA MILITAR DE PERNAMBUCO, UM ESTUDO DE CASO."

Kátia Garcia Pinto

Agosto/2000

Orientador: Prof. Dr. Valdemar Cardoso da Rocha Jr., Ph.D.

Área de Concentração: Comunicações.

Palavras-chave: Criptografia, Gerenciamento de Chaves.

Número de Páginas: 203

Este trabalho foi motivado pela preocupação em salvaguardar informações sigilosas que transitam em rede nos diversos setores da Polícia Militar de Pernambuco, principalmente após a implantação da Rede Corporativa, a qual, ao mesmo tempo em que trará fluidez significativa ao acesso às informações, tornará tais informações ainda mais vulneráveis por parte de pessoas não autorizadas. Nesse contexto, foi abordado um dos aspectos relacionados à política de segurança em redes de computadores, que consiste na introdução de técnicas criptográficas no sistema computacional. Sistemas criptográficos desempenham, atualmente, um papel extremamente relevante no contexto da segurança de dados, permitindo a implementação de sistemas de comunicação e de armazenamento de dados, que apresentam elevados níveis de segurança. Neste trabalho apresentamos uma aplicação da cifra SAFER, proposta pelo Prof. James Massey, associada à técnica de geração pública de chaves secretas proposta por Diffie e Hellman, descrevendo suas propriedades, a forma de emprego e o sistema de gerenciamento de chaves, destacando, nos aspectos conceituais, segurança de dados, criptografia e rede de computadores. Foi feito um estudo preliminar, no campo a que se destina a rede corporativa em tela, com a aplicação de observações diretas e questionário aos comandantes, diretores e chefes de seções sobre segurança de informações sigilosas. As sugestões contidas neste trabalho visam amenizar os efeitos da insegurança na rede, lembrando que outras medidas de segurança também se fazem necessárias à salvaguarda das informações.

Abstract of Dissertation presented to UFPE as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

"CRYPTOGRAPHIC PROTECTION IN THE COMPUTER NETWORK OF THE MILITARY POLICE OF PERNAMBUCO, A CASE STUDY."

Kátia Garcia Pinto

August / 2000

Supervisor: Prof. Dr. Valdemar Cardoso da Rocha Jr., Ph.D.

Area of Concentration: Communications.

Keywords: Cryptography, Key's Management.

Number of Pages: 203.

This work was motivated by the concern of safeguarding sensitive information that flows through a network in the diverse sectors of the Military Policy of Pernambuco, mainly after the deployment of the Corporation Network, which, simultaneously will speed up significantly the access to information, as well as will become more vulnerable to non authorized personnel. In this context, it addresses one of the aspects related to security policy in computer networks, consisting in the introduction of cryptographic techniques in the computational system. Cryptographic systems currently play an extremely relevant role in the context of data security, allowing the implementation communication and data storage systems, enjoying high levels of security. This work presents an application of the cipher SAFER, proposed by Prof. James Massey, associated with the technique of public generation of private keys, proposed by Diffie and Hellman, describing their properties, their application and a key management system, with emphasis on conceptual aspects, data security, cryptography and computer networks. A preliminary study was conducted, in the area concerned with the particular corporative network, with the application of direct observations and a questionnaire to the commanders, directors and heads of sections on security of confidential information. The suggestions contained in this work aim at to reduce the effect of the lack of reliability in the network, bearing in mind that other security measures of security are also necessary to safeguard important information.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

ATM – ASSYNCHRONOUS TRANSFER MODE
CERT - COMPUTER EMERGENCY RESPONSE TEAM
COPOM - CENTRAL DE OPERAÇÕES DA POLÍCIA MILITAR
CPC - CATÁLOGO PÚBLICO CUSTODIADO
CPD - CENTRO DE PROCESSAMENTO DE DADOS
CSMA/CD – CARRIER SENSE MULTIPLE ACCESS WITH COLLISION DETECTION
DES - DATA ENCRYPTION STANDARD
DTE - DATA TERMINAL EQUIPMENT
ERB - ESTAÇÃO RÁDIO BASE
IAB - INTERNET ACTIVITIES BOARD
IEEE - INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS
IPX - INTERNETWORK PACKET EXCHANGE
ISO - INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
LANs – LOCAL AREA NETWORKS
LLC - SUBCAMADA DE CONTROLE DE ENLACE LÓGICO
MAC - CONTROLE DE ACESSO AO MEIO
MANS – METROPOLITAN AREA NETWORKS
NBS - NATIONAL BUREAU OF STANDARDS
NIST - NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
NSA - NATIONAL SECURITY AGENCY
OSI - OPEN SYSTEMS INTERCONNECTION
PMPE - POLÍCIA MILITAR DE PERNAMBUCO
QCG – QUARTEL DO COMANDO GERAL
RFC - REQUEST FOR COMMENTS
RM-OSI- REFERENCE MODEL FOR OPEN SYSTEMS INTERCONNECTION
SAFER - SECURE AND FAST ENCRYPTION ROUTINE
SGC - SISTEMA DE GERENCIAMENTO DE CHAVES
SPX - SEQUENCED PACKET PROTOCOL
TCP/IP – TRANSPORT CONTROL PROTOCOL / INTERNET PROTOCOL
UDP - USER DATAGRAM PROTOCOL
UTP - UNSHIELDED TWISTED PAIR
WANs – WIDE AREA NETWORKS

CAPÍTULO 1

INTRODUÇÃO

Ao longo do século XX a tecnologia dominante tem consistido na coleta, no processamento, na distribuição e no armazenamento da informação. À proporção em que o tempo passa, as diferenças entre coletar, transportar, armazenar e processar informações estão rapidamente desaparecendo.

Organizações com centenas de escritórios espalhados em uma vasta área geográfica esperam, de forma rotineira, poder examinar a situação dos seus negócios, até do seu escritório mais remoto, com um simples premer de botão.

Durante as duas primeiras décadas de sua existência, os sistemas de computadores eram altamente centralizados, em geral em uma única sala grande.

A fusão dos computadores e das comunicações teve uma profunda influência sobre a forma como os computadores são organizados atualmente. O velho modelo de um único computador servindo a todas as necessidades computacionais da organização está rapidamente sendo substituído por outro no qual um grande número de computadores separados, mas interconectados, executam a tarefa. Esses sistemas são chamados de redes de computadores.

Aliada a toda esta tecnologia, surge uma nova preocupação referente à segurança na rede de computadores. Assim como a rede provê uma série de facilidades e dinamismo os serviços executados, pode também ser alvo de ameaças relacionadas à segurança. A ausência de uma política de segurança pode fazer com que uma Organização

se torne vulnerável a ataques às informações sigilosas, os quais poderão deixá-la em uma situação altamente embaraçosa e causar imensos prejuízos.

Ante o exposto e tendo em vista a implantação da Rede Corporativa na Polícia Militar de Pernambuco (PMPE), esta dissertação aborda uma das medidas relacionadas à segurança da rede de computadores por intermédio da introdução de técnicas criptográficas e gerenciamento de chaves.

O resultado final, além da introdução das técnicas criptográficas, visa também educação e investigação, ou seja, visa melhorar o nível geral de competência técnica; dedicando tempo e recursos a essa tarefa. Por meio de um planejamento cuidadoso e testando as soluções propostas, possibilitamos alcançar os benefícios das telecomunicações para a Corporação de uma maneira segura e prudente.

A adoção de medidas para a segurança de conexões com a rede corporativa poderá exigir investimentos significativos em termos de tempo e esforço. Portanto, é necessário comparar custos e benefícios em relação a todas as medidas de controle de segurança na rede a serem consideradas. Um estudo de segurança na rede sempre tenta comparar as vantagens das soluções de segurança em relação a suas desvantagens e custos. Resumindo, este trabalho examina as questões de segurança como um pré-requisito para a confiabilidade dos serviços executados pela polícia.

1.1 Organização da Dissertação

Nosso trabalho apresenta uma visão geral acerca do Teleprocessamento da Polícia Militar de Pernambuco, abordando alguns conceitos básicos, necessários a um bom entendimento do objetivo final desta dissertação, que é a introdução de um dos mecanismos de segurança de dados, a criptografia e o sistema de gerenciamento de chaves, os quais foram organizados da seguinte maneira.

No Capítulo 2, descrevemos conceitos básicos, em nível introdutório, sobre Redes de Computadores, abordando os tipos de redes; as topologias; os protocolos de comunicação; a arquitetura de redes quanto ao modelo RM-OSI e sua aplicabilidade em redes locais; a arquitetura da Internet TCP/IP e a rede Ethernet. O objetivo deste capítulo é facilitar o entendimento dos termos referenciados no Capítulo 6.

No Capítulo 3, introduzimos conceitos básicos sobre Segurança em Redes, mais especificamente em Criptografia. Dentro desse contexto, quanto à Segurança em Redes, temos os conceitos sobre ameaças e ataques; política de segurança; serviços de segurança e os mecanismos de segurança.

Como não é possível em um trabalho desta natureza tratar de todos os mecanismos de segurança, daremos enfoque à adoção de um dos mecanismos de segurança, a criptografia, e passaremos, então, a descrever de forma concisa seu conceito; terminologias adotadas; um relato histórico de como tudo começou; os sistemas criptográficos existentes; conceito de assinatura digital; comércio eletrônico e criptoanálise.

No Capítulo 4, apresentamos o projeto de implementação de uma técnica criptográfica, o SAFER *(Secure And Fast Encryption Routine)*, como uma das medidas adotadas na política de segurança na rede de computadores da Polícia Militar, em que descrevemos, inicialmente, o pressuposto norteador dessa escolha entre tantas técnicas criptográficas existentes e, a seguir, suas propriedades, seu funcionamento, suas vantagens e sua disponibilização.

Descrevemos, ainda, um sistema de gerenciamento de chaves, que foi desenvolvido para aplicação direta na Polícia Militar, usando programação em Visual Basic e, por meio de exemplo, apresentamos o emprego do SAFER.

No Capítulo 5, abordamos o Teleprocessamento da Polícia Militar, em que descrevemos o início da informática na polícia e o seu despertar para uma nova era tecnológica, com a implantação de uma rede corporativa, facilitando o atendimento da sua

destinação legal, que é a segurança pública. Em contrapartida, nos deparamos com as questões envolvendo a proteção dos dados confidenciais e sigilosos.

Como no caso da polícia, todo administrador somente deverá fazer o que estiver expressamente autorizado em lei, pois na Administração Pública só é permitido fazer o que a lei autoriza, diferentemente da esfera particular, na qual é permitida a realização de tudo o que a lei não proíba. Com efeito, se a atividade do administrador não for autorizada por lei, é ilícita. Baseado neste Princípio da Legalidade, neste capítulo, apresentaremos uma fundamentação jurídica da proposta oferecida nesta dissertação, quanto à implantação de um sistema de segurança na rede de computadores da polícia militar, tendo em vista a necessidade de um embasamento legal à validade de qualquer projeto na área do serviço público.

No Capítulo 6, apresentamos a Rede Corporativa da Polícia Militar de Pernambuco. Um relato de como surgiu a necessidade da implantação da rede, é abordado, em seguida fazemos uma breve exposição sobre o estado atual da rede de comunicações da polícia, as aplicações utilizadas e a antiga arquitetura de rede.

Em seguida apresentamos o projeto da rede corporativa, por meio de mapas demonstrativos, abordando seus aspectos gerais de formação e disposição no Quartel do Comando Geral (QCG). Um estudo preliminar foi realizado para fundamentar a análise de risco, quanto aos aspectos de segurança de dados, em função da implantação da rede.

Finalmente, nas conclusões, apresentamos alguns comentários gerais sobre o trabalho e a respectiva sugestão para o desenvolvimento de uma política de segurança na rede de computadores da PMPE.

No Apêndice A, apresentamos informações adicionais sobre os aspectos jurídicos da segurança em informática em outros países; no Apêndice B, temos registrado o questionário formulado para o levantamento de dados sobre segurança na polícia militar; no Apêndice C, detalhamos a utilização do SAFER e o seu respectivo código fonte e; finalmente, no Apêndice D, apresentamos o código fonte em Visual Basic, do Sistema de Gerenciamento de Chaves desenvolvido para a polícia militar.

CAPÍTULO 2

REDES DE COMPUTADORES

A evolução tecnológica e a conseqüente diminuição dos custos dos computadores tornaram cada vez mais atraente a distribuição do poder computacional em módulos processadores localizados em diversos pontos de uma organização. A necessidade de interconexão desses módulos processadores, para permitir o compartilhamento de recursos de *hardware* e *software* e a troca de informações entre seus usuários, criou o ambiente propício para o desenvolvimento das *redes de computadores*.

No princípio, as redes foram implementadas empiricamente, contudo, nas décadas de 1970 e 1980, um conjunto de conhecimentos foi adquirido, tornando possível o seu projeto sistemático.

A evolução contínua da microeletrônica e da tecnologia de comunicações vem, desde então, abrindo novas fronteiras. O emprego de sistemas de comunicação capazes de transportar dados a altas velocidades e a grandes distâncias permitiu a introdução do conceito de *rede única*, capaz de transportar de forma integrada as diferentes mídias de vídeo, áudio, imagens estática e texto [11].

Faremos, neste capítulo, uma exposição de alguns conceitos básicos sobre Redes de Computadores, visando facilitar a compreensão de alguns termos utilizados no Capítulo 6, em que abordaremos o projeto da rede corporativa da Polícia Militar de Pernambuco.

2.1 Conceitos Básicos

A rede de computadores é formada por um conjunto de módulos processadores capazes de trocar informações e compartilhar recursos, interligados por um sistema de comunicação, conforme a figura 1.

FIGURA 1 - REDE DE COMPUTADORES

O *sistema de comunicação* vai se constituir de um arranjo topológico, interligando os vários módulos processadores por enlaces físicos, através dos *meios de transmissão* e de um conjunto de regras, com o fim de organizar a comunicação, os *protocolos de comunicação*.

As Redes Locais (*LANs – Local Area Networks*) originadas dos ambientes de institutos de pesquisa e universidades, surgiram para viabilizar a troca e o compartilhamento de informações e dispositivos periféricos, preservando a independência das várias estações de processamento e permitindo a integração em ambientes de trabalho cooperativo. Pode-se caracterizar uma rede local como sendo uma rede que permite a interconexão de equipamentos de comunicação de dados numa pequena região. Esta definição é bastante vaga, considerando "pequenas distâncias" entre 100m e 25 km, embora as limitações associadas às técnicas utilizadas em redes locais não imponham

limites a essas distâncias. Outra característica típica das LANs são as altas taxas de transmissão (de 0,1 a 100 Mbps) e as baixas taxas de erro (de 10^{-8} a 10^{-11}).

Quando a distância de ligação começa a atingir distâncias metropolitanas, chamamos estes sistemas de Redes Metropolitanas (*MANs – Metropolitan Area Networks*).

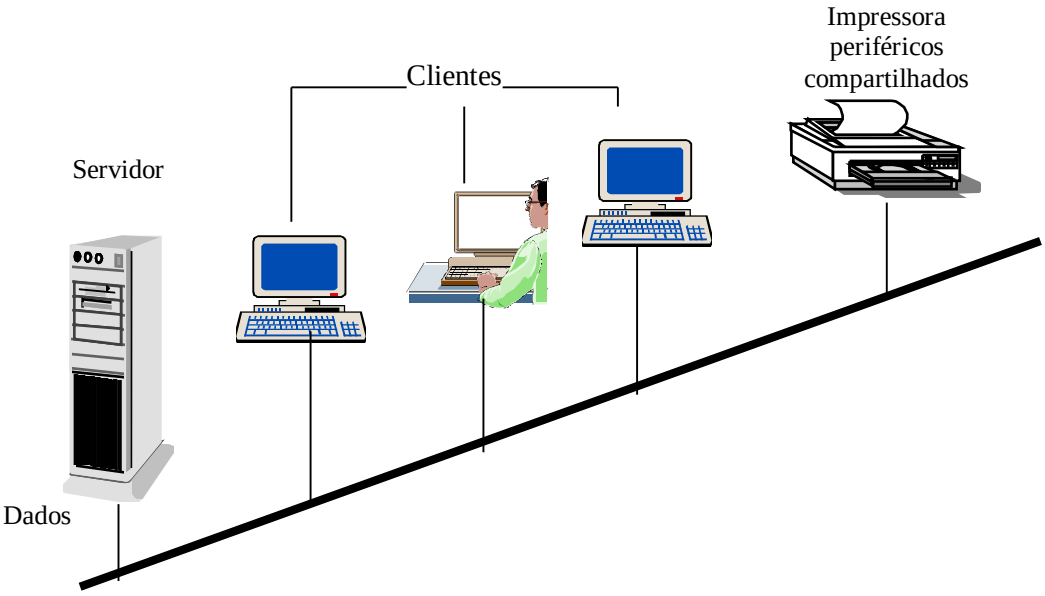
As Redes Geograficamente Distribuídas (*WANs – Wide Area Networks*), ou Redes de Longa Distância, são utilizadas para compartilhar recursos especializados com uma maior comunidade de usuários geograficamente dispersos. Em função dos custos de comunicação serem bastante elevados, circuitos para satélites e enlaces de microondas, estas redes são, em geral, públicas, isto é, o sistema de comunicação, chamado *sub-rede de comunicação*, é mantido, gerenciado e de propriedade de grandes operadoras (públicas ou privadas), sendo seu acesso público. Começam a surgir redes ATM de alta velocidade na faixa de Gbps [8].

2.2 Tipos de Redes

Os tipos de redes são as *Redes Workgroup* e *Redes Baseadas em Servidores*. Em geral, todas as redes têm certos componentes, funções e características em comum, tais como:

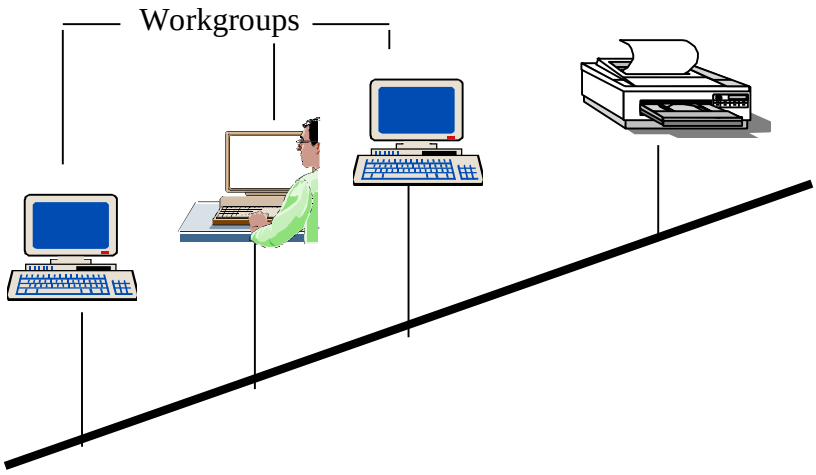
- servidor – computador que fornece o compartilhamento dos recursos para os usuários da rede;
- cliente – computadores que acessam os recursos compartilhados na rede, providos pelo servidor;
- meio – vias, como os computadores são conectados;
- dados compartilhados – arquivos providos pelo servidor através da rede;
- impressoras compartilhadas – outros recursos providos pelo servidor;
- recursos – arquivos, impressora ou outros itens que possam ser usados pelos usuários da rede.

FIGURA 2 - VISÃO GERAL DE UMA REDE



A *Rede Workgroup* (figura 3) é apropriada para ambientes onde há menos do que 10 usuários ficando dispostos na mesma área física; a organização e a rede não têm previsão de crescimento. Sua segurança é descentralizada, ou seja, cada usuário faz a sua, tornando difícil seu controle. Neste sistema de rede, cada usuário administra seu computador, podendo agir como servidor e cliente. O usuário deve, então, ser treinado a habilitar as funções de servidor e agir como usuário e administrador do seu computador, o que envolve uma variedade de tarefas, incluindo: gerenciar usuários e segurança, disponibilizar recursos, manter as aplicações e dados, instalar e atualizar os *softwares* de aplicações.

FIGURA 3 - REDE WORKGROUP



A *Rede Baseada em Servidores* (figura 2) é definida por um modelo padrão com mais de 10 usuários; neste caso, o servidor não deve ser utilizado como cliente e possui grande capacidade para acomodar o crescimento das necessidades do usuário. Existem servidores especializados em arquivo e impressão, aplicação, correio, fax e comunicação. Nesta rede, o servidor e o sistema operacional trabalham juntos como uma unidade. Dentre as vantagens apresentadas pelas redes baseadas em servidores, temos:

- compartilhamento dos recursos – administração e controle centralizados;
- segurança - administrada por um gerente que a define e aplica a cada usuário da rede;
- *backup* – como os dados estão centralizados em um ou poucos servidores, torna-se fácil programar e executar as rotinas de *backup* do sistema;
- redundância – os dados em qualquer servidor podem ser duplicados e solicitados, se houver problemas com os dados originais;
- número de usuários – pode atender a milhares de usuários.

Redes Combinadas são as que procuram reunir as melhores características das abordagens *workgroups* e cliente-servidor. Nas redes combinadas, dois sistemas operacionais trabalham juntos, sendo o sistema operacional do servidor (Windows NT, Novell) responsável por compartilhar os dados e as aplicações, e o computador cliente pode usar o sistema operacional Windows NT Workstation ou Windows 95/98. Ambos podem acessar um servidor designado e, simultaneamente, compartilhar seus recursos e disponibilizar seus dados, quando necessário [14].

2.3 Topologias de Redes

O sistema de comunicações de uma rede se constitui de um arranjo topológico interligando os vários módulos processadores por meio de enlaces físicos (meios de transmissão) e de um conjunto de regras, com o fim de organizar a comunicação (protocolos).

Nesta seção, apresentaremos os tipos de arranjo topológico utilizados em redes. A topologia de uma rede de comunicação refere-se à forma como os enlaces físicos e os nós de comutação estão organizados, determinando os caminhos físicos existentes e utilizáveis entre quaisquer pares de estações conectadas a essa rede.

Topologias Padrão

Os caminhos alternativos entre os nós de uma rede se fazem necessários para aumentar a confiabilidade e o desempenho. Em redes locais e metropolitanas, meios de transmissão de alta velocidade, de baixa taxa de erro, de baixo custo e privados podem ser usados. Topologias muitas vezes inviáveis em ambientes geograficamente distribuídos podem ser utilizadas em redes locais e metropolitanas. Apresentaremos, a seguir, as topologias mais utilizadas nessas redes:

- Estrela (*Star*);
- Anel (*Ring*);
- Barramento (*Bus*).

2.3.1 Topologia em Estrela

Numa rede com topologia em estrela, cada nó é interligado a um nó central (mestre), através do qual as mensagens devem passar. Tal nó age, assim, como centro de controle da rede, interligando os demais nós (escravos). Nada impede que haja comunicações simultâneas, desde que as estações envolvidas sejam diferentes.

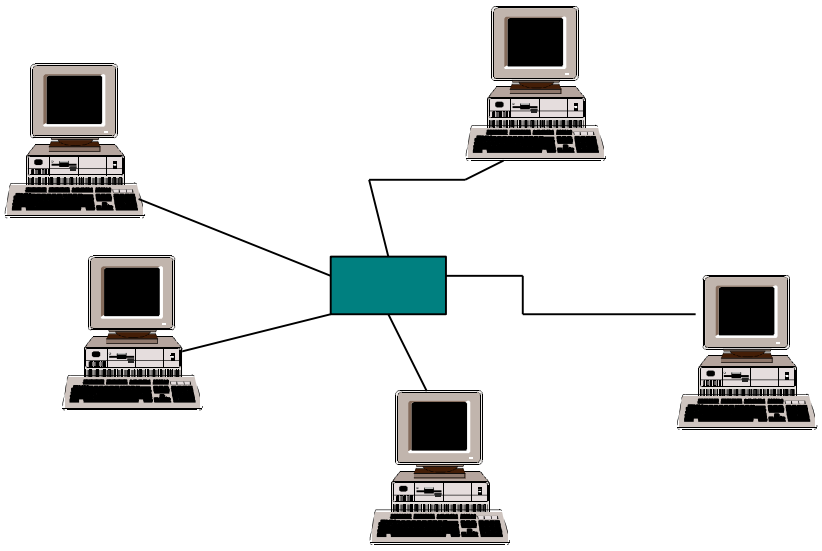


FIGURA 4 - TOPOLOGIA EM ESTRELA

O nó central da rede em estrela tem tanto a função de gerência de comunicação como facilidades de processamento de dados. A função do nó central é o chaveamento (ou comutação) entre as estações que desejam se comunicar, denominado *comutador* ou *switch.*

Confiabilidade é um problema nas redes em estrela. Falhas em um nó escravo apresentam um problema mínimo de confiabilidade, uma vez que o restante da rede ainda continua em funcionamento. Falhas no nó central, por outro lado, podem ocasionar a parada total do sistema.

Outro problema da rede em estrela é relativo à modularidade. A configuração pode ser expandida até um certo limite imposto pelo nó central.

2.3.2 Topologia em Anel

Uma rede em anel consiste em estações conectadas através de um caminho fechado. O anel não interliga as estações diretamente, mas consiste em uma série de

repetidores ligados por um meio físico, sendo cada estação ligada a esses repetidores, conforme apresenta a figura 5.

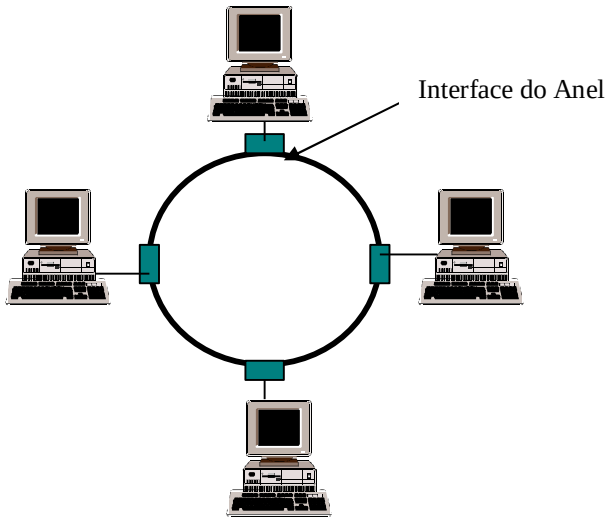


FIGURA 5 - TOPOLOGIA EM ANEL

O protocolo para troca de mensagens é o *Token Passing*. Não há terminador, o sinal viaja em uma direção e passa através de cada computador, cada computador agindo como um repetidor, reativando o sinal e enviando-o ao próximo computador.

2.3.3 Topologia em Barramento

A topologia em barramento é aquele em que todas as estações (nós) de trabalho e os dispositivos de rede se ligam ao mesmo meio de transmissão.

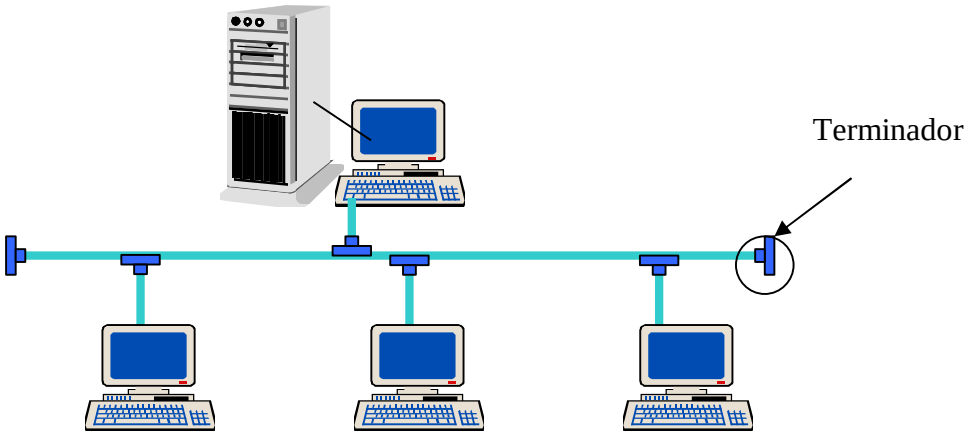


FIGURA 6 - TOPOLOGIA EM BARRAMENTO

Ao contrário das outras topologias que descrevemos até aqui, que são configurações ponto a ponto (isto é, cada enlace físico de transmissão conecta apenas dois dispositivos), a topologia em barra tem uma configuração multiponto (cada enlace físico de transmissão conecta três ou mais dispositivos). Cada dispositivo é conectado diretamente a um cabo principal, chamado de espinha dorsal (*backbone*) da rede. O *backbone* interconecta cada dispositivo da rede e também se liga com outras redes.

A comunicação, no barramento, é feita quando são enviados para todos os computadores da rede, ou seja, cada nó conectado à barra pode ouvir todas as informações transmitidas. Então, só o computador endereçado, um grupo deles (multicast) ou todos (broadcast), aceitam a mensagem e decodificam o sinal original.

Apenas um computador pode enviar uma mensagem por vez, afetando o desempenho quando aumenta o número de computadores ligados ao barramento.

O barramento é uma topologia passiva, o que significa que os computadores não são responsáveis pelos movimentos dos dados entre si [8].

2.4 Hubs e Switches

O *hub* é o componente central em uma topologia de estrela, podendo ser classificado em:

- *hubs ativos,* que são aqueles que regeneram e retransmitem os sinais exatamente como um repetidor. Na verdade, como os hubs normalmente possuem entre oito e doze portas para conectarem os computadores de rede, algumas vezes são chamados repetidores multiportas. Os *hubs ativos* exigem alimentação elétrica para operar;
- *hubs passivos,* são como painéis de fiação, agem como pontos de conexão e não amplificam nem regeneram o sinal; o sinal passa através do hub. Os *hubs passivos* não exigem alimentação elétrica para operar;
- *hubs híbridos* são hubs avançados que acomodam vários tipos diferentes de cabos, uma rede baseada em *hub* pode ser expandida conectando-se mais de um *hub* .

A demanda por maiores taxas de transmissão e melhor utilização dos meios físicos, aliados à evolução contínua da microeletrônica, começou a alterar a construção dos *hubs*, no sentido de que os mesmos não implementassem somente a utilização do meio compartilhado, mas também possibilitassem a troca de mensagens entre várias estações simultaneamente. Dessa forma, as estações poderiam obter para si taxas efetivas de transmissão bem maiores. Esse tipo de elemento, também central, é denominado *switch.*

A topologia de uma rede irá determinar, em parte, o método de acesso utilizado. Métodos de acesso são necessários para regular a utilização dos meios físicos compartilhados. A forte tendência de utilização de *hubs* nas instalações físicas das redes corresponde, fisicamente, à implantação de uma topologia em estrela (em que o *hub* é o componente central). Esta tendência é explicada pela crescente necessidade de melhorar o gerenciamento e a manutenção nessas instalações. A topologia em estrela apresenta uma baixa confiabilidade, porém os avanços da eletrônica já permitem que se construam equipamentos bastante confiáveis, viabilizando este tipo de topologia [8].

A utilização de *hubs* não exige, necessariamente, que as interfaces das estações com a rede a percebam como uma topologia em estrela. O funcionamento continua a ser como no acesso a um barramento ou a um anel, com os seus respectivos métodos de acesso. Sendo assim, podemos diferenciar dois tipos de topologias: uma topologia lógica, que é aquela observada sob o ponto de vista das interfaces das estações com a rede (que inclui o método de acesso), e uma topologia física, que diz respeito à configuração física utilizada na instalação da rede.

2.5 Protocolos de Comunicação

A especificação de um *software* de comunicação deve prever diversos aspectos referentes aos serviços que o sistema oferece. Na verdade, a simples transferência de um arquivo de uma máquina para outra envolve diversas etapas que, se analisadas em conjunto, possuem uma complexidade difícil de controlar. Para diminuir esta complexidade, o *software* de comunicação é dividido em *camadas ou níveis,* cada camada executando um conjunto definido de funções.

O nível *n* de uma máquina mantém uma conversação com o nível *n* de outra máquina. As regras e convenções desta conversação são definidas como o *protocolo do nível n.*

Na realidade, nenhum dado é transferido diretamente de um nível *n* em uma máquina para o nível *n* de outra máquina (exceto no nível mais baixo). Cada nível passa os dados e as informações de controle para o nível imediatamente inferior, até que o nível 1 seja alcançado. No nível 1 existe uma comunicação física com a outra máquina, ao contrário da comunicação virtual usada pelos outros níveis.

Desta forma, podemos distinguir dois conceitos importantes: protocolo e serviço. Protocolos são regras de diálogo entre entidades comunicantes de sistemas diferentes e estão relacionados à *comunicação virtual* que existe entre cada um dos níveis

de protocolo. Os serviços dizem respeito à comunicação entre entidades em um mesmo sistema e estão relacionados à *comunicação real* entre duas camadas adjacentes.

O conjunto de níveis ou camadas de protocolos é denominado *arquitetura da rede* (figura 7). A especificação da arquitetura deve conter informações suficientes para que um implementador possa escrever o *software* ou construir o *hardware* de cada camada de tal forma que obedeça corretamente ao protocolo associado [41].

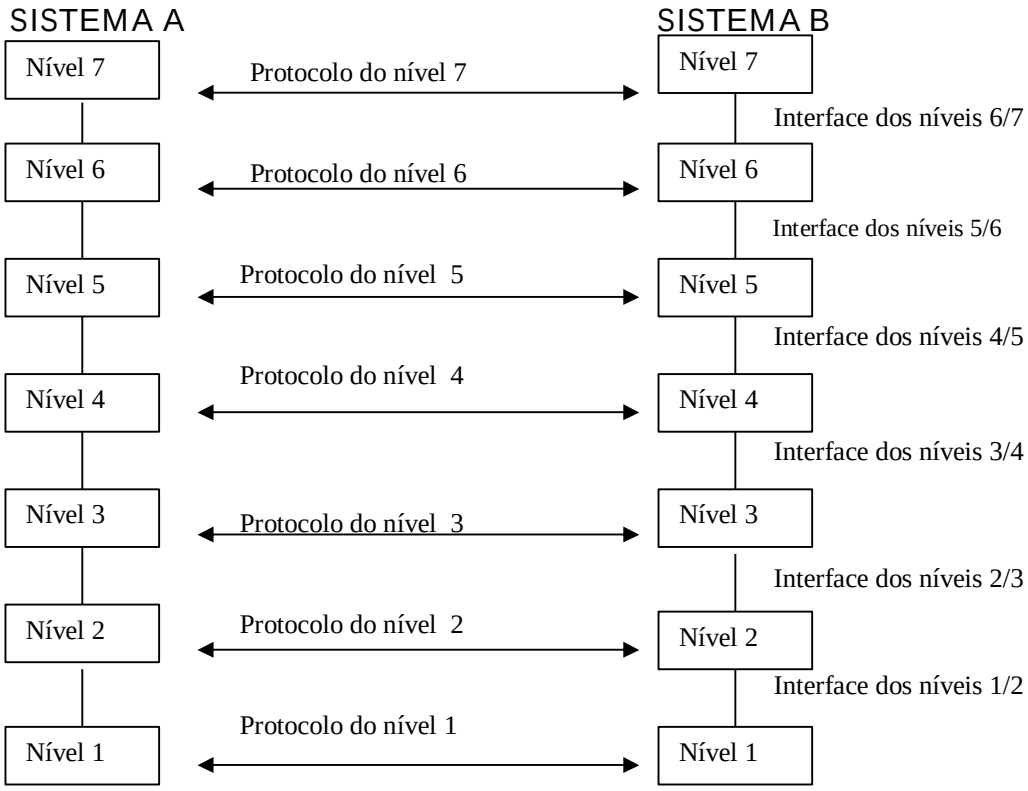


FIGURA 7 - ARQUITETURA DE PROTOCOLOS EM SETE NÍVEIS

2.6 Arquitetura de Redes

Para permitir o intercâmbio de informações entre computadores de fabricantes distintos, tornou-se necessário definir uma arquitetura única; e, para garantir que nenhum fabricante levasse vantagem em relação aos outros, a arquitetura teria que ser aberta e pública. Foi com esse objetivo que a International Organization for Standardization (ISO)

definiu o modelo denominado Reference Model for Open Systems Interconnection (OSI) [ISO 84, ISO 92], que propõe uma estrutura com sete níveis (figura 7) como referência para a arquitetura dos protocolos de redes de computadores.

Embora o modelo OSI, da ISO, possa ser usado tanto em redes de longa distância quanto em redes locais, em princípio, foi pensado para o uso em redes de longa distância. O Institute of Electrical and Electronics Engineers (IEEE) define padrões para os níveis físicos e enlace de redes locais de computadores.

A coexistência de redes heterogêneas (locais, metropolitanas e de longa distância) fez com que se tornasse necessário definir uma arquitetura voltada para a interconexão dessas redes. Uma arquitetura importante no contexto de interconexão de redes heterogêneas é a Arquitetura Internet, que se baseia na família de protocolos TCP/IP [8, 11].

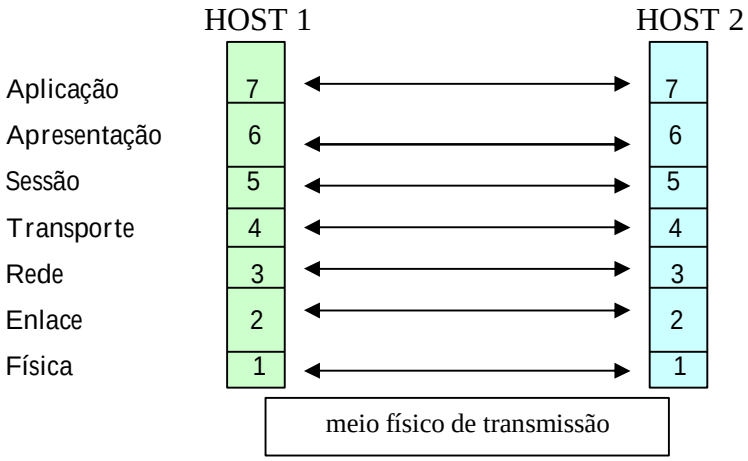
2.6.1 Arquitetura RM-OSI

A ISO (International Organization for Standardization) é uma organização internacional, fundada em 1946, que tem por objetivo a elaboração de padrões internacionais. Os membros da ISO são órgãos nacionais de padronização dos 89 países membros. O representante do Brasil na ISO é a ABNT.

A necessidade de fornecer uma base comum que permita o desenvolvimento coordenado de padrões para interconexão de sistemas, fez com que a ISO estabelecesse o Modelo de Referência para a Interconexão de Sistemas Abertos (RM-OSI).

Para reduzir a complexidade de projeto, a maioria das redes é organizada em camadas ou níveis, cada uma construída sobre sua predecessora. A arquitetura RM-OSI é formada por 7 camadas, conforme mostra a figura 8.

FIGURA 8 - AS SETE CAMADAS DO RM-OSI



A idéia básica desta arquitetura é que cada uma das sete camadas forneça serviços de comunicação, com certo grau de confiabilidade, à camada imediatamente superior. A camada *n* (*n* = 1, 2,..., 7), num determinado sistema, *conversa* com a camada *n* em um outro sistema, obedecendo a regras e convenções do protocolo da camada *n*. Os interlocutores, dentro de uma mesma camada, são conhecidos como *entidades.*

As unidades de dados trocadas na comunicação entre duas entidades de uma camada *n* não são transferidas diretamente entre elas. Em vez disso, cada camada passa dados e informações de controle para a camada inferior, até chegar à camada física, onde os dados são realmente transferidos pelo meio de transmissão até o outro sistema. Do outro lado, o processo se inverte: cada camada retira as informações que lhe pertencem e vai repassando para a camada superior o campo de dados da unidade recebida, até chegar à camada mais alta.

No par de camadas adjacentes existe uma interface que define as operações primitivas e os serviços que a camada inferior oferece à camada superior.

2.6.2 Arquitetura IEEE 802

As distâncias limitadas a que são destinadas às redes locais, permitem que seu protocolo de nível físico possa se dar ao luxo de utilizar um meio de alta velocidade e baixíssima taxa de erros.

Em redes locais, as regras que disciplinam o acesso ao meio físico, para transmissão de dados, são chamadas protocolos de acesso. Nas redes locais, a transmissão dos dados é feita por difusão (todas as estações recebem todos os pacotes), ou elas possuem roteamento único.

O projeto IEEE 802 teve origem na Sociedade de Computação do Instituto de Engenheiros Eletricistas e Eletrônicos dos EUA. O objetivo foi o estabelecimento de uma arquitetura padrão, orientada para o desenvolvimento de redes locais, as quais apresentassem as seguintes características:

- correspondência máxima com o RM-OSI;
- interconexão eficiente de equipamentos a um custo moderado;
- implantação da arquitetura a custo moderado.

A estratégia adotada na elaboração da arquitetura IEEE 802 é a de definir mais de um padrão, de forma que atenda aos requisitos dos sistemas usuários da rede. Na verdade, a arquitetura IEEE 802 pode ser vista como uma adaptação das duas camadas inferiores da arquitetura RM-OSI da ISO, de forma que existam 3 camadas, ou seja, uma equivalente à camada física e duas subcamadas que, juntas, equivalem à camada de enlace. Elas são assim denominadas:

- camada física (PHY);
- subcamada de controle de acesso ao meio (MAC);
- subcamada de controle de enlace lógico (LLC).

a) Camada Física

Esta camada tem como função prover os serviços básicos de transmissão e recepção de *bits* através de conexões físicas. Assim, ela define as características elétricas (níveis de tensão, impedância etc.), as características mecânicas (tipos de conectores, dimensões do suporte físico de transmissão etc.) e as características funcionais e de procedimentos (velocidade de transferência de *bits*, inicialização das funções de transmissão e recepção de bits etc.) das conexões físicas.

b) Subcamada de Controle de Acesso ao Meio

A subcamada de Controle de Acesso ao Meio (MAC), da arquitetura IEEE 802, especifica os mecanismos que permitem gerenciar a comunicação em nível de enlace de dados.

A existência da subcamada MAC na arquitetura IEEE 802 reflete uma característica própria das redes locais, que é a necessidade de gerenciar enlaces de dados com origens e destinatários múltiplos num mesmo meio físico de transmissão, como no caso das topologias em anel e barramento. Além disso, a existência da subcamada MAC permite o desenvolvimento da subcamada superior (LLC) com certo grau de independência da camada física, no que diz respeito à topologia e ao meio de transmissão propriamente dito [11].

Os mecanismos de controle de acesso distribuído apresentam uma forte dependência quanto à topologia da sub-rede de comunicação. Alguns mecanismos de acesso ao meio de transmissão alvo de padronização incluem:

- acesso múltiplo ao barramento ou CSMA/CD;
- passagem de ficha (permissão) em anel ou TOKEN-RING;
- passagem de ficha (permissão) em barramento ou TOKEN-BUS.

Apresentaremos, a seguir, as principais características dos dois métodos de acesso mais conhecidos e utilizados em redes locais.

TOKEN-RING

Introduzida em 1984 como parte da solução de conectividade da IBM para ambiente computacional, envolve: PC, médio porte, *mainframes*. O objetivo da versão IBM foi permitir que uma estrutura simples de fiação, usando cabo par trançado e tomada de paredes, conectasse um computador a uma fiação principal localmente centralizada.

Esta técnica é uma implementação do padrão IEEE 802.5, cujo método de acesso é o *token passing*. Sua arquitetura típica é o anel físico. Na implementação Token Ring da IBM anel em estrela, o anel físico é montado sobre um *hub* central. O anel lógico é representado pela passagem do *token* entre os computadores.

O *token* é uma formação predeterminada de bits que permite ao computador colocar dados ao meio físico. Quando o computador é ligado, a rede gera um *token.* O *token* viaja pelo anel, através dos computadores, até que um deles sinaliza que quer transmitir e toma o controle do *token.* Um computador não pode transmitir, a menos que esteja de posse do *token*, e, enquanto o *token* está sendo usado por um computador, nenhum outro pode transmitir.

Depois que o computador captura o *token*, envia um *frame* de dados para a rede. O *frame* circula no anel até encontrar o computador com o endereço destino do *frame.* O computador de destino copia o *frame* para seu *buffer* de recepção e marca o campo de status do *frame*, indicando que a informação foi recebida. O *frame* volta ao anel, até chegar ao computador que o enviou, onde a transmissão é reconhecida como sucesso. O computador, então, remove o *frame* do anel e coloca um novo *token* no anel.

CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

Esta técnica foi baseada na rede ALOHA, no Havaí, que iniciou sua operação em 1970; consistia de uma forma de comunicação de um computador central com seus terminais espalhados pelas várias ilhas do arquipélago. Eram utilizadas duas freqüências de rádio: uma para difusão de mensagens do computador central para os terminais e outra para mensagens dos terminais para o computador. Uma vez que existia apenas um transmissor no primeiro canal, nenhuma dificuldade era encontrada. O problema aparecia no segundo canal, onde todos os terminais transmitiam em uma mesma freqüência. Esta situação é a mesma encontrada em uma rede com topologia em barramento.

Uma rede com topologia em barramento apresenta as seguintes características: canal de transmissão eletricamente aberto, transmissão por difusão (nos dois sentidos) e estações de acesso passivas. Uma vez que o barramento é um canal de transmissão aberto, não existe a necessidade de se prover mecanismos para a retirada das mensagens transmitidas, pois o esvaziamento do meio é feito naturalmente. Isto dificulta a implantação de um serviço de resposta automática (se comparado com a topologia em anel), mas as características inerentes a esta topologia permitem a implantação de técnicas de controle de acesso com funções totalmente distribuídas.

Um sistema de difusão é aquele no qual uma mensagem transmitida no meio é ouvida, mais ou menos simultaneamente, por todos os nós de comunicação ligados à rede. Neste método, uma estação só transmite sua mensagem após escutar o meio de transmissão e determinar que o mesmo não está sendo utilizado. Qualquer nó de comunicação pode ouvir qualquer mensagem transmitida, não importando se o destinatário é ele, outro nó, ou ambos. Caso a estação detecte o meio ocupado, deve aguardar até que o sinal desapareça para, então, iniciar a sua transmissão. Pode ocorrer que duas ou mais estações estejam aguardando que o meio fique desocupado, para iniciarem suas transmissões, o que ocasionará uma “colisão” de mensagens das estações ao serem transmitidas simultaneamente. Existe a necessidade de se estabelecer uma política de compartilhamento deste meio de transmissão.

Quando duas mensagens são transmitidas ao mesmo tempo, diz-se que elas *colidiram;* o resultado é que as informações de ambas são corrompidas ou perdidas. Os métodos de acesso para redes com topologia em barramento devem, portanto, evitar o máximo possível a ocorrência de colisões; no caso de ocorrência de colisão, as mensagens perdidas ou danificadas devem ser identificadas para posterior retransmissão. Uma vez que duas ou mais estações tenham transmitido simultaneamente, ocasionando colisão de mensagens, um esquema de prioridade deve ser providenciado para evitar que ocorram novas colisões envolvendo as mesmas estações.

A maneira de se evitar que a colisão só seja percebida quando uma resposta não for recebida, é "escutar" o meio de transmissão antes (*carrier sense*) e durante (*collision detection*) a transmissão da mensagem. Dessa forma, a estação emissora poderá identificar se existe outro sinal misturado com o seu; uma vez detectada a colisão, todas as estações envolvidas param de transmitir e tentam outra vez, após um tempo de espera.

Para evitar que as mesmas mensagens colidam novamente, o tempo que cada estação aguarda para retransmitir é calculado de forma aleatória, o que não evita que a mensagem desta estação colida com a mensagem de uma outra. Neste caso, o procedimento se repete, mas o tempo de espera para uma nova tentativa é maior que o anterior, penalizando a estação que colide muitas vezes. Após um número específico de tentativas sem sucesso, a estação envia uma mensagem ao seu usuário indicando a impossibilidade de efetuar o serviço solicitado. À medida que a carga da rede cresce, cada equipamento tem a sua taxa de transmissão reduzida. Dessa forma, com a redução total da carga da rede, as transmissões vão se ajustando gradativamente.

Este método de acesso foi um dos escolhidos como padrão, e é de fato o método mais difundido em redes locais. O resultado é que o CSMA/CD tem sido escolhido para a maioria dos projetos de redes locais. O principal exemplo de utilização do CSMA/CD é dado pela rede Ethernet [41].

A seguir, na figura 9, ilustramos a colisão em redes de banda básica.

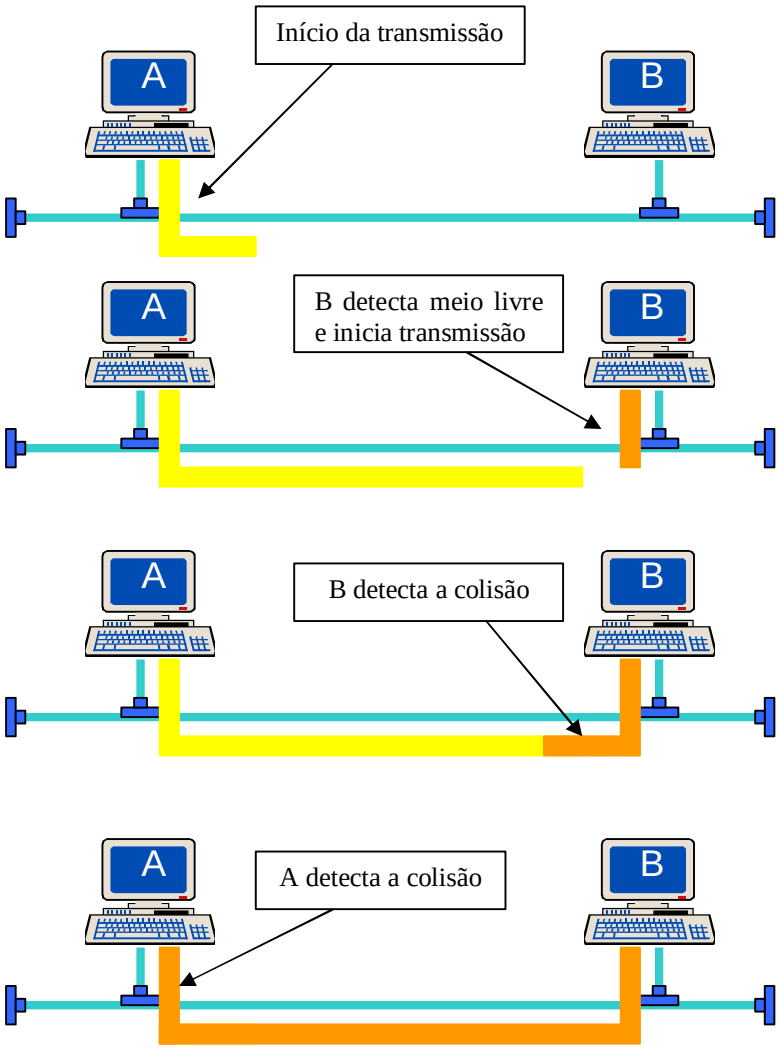


FIGURA 9 - COLISÃO EM REDES EM BANDA BÁSICA

c) Subcamada de Controle de Enlace Lógico

A subcamada de Controle de Enlace Lógico (LLC) é a camada da arquitetura IEEE 802 que se encarrega de prover às camadas superiores os serviços que permitam uma comunicação confiável de seqüência de bits (quadros) entre os sistemas usuários da rede. A especificação da subcamada LLC prevê a existência de três tipos de serviços básicos, fornecidos à camada superior.

Um primeiro serviço oferecido pela subcamada LLC permite que as unidades de informação sejam trocadas sem o estabelecimento prévio de uma conexão em nível de enlace de dados. Neste tipo de serviço não há, portanto, nem controle para recuperação de erros ou anomalias, nem controle da cadência de transferência das unidades de dados (controle de fluxo). Supõe-se que as camadas superiores possuam tais mecanismos, de modo que tornem desnecessária sua duplicação nas camadas inferiores.

Um segundo serviço fornecido pela subcamada LLC consiste no estabelecimento de uma conexão em nível de enlace de dados, antes da fase de troca de dados propriamente dita, de modo que incorpore as funções de recuperação de erros, de seqüênciamento e de controle de fluxo.

O terceiro refere-se a um serviço sem conexão com reconhecimento utilizado em aplicações que necessitam de segurança, mas não acomodam o *"overhead"* de estabelecimento de conexão.

2.6.3 Arquitetura TCP/IP

A arquitetura TCP/IP é largamente utilizada para interconexão e interoperação de sistemas computacionais heterogêneos. Tal arquitetura foi lançada pelo Departamento de Defesa do governo americano e escolhida para ser o padrão obrigatório de comunicação entre os diversos sistemas daquela organização. Os padrões não são definidos por entidades de padronização internacional, como a ISO, por exemplo. As definições dos protocolos são encontradas em documentos denominados RFC (*Request for Comments*), os quais são elaborados pelo IAB (*Internet Activities Board)* [42].

Esta arquitetura TCP/IP dá uma ênfase toda especial à interligação de diferentes tecnologias de redes. A única forma de permitir que um grande volume de usuários possa trocar informações é interligar as redes às quais eles estão conectados, formando assim uma *inter-rede*.

Para interligar duas redes distintas, é necessário conectar uma máquina a ambas as redes. Tal máquina fica responsável pela tarefa de transferir mensagens de uma rede para a outra. Uma máquina que conecta duas ou mais redes é denominada *internet gateway* ou *internet router.* Para ser capaz de rotear corretamente as mensagens, os gateways precisam conhecer a topologia da inter-rede, ou seja, precisam saber como as diversas redes estão interconectadas.

A arquitetura Internet TCP/IP é organizada em quatro camadas conceituais construídas sobre uma quinta camada que não faz parte do modelo, a camada intra-rede. Ela é composta por dois protocolos principais: o IP (*Internet Protocol*) e o TCP (*Transmission Control Protocol*).

A figura 10 adiante, ilustra a estrutura em camadas da arquitetura Internet.

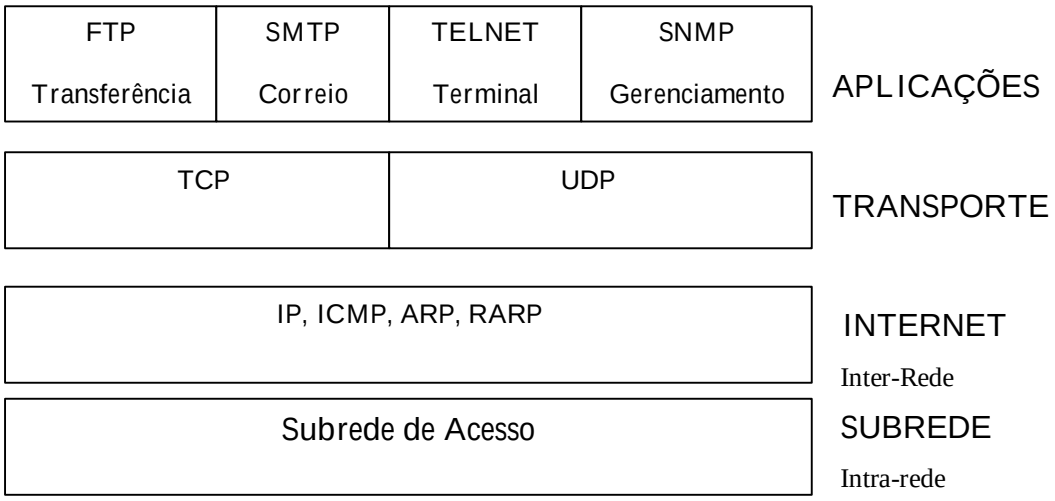


FIGURA 10 - CAMADAS DA ARQUITETURA INTERNET

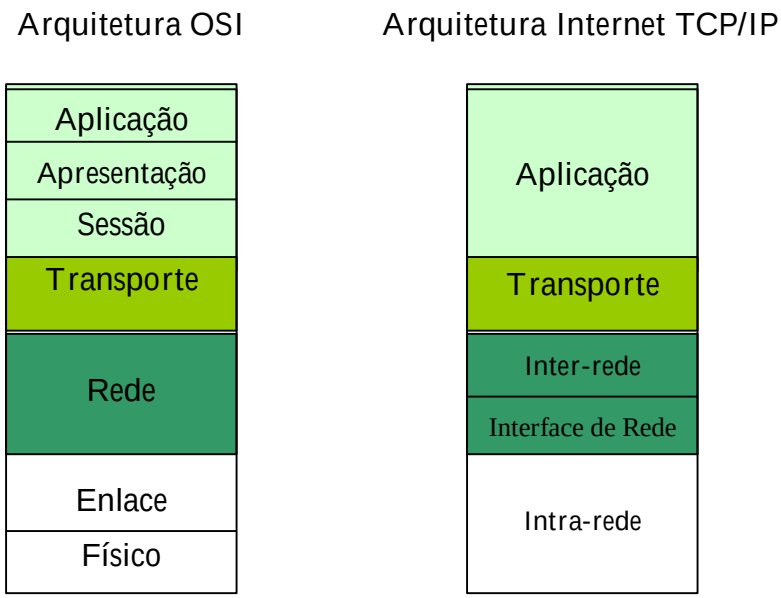
No *nível de aplicação,* os usuários usam programas de aplicação para acessar os serviços disponíveis na inter-rede. As aplicações interagem com o nível de transporte, para enviar e receber dados. As aplicações podem usar o serviço orientado à conexão, fornecido pelo TCP (serviço de circuito virtual), ou o serviço não-orientado à conexão, fornecido pelo User Datagram Protocol – UDP (serviço de datagrama não confiável). Algumas aplicações disponíveis na Internet TCP/IP estão citadas na figura 10.

A função básica do *nível de transporte* é permitir a comunicação fim-a-fim entre aplicações. As funções do nível de transporte na arquitetura Internet TCP/IP são semelhantes às do mesmo nível do RM-OSI.

O nível inter-rede é o responsável pela transferência de dados através da rede, desde a máquina de origem até a máquina de destino. Esse nível recebe pedidos do nível de transporte para transmitir pacotes que, ao solicitar a transmissão, informa o endereço da máquina onde o pacote deverá ser entregue. O pacote é encapsulado em um datagrama IP, e o algoritmo de roteamento é executado para determinar se o datagrama IP pode ser entregue diretamente ou se deve ser repassado para um gateway.

A arquitetura Internet TCP/IP não faz nenhuma restrição às redes que são interligadas para formar a rede de comunicação. Portanto, qualquer tipo de rede pode ser ligado, bastando, para isso, que seja desenvolvida uma interface que compatibilize a tecnologia específica da rede com o protocolo IP [8].

FIGURA 11 - COMPARAÇÃO ENTRE AS ARQUITETURAS OSI E INTERNET TCP/IP



Como pode ser observado na figura 11, a primeira diferença entre as arquiteturas OSI e Internet TCP/IP está no número de camadas. Enquanto na arquitetura OSI são definidas sete camadas, na arquitetura Internet TCP/IP são definidas cinco.

No RM-OSI são descritos formalmente os serviços de cada camada: a interface usada pelas camadas adjacentes para troca de informações e o protocolo que define regras de comunicação para cada uma das camadas. Alguns dos serviços definidos para as camadas do RM-OSI são opcionais. Essa flexibilidade tem aspectos positivos, mas, por outro lado, pode levar a situações em que dois sistemas em conformidade com a arquitetura OSI não consigam se comunicar, bastando para tal que implementem perfis funcionais incompatíveis.

A arquitetura TCP/IP foi desenvolvida com o objetivo de resolver um problema prático: interligar redes com tecnologias distintas. Para tal, foi desenvolvido um conjunto específico de protocolos que resolveu o problema de forma bastante simples e satisfatória.

Os serviços do nível de rede OSI, relativos à interconexão de redes distintas, são implementados na arquitetura Internet TCP/IP pelo protocolo IP. Em outras palavras, nessa arquitetura só existe uma opção de protocolo e serviço para esta subcamada do nível de rede: o protocolo IP. Esta inflexibilidade da arquitetura Internet TCP/IP no nível inter-rede é uma das principais razões de seu sucesso. O fato de um sistema utilizar ou não o protocolo IP foi usado, inclusive, para distinguir os sistemas que “estão na Internet” dos que não estão.

Os protocolos da arquitetura Internet TCP/IP oferecem uma solução simples, porém bastante funcional, para o problema da interconexão de sistemas abertos. O fato de implementações de seus protocolos terem sido a primeira opção de solução não-proprietária para a interconexão de sistemas fez com que essa arquitetura se tornasse um padrão de fato [8].

2.7 A Rede Local Ethernet

O padrão Ethernet original foi desenvolvido por um comitê composto de representantes de três grandes empresas: Digital Equipment Corporation, Intel e Xerox. Publicado em 1980, esse padrão começou a ser conhecido com padrão DIX Ethernet (para abranger as iniciais das três empresas). Uma revisão do padrão foi posteriormente publicada em 1985, conhecida como Ethernet II. Esse documento foi então passado para o IEEE para ampla padronização na indústria. O documento resultante, ratificado em 1985, foi oficialmente intitulado IEEE 802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Acess Method and Physical Layer Specifications". Isso poderia esclarecer por que a maioria das pessoas na indústria menteve o nome Ethernet, a despeito de praticamente todos os equipamentos vendidos atualmente obedecerem, na realidade, ao padrão 802.3.

Em muitos aspectos, o padrão 802.3 é um superconjunto do padrão DIX Ethernet. Embora o padrão original especifique apenas o uso de cabo coaxial grosso Ethernet e o Ethernet II acrescente o cabo coaxial fino, o padrão 802.3 adiciona a capacidade de usar outros tipos de cabos, como o par trançado não-blindado (unshielded twisted pair – UTP) e a fibra óptica, que praticamente ofuscaram o uso do Ethernet cabo grosso, como a forma mais comum nas redes.

O IEEE 802.3 é o padrão para redes em barra, utilizando o CSMA/CD como método de acesso. O padrão provê a especificação necessária para redes em banda básica, operando em 1 e 10 Mbps, e para redes em banda larga, operando em 10 Mbps.

Ao tratar de redes em banda básica a 10 Mbps, o padrão IEEE 802.3 converge para a especificação da rede Ethernet.

A rede local Ethernet foi otimizada para troca de dados a altas velocidades entre equipamentos processadores de informação dentro de áreas geográficas de tamanho

moderado, permitindo maximizar as comunicações entre uma grande variedade de equipamentos oriundos de diversos fabricantes.

As características básicas da Ethernet são:

velocidade de transmissão – a Ethernet é considerada como uma rede local que apresenta alta velocidade de transmissão de sinais de dados, devido à combinação de mecanismo de controle de acesso CSMA/CD (proporcionando acesso distribuído a todos as estações), sinalização em banda base e cabo coaxial ou par trançado de alto desempenho. Podem-se conectar várias estações na Ethernet, permitindo uma taxa de transferência de até 1 Gbps, sendo que todos os pontos apresentam oportunidades iguais de enviar seus pacotes, desde que o canal esteja livre. Uma das principais características se traduz na difusão das mensagens no meio físico de transmissão, dando oportunidade para todas as estações ouvirem as mensagens.

desempenho – a Ethernet não apresenta nenhuma instabilidade quando submetida a cargas pesadas. Mesmo nas horas de maior movimento, o mecanismo de acesso CSMA/CD proporciona capacidade plena distribuída às estações.

qualidade – existem certas implementações que controlam o funcionamento da rede, tornando-a mais segura:

- falhas no cabo de transmissão podem ser localizadas através de um temporizador que é acionado quando um pacote é transmitido, e o tempo de cada colisão é anotado. Se o defeito for no cabo, o tempo de colisão será constante, sendo, assim, possível localizar-se o ponto que apresenta defeito através de uma relação tempo – velocidade, associada a fórmulas específicas;
- as estações e os componentes defeituosos podem ser prevenidos para se autodesligarem da rede, no caso de suas transmissões ultrapassarem o tamanho permitido pela Ethernet;
- tem-se uma controladora monitorando continuamente os aspectos operacionais da rede. Por exemplo, após cada transmissão o circuito sensor de colisão é testado.

custo – Os equipamentos da rede Ethernet apresentam um custo relativamente baixo em comparação com o que a rede pode proporcionar: compatibilidade de recursos; flexibilidade de instalação em vários tipos de lugares quer seja num edifício, num campus, num complexo industrial etc; grande capacidade de incorporação de novas estações no sistema sem, com isso, perturbar o funcionamento da rede; e simplicidade em sua configuração e na conexão [41].

Nível Físico

O padrão IEEE 802.3 trata dos componentes físicos usados para interligar estações (Data Terminal Equipment – DTE) através de uma rede local Ethernet, definindo várias opções de meio físico e taxa de transmissão. Dentre as opções especificadas, abordaremos o 10BASE2 e o 10BASET, os quais fazem parte do nosso objeto de estudo.

Especificação 10BASE2

A especificação 10BASE2 foi elaborada com o intuito de prover um meio simples, barato e flexível de ligar dispositivos ao meio físico de transmissão de uma rede local de computadores. A interconexão das estações é implementada com o uso de cabos coaxiais finos (thinnet), conectores BNC e terminadores BNC.

O meio de transmissão especificado no padrão 10BASE2 é o cabo coaxial fino, que é mais flexível e fácil de manipular. O comprimento máximo do cabo é de 185 metros. A taxa de transmissão é de 10 Mbps, usando sinalização digital.

A especificação 10BASE2 aplica o esquema de detecção de colisão e permite que o comprimento da rede seja estendido com a utilização de repetidores. A distância máxima entre duas estações da rede deve, no entanto, ser limitada pela especificação do tamanho mínimo da mensagem [8].

Especificação 10BASE-T

A especificação 10BASE-T também fornece um meio simples, barato e flexível de ligar dispositivos ao meio físico de transmissão.

O meio de transmissão definido no 10BASE-T é o par trançado (*Twisted-Pair*), sendo este o motivo do "T" no título da especificação; com o uso de conectores RJ-45, suporta uma taxa de transmissão de 10 Mbps, em distâncias de até 100 metros, podendo ser aumentada, dependendo da qualidade do par trançado. A técnica de transmissão utilizada é a sinalização em banda básica.

Esta especificação define enlaces ponto-a-ponto full-duplex, utilizando dois pares trançados, um para transmissão e o outro para recepção de dados. A construção de redes com mais de duas estações requer o uso de repetidores multiporta (hubs/switches) para interligar dois ou mais enlaces [8].

No próximo capítulo apresentamos segurança em redes e criptografia.

CAPÍTULO 3

SEGURANÇA EM REDES E CRIPTOGRAFIA

Neste capítulo, abordaremos conceitos básicos sobre segurança em redes de computadores, bem como criptografia, que é o objeto principal da nossa dissertação, tendo em vista a utilização de técnicas criptográficas como um dos instrumentos de segurança em redes.

3.1 Segurança em Redes

O termo segurança é usado com o significado de minimizar a vulnerabilidade de bens (qualquer coisa de valor) e recursos. Vulnerabilidade é qualquer fraqueza que pode ser explorada para se violar um sistema ou as informações que ele contém.

A segurança está relacionada à necessidade de proteção contra o acesso ou a manipulação, intencional ou não, de informações confidenciais por elementos não autorizados e a utilização não autorizada do computador ou de seus dispositivos periféricos. [9]

No Brasil a cultura no assunto é ainda muito recente, é raríssima a preocupação com segurança, coisa que já mudou nos Estados Unidos, onde alguns desastres já ocorreram, como o caso de Morris que escreveu o "*worm*", ou verme [12]. Mesmo a criação do CERT(*Computer Emergency Response Team*, ou Equipe de Resposta a

Emergência de Computadores) não foi suficiente. Em 1990 um grupo de jovens alemães de Hamburgo do grupo “CHAOS” foi preso após invadir computadores militares americanos e vender informações obtidas para a KGB. Um dos jovens foi encontrado morto dias depois, em um estranho “suicídio”. No mesmo ano, toda a rede telefônica de longa distância da AT&T nos Estados Unidos ficou desligada durante doze horas, levando à posterior prisão de vários jovens e condenação de três deles, pertencentes ao grupo Legion of Doom. Novas medidas de segurança foram incorporadas aos sistemas operacionais e falhas foram corrigidas, mas, no caso talvez mais famoso, em 1994 Kevin Mitnick, utilizando uma falha do protocolo TCP, invadiu o sistema do especialista em segurança Tsotomu Shimomura e roubou diversos *softwares*. Após dois meses de caçada Mitnick foi finalmente preso, em uma operação que envolveu a NSA e o FBI. Se até um famoso especialista em segurança teve seu computador invadido, que grau de proteção pode um usuário comum esperar ter em seu computador pessoal? Todo esse histórico gerou a crença que os sistemas ligados à Internet são inerentemente inseguros e ligar-se a uma rede é uma garantia de que seus dados mais cedo ou mais tarde serão inevitavelmente roubados.

Mas esta crença é falsa: um sistema Internet ou uma rede de computadores podem ser seguros, sem que seja preciso ser um “guru” em segurança para assim torná-lo. A necessidade de proteção deve ser definida em termos das possíveis ameaças e riscos e dos objetivos de uma organização, formalizados nos termos de uma política de segurança, a qual abrange desde uma fechadura na porta da sala de computadores até o uso de técnicas criptográficas sofisticadas e códigos de acesso[12].

3.1.1 Ameaças e Ataques

Uma ameaça consiste em uma possível violação da segurança de um sistema. Algumas das principais ameaças às redes de computadores são:

- a destruição de informação ou de outros recursos;
- a modificação da informação;
- roubo, remoção ou perda de informação ou de outros recursos;

§ a revelação de informação;

§ a interrupção de serviços.

As ameaças podem ser classificadas como acidentais ou intencionais, podendo ambas ser ativas ou passivas.

Ameaças acidentais são as que não estão associadas à intenção premeditada.

Exemplos:

- Descuidos operacionais.
- *Bugs* de *software* e *hardware*.

Ameaças intencionais são as que estão associadas à intenção premeditada.

Exemplos:

- Observação de dados com ferramentas simples de monitoramento das redes.
- Alteração de dados baseadas no conhecimento do sistema.

Ameaças passivas são as que, quando realizadas, não resultam em qualquer modificação nas informações contidas em um sistema.

As *ameaças ativas* envolvem alterações de informações contidas no sistema ou modificações em seu estado ou operação.

A materialização de uma ameaça intencional configura um ataque. Alguns dos principais ataques que podem ocorrer em um ambiente de processamento e comunicação de dados são os seguintes:

Personificação (masquerade)

Uma entidade faz-se passar por outra. Uma entidade que possui poucos privilégios pode fingir ser outra, para obter privilégios extras.

Replay

Uma mensagem, ou parte dela, é interceptada e, posteriormente, transmitida para produzir um efeito não autorizado.

Modificação

O conteúdo de uma mensagem é alterado, implicando em efeitos não autorizados, sem que o sistema consiga detectar a alteração.

Por exemplo: alteração da mensagem “Aumentar o salário de José para R$300,00” para “Aumentar o salário de José para R$ 3000,00”

Recusa ou Impedimento de Serviço

Ocorre quando uma entidade não executa sua função apropriadamente ou atua de forma que impeça que outras entidades executem suas funções.

Por exemplo: Geração de mensagens com o intuito de atrapalhar o funcionamento de algoritmos de roteamento.

Ataques internos

Ocorrem quando usuários legítimos se comportam de modo não autorizado ou não esperado.

Armadilhas (Trapdoor)

Ocorrem quando uma entidade do sistema é alterada para produzir efeitos não autorizados em resposta a um comando (emitido pela entidade que está atacando o sistema), a um evento ou a uma seqüência de eventos, premeditados.

Por exemplo: a modificação do processo de autenticação de usuários para fornecer a senha, em resposta a uma combinação de teclas específicas.

Cavalos de Tróia

Código não autorizado contido dentro de um programa legítimo. Esse código não autorizado realiza funções desconhecidas (e provavelmente indesejáveis tais como roubar senhas ou copiar arquivos) para o usuário.

Por exemplo: um login modificado, que, ao iniciar a sua sessão, grava as senhas em um arquivo desprotegido.

Ataques baseados em senhas (Cracker de senha)

Um *cracker* de senha é qualquer programa que supera a segurança de senha revelando senhas que anteriormente foram criptografadas. Entretanto, isso não significa que um *cracker* de senha possa necessariamente decifrar qualquer coisa. De fato, a grande maioria dos *crackers* de senha não consegue fazer isso.

Em geral, não se pode decifrar senhas que estejam criptografadas com algoritmos fortes. Vários dos processos modernos de criptografia são tais que não existe nenhum processo para inverter a cifragem (ou pelo menos não em um período razoável de tempo).

Muitos *crackers* de senha não são nada além de sistemas de força bruta, programas que tentam uma palavra depois da outra, freqüentemente em velocidades muito altas. Esses programas se baseiam na hipótese de que mais cedo ou mais tarde se encontrará a palavra ou frase certa. Essa hipótese é factível porque os humanos são, em parte, criaturas descuidadas. Eles raramente atentam para o problema de criar senhas fortes.

Ataques que exploram o acesso confiável

Muitos sistemas operacionais como o UNIX e WINDOWS NT têm mecanismos de acesso confiável, projetados para facilitar o acesso a outros sistemas e domínios. Com relação aos sistemas conectados à Internet, os sistemas UNIX são muito

mais explorados do que qualquer outra plataforma. Os sistemas UNIX possibilitam o uso de arquivos de *host* confiáveis formados por nomes de *hosts* ou endereços a partir dos quais um usuário pode obter acesso sem utilizar uma senha, apenas executando um comando rlogin, ou outro semelhante.

A maioria dos ambientes empresariais proíbe ou restringem o acesso confiável a *hosts* em redes conectados à Internet. Contrapondo-se a isto, o alto custo da administração do sistema em ambientes com conectividade limitada e o valor comercial relativamente baixo dos dados e operações computacionais justificam o uso do acesso confiável a *hosts*.

Spoofing do IP

O *spoofing* interfere na forma como um cliente e um servidor estabelecem uma conexão, e envolve o fornecimento de falsas informações sobre uma pessoa ou sobre a identidade de um *host* para obter acesso não autorizado a sistemas.

O *spoofing* do IP é o mais conhecido dentre todos os ataques de *spoofing*.

Os ataques de *spoofing* exigem a previsão de números de seqüência desconhecidos e permitem apenas uma conexão em uma só via com uma rede. Portanto os intrusos podem enviar mensagens para uma rede, mas não podem recebê-las.

Seqüestro de Sessão

O seqüestro (ou roubo) de sessão é semelhante ao *spoofing* do IP; algumas pessoas o consideram como um tipo especial de *spoofing* do IP.

No seqüestro de sessão, um intruso procura por uma conexão já existente entre dois *hosts* e tenta ter o controle sobre ela. Depois que o intruso obtém o controle da máquina, ele monitora a conexão que está sendo efetuada, determinando os números de seqüência utilizados por ambos os lados da conexão.

Após ver a conexão, o intruso pode gerar um tráfego que parece vir de um dos *hosts*, simplesmente “roubando” a sessão de uma das duas pessoas envolvidas no processo. Ao fazê-lo, o intruso obtém os mesmos privilégios de acesso que o usuário legítimo. Este, então, é descartado da conexão e o intruso pode continuar o que o usuário havia começado.

A proteção contra seqüestro de sessão é extremamente difícil, até mesmo mecanismos de autenticação mais rígidos nem sempre têm êxito ao impedir ataques de seqüestro. A única real defesa contra esse ataque é o intenso uso de criptografia.

Ingerência na rede/rastreamento de pacote

Uma rede de meios físicos compartilhados é uma rede na qual os pacotes são transmitidos de todas as partes da rede à medida que trafegam dos pontos de origem para os pontos de destino.

Nas redes de meios físicos compartilhados os pacotes podem ser interceptados em qualquer ponto dessas redes, exigindo medidas especiais de controle. A captura de pacotes dessa forma é conhecida como rastreamento da rede (ou rastreamento de pacote ou monitoração promíscua).

Como a Internet é uma rede de meios físicos compartilhados, ela é vulnerável a esse tipo de ingerência.

Esse método de ataque é útil para que intrusos não apenas capturem informações de *login*, mas também obtenham ilegalmente dados e mensagens eletrônicas. E a proteção contra atos de ingerência na rede geralmente é muito difícil.

Sniffers

Sniffers são dispositivos que capturam pacotes de rede. Seu propósito legítimo é analisar tráfego de rede e identificar áreas potenciais de interesse/preocupação. Por exemplo, suponha que um segmento de sua rede esteja sendo executado precariamente: a

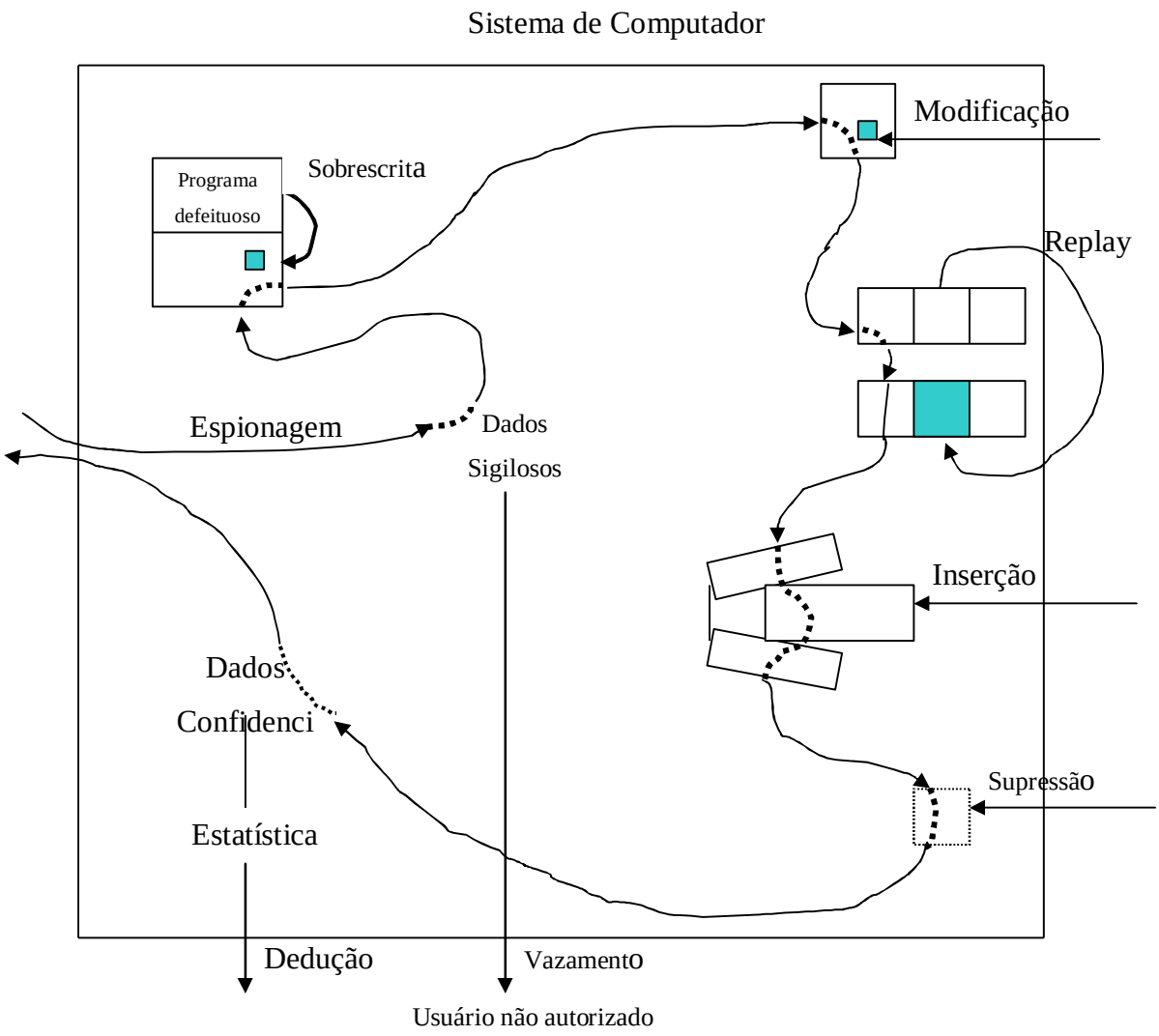
entrega de pacote parece incrivelmente lenta ou as máquinas inexplicavelmente bloqueiam em uma iniciação de rede. Utiliza-se um *sniffer* para determinar a causa precisa.

Em contrapartida os *sniffers* podem representar um alto risco à segurança, pois eles podem capturar senhas e informações confidenciais de proprietário, além de poderem abrir brechas na segurança de redes vizinhas ou ganhar acessos de alto nível.

A existência de um *sniffer* não autorizado na rede pode indicar que seu sistema já está comprometido [12].

A seguir apresentamos na figura 12, um diagrama das possíveis ameaças aos dados armazenados em sistemas de computadores.

FIGURA 12 - AMEAÇAS AOS DADOS ARMAZENADOS EM SISTEMAS DE COMPUTADOR



3.1.2 Política de Segurança

Uma política de segurança é um conjunto de leis, regras e práticas que regula como uma organização gerencia, protege e distribui suas informações e recursos.

Uma política de segurança deve incluir regras detalhadas, definindo como as informações e os recursos da organização devem ser manipulados; deve definir, também, o que é e o que não é permitido, em termos de segurança, durante a operação de um dado sistema.

Existem dois tipos de política:

- *Política baseada em regras*: as regras deste tipo de política utilizam os rótulos dos recursos e dos processos para determinar o tipo de acesso que pode ser efetuado. No caso de uma rede de computadores, os dispositivos que implementam os canais de comunicação, quando é permitido transmitir dados nesses canais.

- *Política baseada em segurança*: o objetivo deste tipo de política é permitir a implementação de um esquema de controle de acesso que possibilite especificar o que cada indivíduo pode ler, modificar ou usar.

3.1.3 Educação em Segurança

Determinar o quanto precisamos nos preocupar com segurança, vai depender da posição ocupada na empresa. A privacidade é um interesse. Todo usuário deve estar ciente que a comunicação não criptografada através da Internet é totalmente insegura. Mesmo assim, cada usuário deve estar ciente de que os órgãos de governo rotineiramente falham em sua missão quanto à segurança. Embora a Internet seja um recurso maravilhoso

para pesquisa ou recreação, existem riscos (pelo menos, não se tiver qualquer coisa a ocultar).

O pessoal administrativo às vezes é rápido em recusar (ou restringir) financiamento para segurança dentro da sua corporação. Tratam esse custo como desnecessário, principalmente porque não entendem a terrível natureza da alternativa. A realidade é esta: um ou mais *crackers* com talento podem – em minutos ou horas – destruir vários anos de acumulo de dados.

Um certo nível de segurança aceitável deve ser alcançado antes de algum negócio na Internet poder ser confiavelmente conduzido. A educação é uma maneira econômica para as empresas alcançarem segurança pelo menos mínima. O que eles gastam agora pode vir a economizar muitas horas no futuro.

3.1.4 Serviços de Segurança

Os serviços de segurança em uma rede de computadores têm como função:

- *a confidencialidade* – consiste em proteger os dados contra leitura por pessoas não autorizadas;
- *a integridade dos dados* – consiste em evitar que pessoas não autorizadas insiram, excluam ou modifiquem mensagens;
- *a autenticação das partes envolvidas* – consiste em verificar o transmissor de cada mensagem e tornar possível aos usuários o envio de documentos eletronicamente assinados.

3.1.5 Mecanismos de Segurança

Uma política de segurança e seus serviços podem ser implementados por meio de vários mecanismos de segurança, entre eles:

Segurança Física e de Pessoal

Procedimentos operacionais devem ser definidos para delinear responsabilidades do pessoal que interage com um dado sistema. A segurança de qualquer sistema depende, em última instância, da segurança física dos seus recursos e do grau de confiança do pessoal que opera o sistema. Ou seja, não adianta utilizar mecanismos sofisticados de segurança se os intrusos puderem acessar fisicamente os recursos do sistema.

Controle de Acesso

Os mecanismos de controle de acesso são usados para garantir que o acesso a um recurso seja limitado aos usuários devidamente autorizados. As técnicas utilizadas incluem a utilização de listas ou matrizes de controles de acesso, que associam recursos a usuários autorizados, ou *passwords*, *capabilities* e *tokens* associados aos recursos, cuja posse determina os direitos de acesso do usuário que as possui.

Integridade de dados

Para garantir a integridade dos dados, podem ser usadas técnicas de detecção de modificação, normalmente associadas com a detecção de erros em bits, em blocos, ou erros de seqüência introduzidos por enlaces e redes de comunicação. Entretanto, se os cabeçalhos e fechos carregando informações de controle não forem protegidos contra modificações, um intruso que conheça as técnicas pode contornar a verificação.

Enchimento de Tráfego

A geração do tráfego espúrio e o enchimento das unidades de dados fazendo com que elas apresentem um comprimento constante são formas para fornecer proteção contra análise do tráfego. Cabe ressaltar que o mecanismo de enchimento de tráfego só tem sentido caso as unidades de dados sejam criptografadas, impedindo que o tráfego espúrio seja distinguido do tráfego real.

Controle de Roteamento

A possibilidade de controlar o roteamento, especificando rotas preferenciais (ou obrigatórias) para a transferência de dados, pode ser utilizada para garantir que os dados sejam transmitidos em rotas fisicamente seguras ou para garantir que a informação sensível seja transportada em rotas cujos canais de comunicação forneçam os níveis apropriados de proteção.

Registro de Eventos

O registro de eventos possibilita a detecção e investigação de possíveis violações da segurança de um sistema, além de tornar possível a realização de auditorias de segurança.

A auditoria de segurança envolve duas tarefas: o registro dos eventos no arquivo de auditoria de segurança (*security audit log*) e a análise das informações armazenadas nesse arquivo para a geração de relatórios.

Firewalls

Firewalls são mecanismos muito utilizados para aumentar a segurança de redes ligadas à Internet, espécies de barreira de proteção, constituídas de um conjunto de *hardware* e *software*.

Firewall é um sistema ou um grupo de sistemas que garante uma política de controle de acesso entre duas redes (normalmente a Internet e uma rede local). Em princípio *firewalls* podem ser vistos como um par de mecanismos: um que existe para bloquear o tráfego e outro que existe para permitir o tráfego. Alguns *firewalls* dão maior ênfase ao bloqueio de tráfego, enquanto outros enfatizam a permissão do tráfego, o importante é configurar o *firewall* de acordo com a política de segurança da organização que o utiliza, estabelecendo o tipo de acesso que deve ser permitido ou negado [12].

Microsoft Proxy Server (Utilizado na Polícia Militar de Pernambuco)

O Microsoft Proxy Server, é um servidor extensível de *firewall* e *cache* da Web, que fornece segurança de várias camadas na Internet, enquanto melhora o tempo de resposta e a eficiência da rede. Ele atua como um *gateway* com a segurança de classe de *firewall* entre uma LAN (Local Area Network, rede local) e a Internet (figura 13).

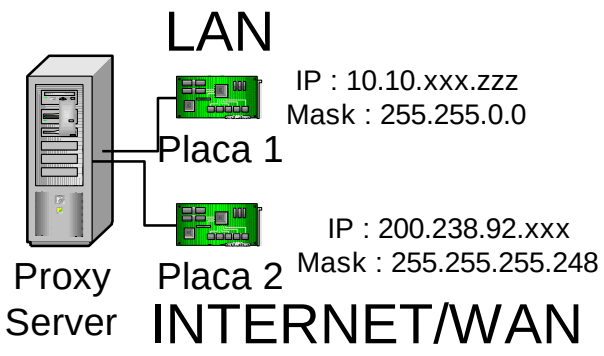


FIGURA 13 – PROXY SERVER : UTILIZADO COMO FIREWALL NA PMPE

Criptografia

Criptografia é um dos mecanismos de segurança, que será abordado no próximo item, e representa um tema central desta dissertação.

3.2 Criptografia

Segurança é uma propriedade complexa, difícil de modelar e otimizar. Na verdade, trata-se de um conceito que se baseia em quanto se está protegido de um possível oponente. Projetar um sistema de segurança significa analisar o potencial do possível adversário e desenvolver uma estratégia que consiga neutralizar seus "ataques". Segurança de dados é a disciplina que estuda os métodos de proteção de dados em sistemas de comunicação, armazenamento e transmissão. Dentre os tipos de proteção utilizados em segurança de dados, encontra-se a *Criptografia.*

A criptografia surgiu da necessidade de se enviar informações sensíveis por meios de comunicação não confiáveis, ou seja, meios em que não é possível garantir que um intruso não irá interceptar o fluxo de dados para leitura (intruso passivo) ou para modificá-lo (intruso ativo). A forma encontrada baseia-se na utilização de um método para modificar o texto original de uma mensagem a ser transmitida (texto claro), gerando um texto criptografado na origem, por meio de um processo de cifragem definido por um método de criptografia, garantindo, assim, a confidencialidade e a autenticidade da informação transmitida.

Os objetivos da criptografia, que são essencialmente *sigilo e autenticidade*, não são fáceis de serem distinguidos. De fato, estes dois objetivos são completamente independentes. Xuejia Lai, pesquisador radicado na Suíça deu, talvez, a melhor regra para distinguir entre *sigilo e autenticidade*. Uma técnica provê *sigilo* se ela determina quem pode receber a mensagem; ela provê autenticidade se ela determina quem pode ter enviado a mensagem [13].

A criptografia está intimamente relacionada com a segurança e assume um papel cada vez mais importante, devido à grande quantidade de informações transmitidas pela Corporação e à utilização crescente de redes de computadores, podendo ser usada para cifrar dados e mensagens antes que sejam armazenados ou enviados por via de

comunicação, para que, mesmo que sejam interceptados por pessoas não-autorizadas, dificilmente possam ser decifrados corretamente.

3.2.1 Terminologias

Criptografia (Kriptós = escondido, oculto; grápho = grafia) é a arte ou ciência de escrever em cifra ou em código, de forma que se permita normalmente que apenas um destinatário autorizado a decifre e compreenda. Quase sempre a decifragem requer uma chave, uma informação secreta disponível ao destinatário.

Criptoanálise (kriptós; análisis = decomposição) é a arte ou ciência de determinar a chave ou decifrar mensagens sem conhecer a chave. Uma tentativa de criptoanálise é chamada ataque.

Criptologia (kriptós; logo = estudo, ciência) é a ciência que reúne a criptografia e a criptoanálise.

A criptografia computacional é usada para garantir:

- o sigilo: somente os usuários autorizados têm acesso à informação.
- a integridade da informação: garantia oferecida ao usuário de que a informação correta, original, não foi alterada, nem intencionalmente, nem acidentalmente.
- a autenticação do usuário: é o processo que permite ao sistema verificar se a pessoa com quem está se comunicando é, de fato, a pessoa que alega ser.
- a autenticação de remetente: é o processo que permite a um usuário certificar-se de que a mensagem recebida foi de fato enviada pelo remetente, podendo, inclusive, provar, perante um juiz, que o remetente enviou aquela mensagem.
- a autenticação do destinatário: consiste em se ter uma prova de que a mensagem enviada foi como tal recebida pelo destinatário.

- a autenticação de atualidade: consiste em provar que a mensagem é atual, não se tratando de mensagem antiga reenviada.

Cifrar é o ato de transformar dados em alguma forma ininteligível reversível. Seu propósito é o de garantir a privacidade, mantendo a informação escondida de qualquer pessoa não autorizada, mesmo que esta consiga visualizar os dados criptografados.

Decifrar é o processo inverso, que consiste em transformar em inteligíveis os dados criptografados na sua forma original.

Para cifrarmos ou decifrarmos uma mensagem, necessitamos de informações confidenciais geralmente denominadas de *chave*. Dependendo do método de criptografia empregado, a mesma chave pode ser utilizada tanto para cifrar como para decifrar mensagens, enquanto outros mecanismos utilizam chaves distintas para cifragem e decifragem [16].

A criptografia, hoje em dia, é bem mais que somente misturar e desembaralhar informações. Os processos criptográficos atuais nos fornecem mecanismos para implementarmos autenticidade e sigilo. Esses mecanismos podem ser usados para controlar, por exemplo, acessos a discos rígidos compartilhados ou controlar canais de TV pagos por tempo de uso. As aplicações para o campo da criptografia são muito amplas. Com algumas ferramentas básicas, é possível elaborar esquemas e protocolos que nos permitam o uso do “dinheiro eletrônico”, ou provar que se tem acesso a certa informação sem a necessidade de revelá-la, ou, então, dividir um contexto sigiloso de modo que menos de 3, de um grupo de 5 pessoas, por exemplo, possam reconstruí-lo.

3.2.2 Relato Histórico

Desde os tempos mais remotos, servimo-nos de códigos secretos para tornar incompreensível uma mensagem a quem não fosse autorizado a lê-la. É bastante vasto o de

que se tem conhecimento a respeito dos códigos e das cifras, datando aproximadamente de 4.000 anos atrás, do tempo da grande civilização Egípcia, em cujas antigas tumbas são numerosíssimos os exemplos de escrita cifrada: a ela atribuía-se um valor mágico e religioso, além de prático.

É provável que a arte de se esconder o verdadeiro sentido das comunicações escritas date de épocas ainda mais remotas. Os códigos secretos sempre tiveram uma posição de destaque em algumas técnicas criptográficas.

Na Antigüidade, a criptologia era praticada, entre os gregos, mais como uma arte do que mesmo como ciência [15].

Júlio César escrevia a Cícero e a outros amigos, na época da Roma Antiga, há aproximadamente 2.000 anos atrás, empregando técnicas de cifragem extremamente simples para a época atual. Esta técnica recebeu o nome de *Cifra de César,* em que cada letra era substituída pela que a seguia três posições adiante no círculo do alfabeto latino. Esta cifra é classificada como Substituição Simples [15].

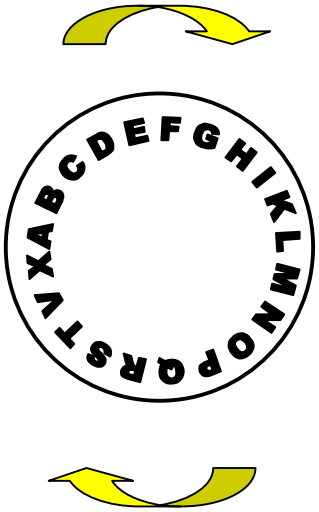
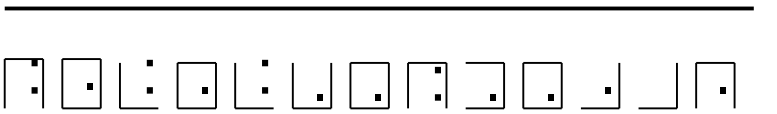


FIGURA 14 - O CIFRADOR DE CÉSAR: CADA LETRA É SUBSTITUÍDA POR AQUELA QUE A SEGUE TRÊS POSIÇÕES ADIANTE, NO CÍRCULO. SEGUNDO O TESTEMUNHO DO ESCRITOR LATINO SUETÔNIO, O IMPERADOR AUGUSTO SE LIMITAVA, POR SUA VEZ, A AVANÇAR APENAS UMA POSIÇÃO.

Em 1794, em Nova York, foi gravada uma inscrição cifrada numa tumba, nos fundos da Igreja de Trinity, na qual não se utilizou um alfabeto convencional [16]. A chave da cifra é dada por um diagrama, como apresentado na figura 15. Uma cifra similar também foi encontrada em uma tumba na Igreja de St. Paul, em Nova York, em 1796. A primeira solução publicada para estas cifras apareceu 100 anos depois, no New York Herald, em 1896 [18].



Texto cifrado

A .	B .	C .	K :	L :	M :	T	U	V
D .	E .	F .	N :	O :	P :	W	X	Y
G .	H .	I-J .	Q :	R :	S :	Z		

FIGURA 15 - CIFRA ENCONTRADA NA TUMBA DOS FUNDOS DA IGREJA DE TRINITY

Mensagens também foram codificadas em símbolos musicais [19]. Um método comum era a Substituição Simples de notas individuais por letras. Vários compositores usaram tais esquemas para codificar nomes de pessoas em seus trabalhos; então, elaboravam temas em volta de cifras. Bach, por exemplo, incorporou seu próprio nome no “Musical Offering” e no “Art of the Fugue”.

Cifras de Substituição Simples são geralmente fáceis de quebrar através do ataque somente ao texto cifrado, utilizando a distribuição de freqüência relativa das letras. Cifras de Substituição Homofônicas mapeiam cada caractere do alfabeto do texto claro em um conjunto de elementos cifrados que são chamados de homofonemas [23] e são imunes a ataques de freqüência relativa.

Por vários séculos, criptoanalistas amadores tentaram solucionar uma cifra deixada por Thomas Jefferson Beale, por volta de 1820, a qual continha indicações para se

chegar a um tesouro enterrado na Virgínia. Esta cifra foi a primeira de três cifras deixada por Beale. A segunda cifra foi solucionada por James Ward [20], por volta do ano de 1880, e descreve o alegado tesouro e a direção para encontrá-lo. A terceira cifra é ainda um enigma. Existem membros da Associação da Cifra de Beale que ainda tentam encontrar o tesouro ou simplesmente solucionar a cifra [21, 22].

O desenvolvimento das cifras polialfabéticas, as quais utilizam múltiplas substituições, iniciou com Leon Battista Alberti, em 1568, por intermédio da publicação em que descrevia um disco de cifra que definia várias substituições múltiplas [23].

Ainda por volta do século 16, atribuiu-se a um criptologista francês, Blaise de Vigenère, uma cifra que se baseia na substituição dos caracteres de um alfabeto por outros do alfabeto deslocado [23].

A Cifra de Playfair, homenagem ao cientista inglês Lyon Playfair, tendo sido inventada em 1854 por um amigo deste, Charles Wheatstone, e utilizada pelos ingleses durante a Primeira Guerra Mundial, corresponde a uma cifra de substituição poligrâmica [23].

Em 1917, Gilbert Vernam, funcionário da American Telephone and Telegraph Company, implementou um dispositivo criptográfico para as comunicações telefônicas baseadas no código Baudot, de 32 caracteres. Cada caractere é representado como uma combinação de 5 marcas e espaços, correspondentes aos bits 1 e 0, respectivamente, nos computadores digitais. Esta cifra é similar à de Vigenère. A grande idéia de Vernam foi a introdução de uma cifra chamada “one-time pad”, a qual consiste na utilização de uma chave com seqüência aleatória de caracteres, utilizada apenas uma vez. Cada dígito da chave (escolhido aleatoriamente) é usado para cifrar um dígito da mensagem, possibilitando assim, a segurança máxima contra ataques criptoanalíticos ao texto cifrado [23].

Durante a Segunda Guerra Mundial, surgiram as máquinas a Rotor, de Hagelin, que definem Cifras de Substituição Polialfabética consistindo em um banco de rotores ou discos ligados por um fio. Foram inventadas por Boris Hagelin, entre 1920 e 1930. Embora

gerassem fluxo de chaves longas, elas não eram aleatórias e tornavam-se vulneráveis ao criptoanalista. A máquina Enigma, por exemplo, inventada por Arthur Scherbius e utilizada pelos alemães, usava um odômetro a rotor. O primeiro rotor avançava para a próxima posição depois que cada caractere era cifrado. Depois que este fazia uma rotação completa, o segundo rotor avançava para a próxima posição. Similarmente, depois que o segundo rotor fazia a rotação completa, o terceiro avançava e, assim, todos os rotores se moviam em todas as posições [23].

Foi justamente por volta da Segunda Guerra Mundial que a comunidade científica reconheceu que os matemáticos poderiam prestar contribuições à Criptologia. Então, em 1949, com a publicação do trabalho de Shannon, “Communication Theory of Secrecy Systems” [24], introduziu-se a era científica da criptografia da chave secreta. Shannon mediu a segurança teórica de uma cifra pela incerteza do texto claro, dado um texto cifrado recebido. Uma cifra é incondicionalmente segura quando não existe informação suficiente no texto cifrado para determinar um único texto claro [24].

Em 1976, a criptografia tomou um novo rumo com a publicação do trabalho de Diffie e Hellman [25]. Deu-se o início da nova era dos sistemas criptográficos de chave pública, levando à divisão da criptografia em duas fases bem distintas: clássica ou convencional e moderna ou de chave pública.

A partir daí, em 1977, ocorreu um desenvolvimento importante, quando o National Bureau of Standards (NBS) anunciou o Data Encryption Standard (DES), para ser utilizado em aplicações não-reservadas do governo dos Estados Unidos. O DES é uma técnica de criptografia convencional que cifra blocos de dados de 64 bits com uma chave de cifragem de 56 bits [26].

Ainda em 1978, Pohlig e Hellman publicaram um esquema de cifragem que se baseia no cálculo de exponenciais em um corpo finito [27]. Nessa mesma época, Rivest, Shamir e Adleman também publicaram um esquema similar que ficou conhecido no mundo científico como RSA [28].

O desenvolvimento tecnológico, com a introdução dos computadores pessoais, criou uma variedade enorme de aplicações para as comunicações eletrônicas, atendendo aos mais variados tipos de serviços. Conseqüentemente, houve um aumento da quantidade e uma maior variedade de informações possíveis de interceptação, além da oferta de equipamentos mais eficientes ao criptoanalista. Por outro lado, os mesmos fatores reduziram o custo e aumentaram a eficiência dos equipamentos de criptografia. Desse modo, cada vez mais se empregam sistemas de criptografia para garantir a segurança dos dados.

Antigamente, a criptografia era restrita aos meios militares e diplomáticos, devido ao alto custo de equipamentos confiáveis e a pouca difusão das comunicações eletrônicas. Hoje, no entanto, vários fatores se combinam para estimular o interesse em aplicações privadas e comerciais.

3.2.3 Sistemas Criptográficos

Criptografia, como vimos, é a ciência que estuda a escrita secreta. Uma cifra é um método de escrita secreta, em que um texto claro é transformado em texto cifrado. O processo de transformar texto claro em texto cifrado é chamado cifragem; o processo inverso de transformação do texto cifrado em texto claro é chamado decifragem. Ambas, cifragem e decifragem, são controladas por chaves criptográficas.

Os sistemas de criptografia podem ser classificados em sistemas de criptografia clássica e sistemas de criptografia moderna. O emprego de um ou de outro, ou de ambos, depende do projeto do sistema de segurança.

Diz-se que um sistema é incondicionalmente seguro quando é computacionalmente inviável de se solucionar. Todavia, em todo sistema proposto, é preciso que se considerem os recursos computacionais do provável oponente e o tempo

durante o qual se deseja resguardar uma informação. A busca por códigos inquebráveis tem sido um dos temas mais abrangentes na pesquisa criptográfica.

Os sistemas criptográficos podem ser divididos em duas categorias: sistema de cifragem bit a bit e sistemas de cifragem de bloco.

Os Sistemas de cifragem bit a bit (*stream ciphers*) processam o texto claro caractere a caractere, produzindo adicionalmente uma seqüência de bits peseudo-aleatória que é adicionada módulo 2 aos bits do texto claro. A mensagem M é segmentada em sucessivos caracteres m_1, m_2, .., cifrando-se cada m_i com o i-ésimo elemento k_i de uma chave $k = k_1, k_2, ...$, isto é,

$$E_k (M) = E_{k1} (m_1)\ E_{k2}\ (m_2) ...$$

Sistemas de cifragem de bloco (*block ciphers*) atuam em grandes blocos de texto claro de forma que uma mudança pequena em um bloco de entrada produza uma grande mudança no bloco de saída resultante [17]. Neste caso, a mensagem é segmentada em blocos sucessivos M_1, M_2, ... , cifrando-se cada M_i com a mesma chave K, isto é,

$$E_k (M) = E_k (M_1)\ E_k (M_2) ...$$

Diffie e Hellman observaram que se os erros são propagados pelo algoritmo de decifragem, aplicando-se códigos detectores de erros antes da cifragem (e depois da decifragem), este mecanismo fornece autenticidade, porque modificações no texto cifrado são detectadas pelo decodificador de erros [29]. Esta técnica pode ser empregada para as cifras de blocos que propagam erros. Cifras bit a bit, por sua vez, não propagam erros, pois cada caractere do texto cifrado é independentemente cifrado e decifrado. Códigos corretores de erro são normalmente aplicados após a cifragem com o fim de proteger as informações contra os efeitos do ruído no canal. [16].

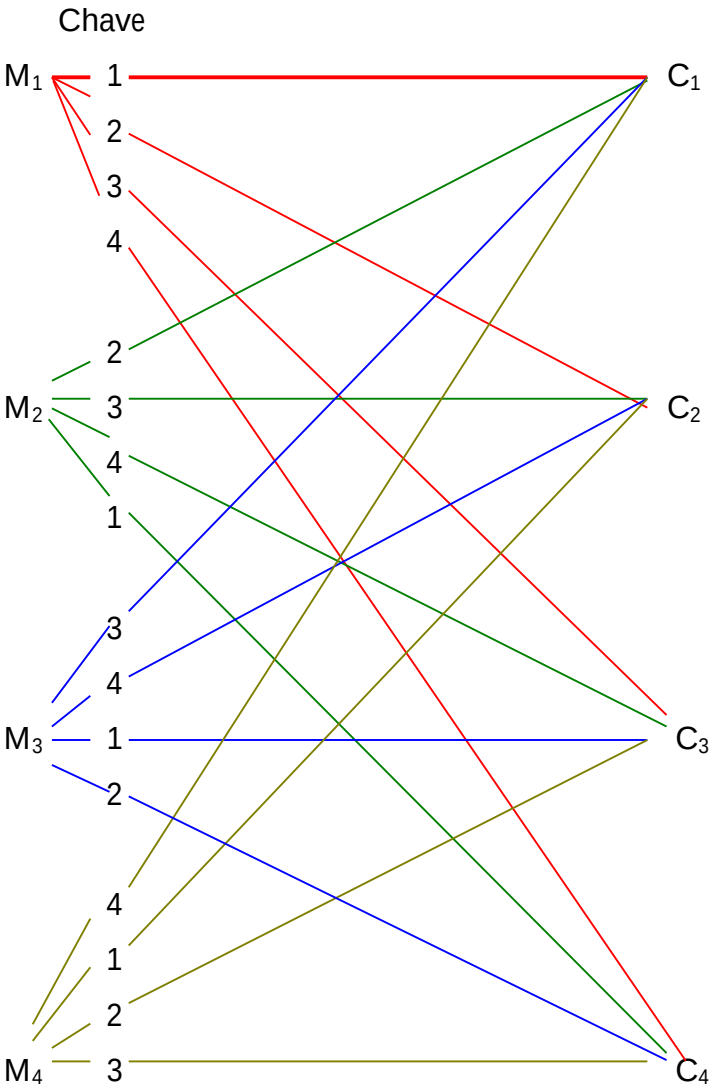
Sigilo Perfeito

A definição de Sigilo Perfeito é usada para caracterizar a situação em que a probabilidade de quebra de chave é inviável, independentemente dos recursos computacionais e do tempo do oponente para descobrir a mensagem. Não importa a quantidade de texto cifrado interceptado pelo criptoanalista, pois este não terá nenhuma informação adicional capaz de revelar o texto claro original. Segurança contra este oponente (inimigo), sem restrições computacionais, é chamada segurança incondicional. Shannon usou a terminologia segurança “teórica”.

Shannon caracterizou as propriedades de teoria da informação dos sistemas criptográficos, em que a probabilidade de receber um texto cifrado C, dado que M foi enviado (cifrado sobre várias chaves), é a mesma probabilidade de receber C, dado que alguma outra mensagem M’ foi enviada (cifrada sobre uma chave diferente).

Sigilo perfeito é possível, usando-se chaves completamente aleatórias, pelo menos tão longas quanto as mensagens a serem cifradas. Na figura 16, mostramos um exemplo de um sistema de sigilo perfeito com 4 mensagens, todas igualmente prováveis, e 4 chaves, também igualmente prováveis. Um criptoanalista interceptando uma das mensagens cifradas - C_1, C_2, C_3, ou C_4 - não teria nenhum caminho para determinar qual das 4 chaves foi usada e portanto, se a mensagem correta é M_1, M_2, M_3 ou M_4.

FIGURA 16 - SIGILO PERFEITO [24]



Ainda em seu estudo sobre a segurança teórica, Shannon fez duas suposições fundamentais. A primeira dessas suposições é de que a chave secreta de cifragem seria utilizada apenas uma única vez, o que seria impraticável em muitas aplicações devido à quantidade de chaves que deveriam ser geradas ao mesmo tempo e ao tamanho da chave. A segunda suposição é de que o criptoanalista só teria acesso aos criptogramas, ficando limitado, portanto, a apenas um tipo de ataque criptoanalítico.

Portanto, a condição de sigilo perfeito requer que o número de chaves de cifragem seja pelo menos igual ao número de mensagens, que essas chaves sejam utilizadas de forma aleatória e que o comprimento delas seja maior ou igual ao da

mensagem que irá ser cifrada. A única cifra de sigilo perfeito é a “one-time pad” [28], o que torna a construção de uma cifra, teoricamente segura, extremamente complexa para certas aplicações.

A maioria dos sistemas criptográficos, usados na prática, baseia-se não na impossibilidade de serem quebrados mas sim na dificuldade de tal quebra. Segurança contra um inimigo, que tem uma certa limitação de tempo e de poder de computação disponíveis para seu ataque, é atualmente chamada de segurança computacional. Shannon usou a terminologia segurança “prática” [13].

Sigilo Imperfeito e Distância de Unicidade

A fim de determinar quando uma cifra, que não oferecia sigilo perfeito, poderia em princípio ser quebrada, Shannon, em 1949 [24], mediu a segurança da cifra em termos da função equivocação de chave, H_c (K) da chave K para um dado texto cifrado C, isto é, a quantidade de incerteza de K dado C [16].

Se H_c (K) é zero, então não existe incerteza sobre a chave, e a cifra é teoricamente quebrável. Aumentando-se o comprimento (N) do texto cifrado a incerteza sobre a chave usualmente diminui.

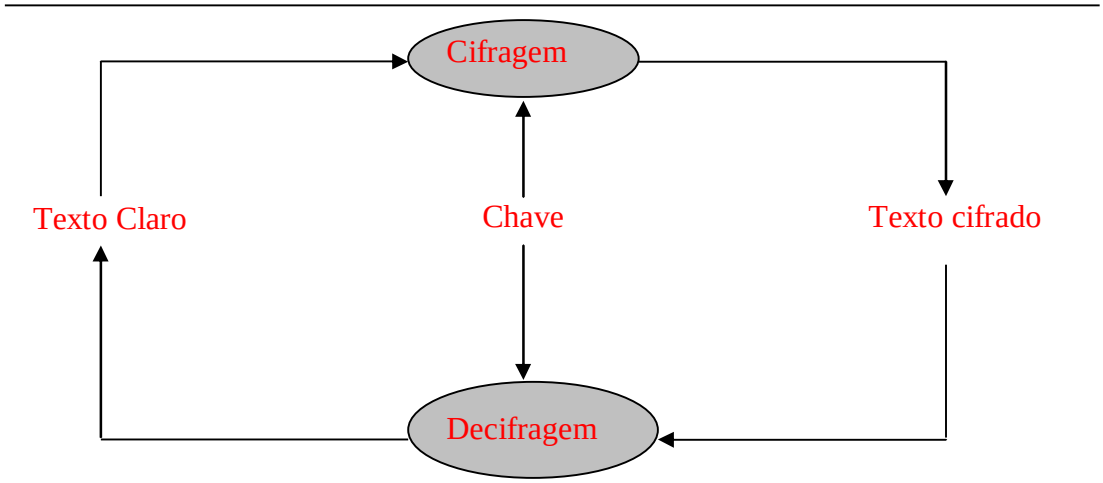
A distância de unicidade é, portanto, a menor quantidade de texto cifrado (N), tal que H_c (K) é próximo à zero, isto é, é a quantidade de texto cifrado, requerida pelo criptoanalista para determinar a chave de modo essencialmente único, em um ataque de apenas texto cifrado.

A seguir apresentamos as técnicas, os métodos e os aspectos relacionados aos sistemas criptográficos.

3.2.3.1 Sistemas de Criptografia Clássica (ou Simétricos)

Em um criptosistema simétrico, a cifragem e a decifragem são feitas com uma única chave, ou seja, tanto o transmissor quanto o receptor usam a mesma chave. O conhecimento dessa chave é que permitirá que qualquer pessoa cifre ou decifre uma determinada mensagem. O transmissor e o receptor de uma dada mensagem cifrada compartilham a mesma chave criptográfica através de um canal seguro. Para que a segurança dos dados seja preservada, essa chave compartilhada deve ser enviada por meios seguros, tais como carta registrada, mensageiro confiável, encontros pessoais etc., e, posteriormente, mantida em sigilo absoluto [30].

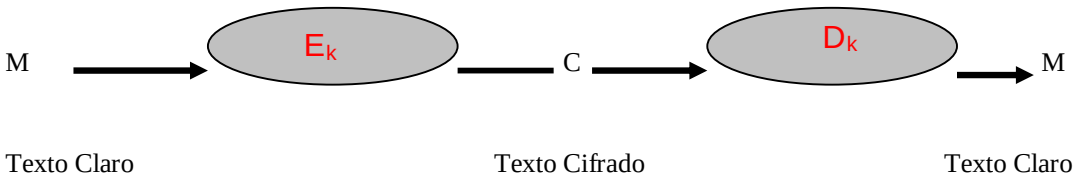
FIGURA 17 - ESCRITA SECRETA COM CRIPTOSISTEMA SIMÉTRICO



Um sistema criptográfico clássico apresenta cinco componentes:

1. Mensagem do Texto Claro, M.
2. Mensagem do Texto Cifrado, C.
3. Chave, *K*.
4. Transformação de cifragem, E_k: M $\rightarrow$C, em que k ε *K*.
5. Transformação de decifragem, D_k: C $\rightarrow$ M, em que k ε *K*.

FIGURA 18 - SISTEMA CRIPTOGRÁFICO



Um sistema de criptografia clássica é uma família de transformações inversíveis (biunívocas) $\{E_k , D_k\}$ k ε *K*, em que o índice k caracteriza uma CHAVE pertencente ao espaço de chaves *K*. As mensagens M podem ser vistas como elementos de um espaço de mensagens M e os textos cifrados C, como elementos de um espaço de criptogramas C (figura 18).

Para conseguirmos estabelecer um sistema de comunicações com privacidade e/ou autenticidade, utilizando criptografia clássica, seria necessário que a chave k, usada para cifrar e decifrar a mensagem, fosse enviada através de um canal seguro, o que geralmente não ocorre, tornando o sistema susceptível ao ataque de criptoanalistas como ilustramos na figura 19 a seguir.

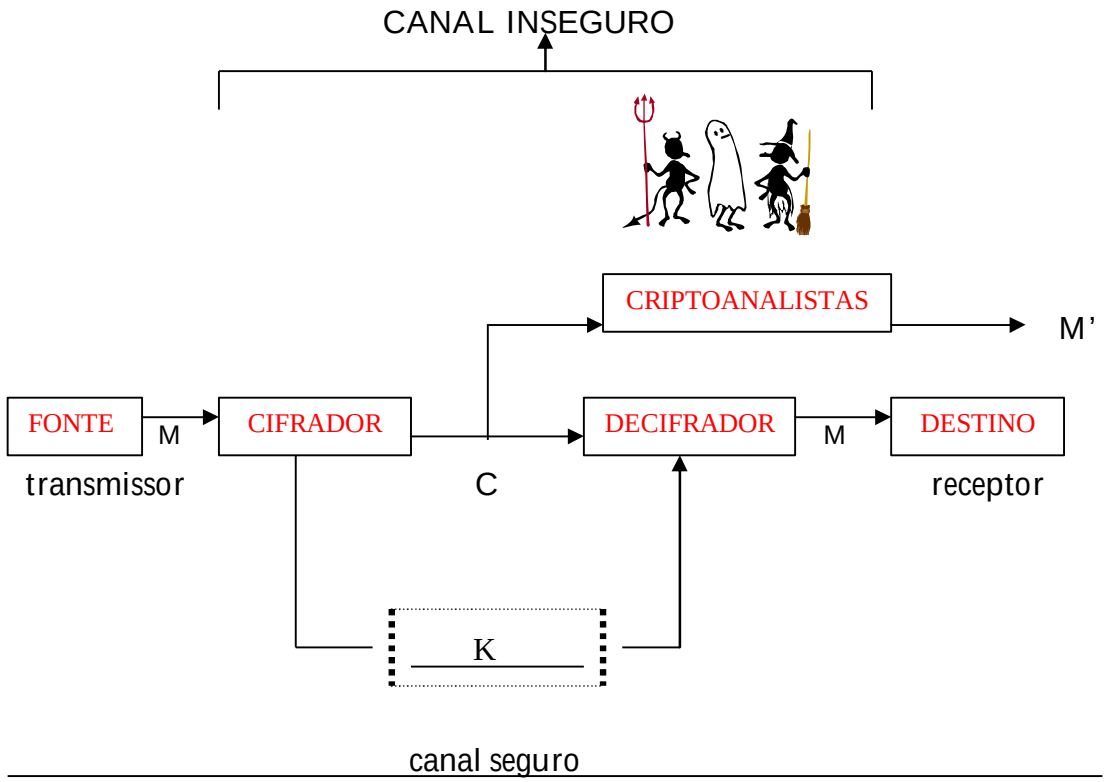


FIGURA 19 - SISTEMA CRIPTOGRÁFICO CONVENCIONAL

Observando-se o diagrama da figura 19, vêem-se três importantes elementos: o transmissor, o receptor e o criptoanalista.

O transmissor gera uma mensagem M a ser enviada, através de um canal inseguro, a um receptor. Para evitar que um indivíduo não autorizado tenha acesso à mensagem M, o transmissor aplica a M uma função matemática E, capaz de codificá-la, obtendo uma mensagem cifrada,

$$C = E_k (M)$$

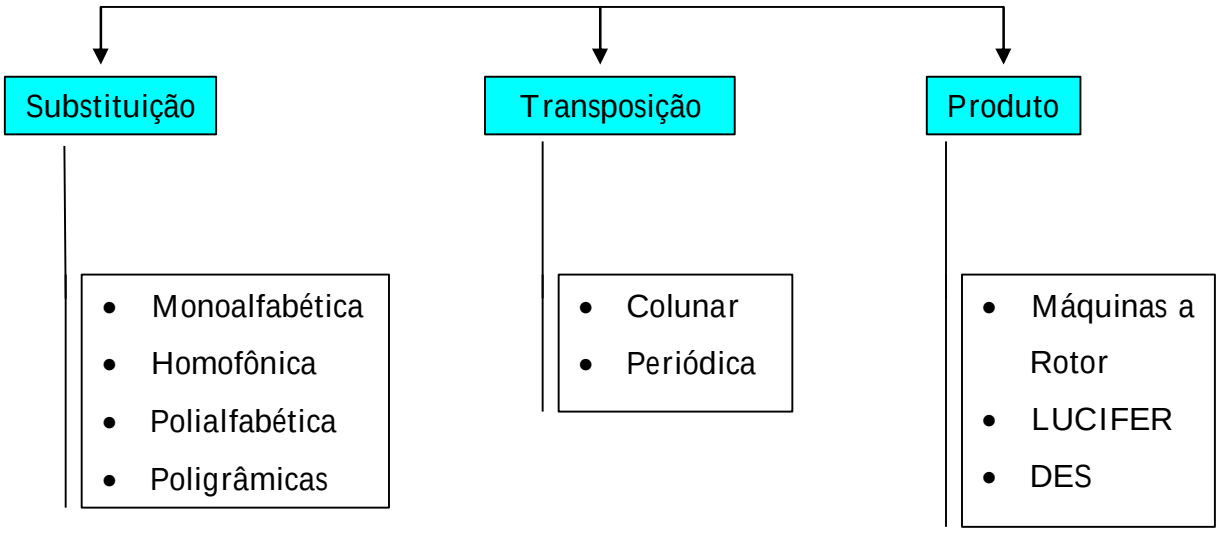
Essa função E corresponde à transformação de cifragem de uma mensagem de texto claro em um texto cifrado.

Observa-se que a chave k, compartilhada por meio de um canal seguro, é única, e, uma vez que é conhecida pelo receptor, este pode decifrar o criptograma C pela obtenção da transformada inversa, D_k , obtendo, assim, a informação original a ele enviada, isto é,

$$D_k (C) = D_k (E_k (M)) = M$$

Na criptografia clássica utilizam-se cifras nas quais se empregam duas transformações básicas, substituições e transposições. Podem-se, ainda, encontrar essas técnicas como partes integrantes de cifras mais sofisticadas, como as cifras- produto [16]. Em sua classificação geral, apresentamos o diagrama a seguir:

FIGURA 20 - TÉCNICAS CLÁSSICAS DE CIFRAGEM



a) Cifras por Substituição

Existem 4 tipos de substituição de cifras: substituição monoalfabética, substituição homofônica, substituição polialfabética e substituição poligrâmica. As Cifras por Substituição consistem na mudança de cada caractere do texto claro em um caractere correspondente do texto cifrado.

Ø *Cifras de Substituição Monoalfabética*

As cifras de Substituição Monoalfabética são aquelas que substituem cada caractere de um alfabeto A por um outro caractere de um outro alfabeto C que se apresenta em uma dada ordem, ou seja, cada caractere do alfabeto do texto claro é substituído por um único caractere do texto cifrado: o mapeamento é um para um.

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
A
CIFRA : D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
C

Então, o texto claro ABRA CADABRA é cifrado como:

M = ABRA CADABRA
E_k (M) = C = DEUD FDGDEUD

Como a substituição da letra é feita por um “alfabeto misturado”, isto aumenta um pouco a segurança, uma vez que existem 26! Chaves possíveis. Neste modo de cifragem, a freqüência relativa das letras é preservada, o que torna esta cifra vulnerável a ataques apenas por texto cifrado.

A *Cifra de César* é um exemplo da Cifra de Substituição Monoalfabética. Esta cifra expressa o alfabeto de forma cíclica. Para cada letra de texto claro, selecionamos uma letra deslocada de um número específico de posições e em uma determinada direção. O interesse por esta cifra é histórico, conforme mencionado no Relato Histórico.

A Cifra de César consiste em substituir as letras de uma mensagem pelo alfabeto deslocado por *K* posições à direita. Júlio César utilizava *K* =3. Assim, o texto criptografado pode variar de acordo com o valor de k, que é a chave utilizada nos processos de cifragem e decifragem. Como existem apenas 25 chaves possíveis, este método torna-se vulnerável a uma busca exaustiva da chave.

TABELA 1 - A CIFRA DE CÉSAR

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

Cifras de Substituição Simples são geralmente fáceis de quebrar por meio do ataque apenas ao texto cifrado, utilizando a distribuição da freqüência relativa das letras do alfabeto utilizado. A freqüência relativa de caracteres em um determinado idioma corresponde à incidência natural desses caracteres nesse idioma. Por exemplo, no idioma Português o caractere de maior freqüência relativa é o "a", enquanto que no Inglês o caractere é o "e". Cifras baseadas no deslocamento de alfabetos são quebradas, pois cada letra do texto cifrado possui uma distância constante da letra correspondente do texto claro.

- *Cifras de Substituição Homofônica*

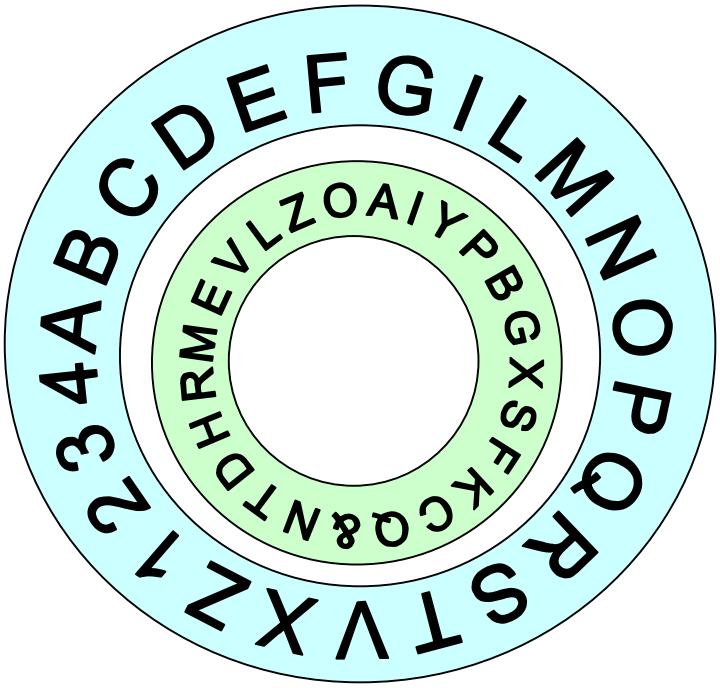
A Cifra de Substituição Homofônica é similar à de Substituição Monoalfabética, exceto que mapeia cada caractere *a* do alfabeto do texto claro em um conjunto de elementos do texto cifrado f(a), chamados de homofonemas. Uma mensagem $M = m_1\ m_2$ ... é cifrada como $C = c_1\ c_2$ ....., em que cada c_i é tomado aleatoriamente de um conjunto de homofonemas $f(m_i)$.

A Cifra de Beale é um exemplo desse tipo de técnica de cifragem, cuja chave foi a Declaração de Independência [20]. A vantagem desse tipo de cifra é que os criptogramas não preservam a estatística das letras do alfabeto original, tornando inviável uma análise por meio da freqüência relativa das letras do texto cifrado [16].

- *Cifras de Substituição Polialfabética*

A Cifra de Substituição Polialfabética esconde, também, a distribuição da freqüência relativa de letras individuais do texto claro, utilizando múltiplas substituições, ou seja, não existe um único alfabeto misturado para usar na substituição, mas um certo número deles, os quais são usados ciclicamente, para cada caractere da mensagem do texto claro. É uma substituição distinta que conduz a uma seqüência de alfabetos.

FIGURA 21 - DISCO DE CIFRAR



A Cifra de Vigenère

Uma forma popular de cifra de substituição periódica, baseada no deslocamento de alfabetos, é a Cifra de Vigenère [23]. Uma palavra chave é utilizada e determina qual alfabeto deslocado será usado para cifrar cada letra sucessiva do texto claro.

Na tabela 2 que se segue, apresentamos todos os alfabetos deslocados que aparecem na aplicação da Cifra de Vigenère.

TABELA 2 - CIFRA DE VIGENÈRE

Chave \ Texto Claro	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
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	S	R	T	U	V	W	X	Y	Z
B	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	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	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

Vamos exemplificar montando um texto cifrado com a palavra chave “LOUYSE”. A cifra da expressão ABRA CADABRA sobre a palavra chave “LOUYSE” é mostrada a seguir:

Texto Claro = M = A B R A C A D A B R A

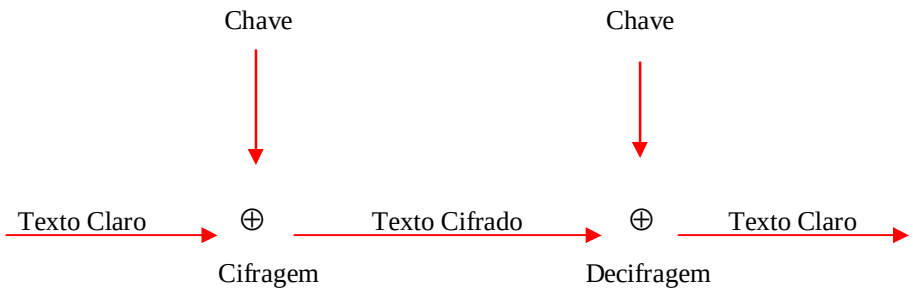
Palavra Chave = K = L O U Y S E L O U Y S

Texto Cifrado = $E_k(M)$ = L P L Y U E O O V P S

A Cifra de Vernam

Se a chave para uma cifra de substituição é uma seqüência aleatória de caracteres não repetida, então não existe informação suficiente para quebrar a cifra. Tal cifra é chamada de Cifra de Vernam ou “One-Time Pad”, pois cada letra da chave é usada apenas uma vez.

FIGURA 22 - CIFRAGEM E DECIFRAGEM UTILIZANDO A CIFRA DE VERNAM



Se uma seqüência aleatória de números for usada como chave em uma cifra de Vernam, então a cifra é totalmente inquebrável ou incondicionalmente segura, pois, para cada texto cifrado com uma chave aleatória desconhecida, todos os possíveis textos claros de mesmo comprimento são igualmente prováveis.

Ø *Cifra de Substituição Poligrâmica*

É aquela que cifra blocos de símbolos da mensagem em blocos de texto cifrado, destruindo a freqüência relativa dos símbolos do alfabeto original utilizado. Como exemplo, temos as cifras de Playfair e de Hill [23].

b) Cifras por Transposição

Cifras por Transposição rearranjam os caracteres do texto cifrado com o auxílio de alguma figura geométrica, conseguindo-se obter o texto cifrado através da maneira como se põe e se retira o texto claro dessa figura.

Em muitos casos, a figura é um arranjo bidimensional, podendo ser, contudo, n-dimensional:

1. Na transposição de colunas, o texto claro é escrito na matriz por linhas, sendo o criptograma obtido, tomando-se as colunas dessa matriz em alguma ordem. Como exemplo, considerem-se uma matriz 3 X 4 e a mensagem *M* = CRESCIMENTO; então,

	1	2	3	4
1	C	R	E	S
2	C	I	M	E
3	N	T	O	

Tomando-se as colunas segundo a ordem 2-4-1-3, tem-se o criptograma

C = RITSECCNEMO

Como se vê, para processar tanto a cifragem como a decifragem, tem-se que gerar toda a matriz.

2. Muitas Cifras de Transposição permutam os caracteres do texto claro com período fixo *d*. Seja Z_d , o conjunto dos inteiros positivos de 1 a *d*, e seja f: $Z_d \rightarrow Z_d$, a permutação sobre Z_d. Por exemplo, se a chave para cifragem é dada pelo par K = (d, f) = (4, f (i)), em que f (i): 2 4 1 3 para i = 1 2 3 4. Temos que para *M* = CRESCIMENTO,

E_k (M) = RSCEIECMTNO

Cada bloco pode ser cifrado e decifrado independentemente.

c) Cifras Produto

Esse tipo de cifra é a combinação das duas técnicas de cifragem anteriormente mencionadas, transposição e substituição, e é encontrado, por exemplo, nas conhecidas máquinas a Rotor e no DES. Podemos citar dois exemplos de cifras dessa natureza:

1. A cifra LUCIFER, projetada por Feistel [32], da IBM, que utiliza uma transformação que aplica alternadamente substituições S_i e transposições P_j, isto é,

   $$C = E_k(M) = S_t \circ P_{t-1} \circ \ldots S_2 \circ P_1 \circ S_1 (M)$$

   Em que cada S_i é uma função da chave k.

2. O algoritmo *Data Encryption Standard (DES)* é um dos principais métodos de criptografia clássica. Baseado em chave secreta, foi desenvolvido pela IBM, sendo um bloco de entrada T transposto sob uma permutação inicial IP, gerando T_o = IP (T). Após passar por 16 iterações de uma função f que combina substituição e transposição, faz-se a permutação inversa IP^{-1}, originando o resultado final.

3.2.3.2 Sistemas de Criptografia Moderna (ou Assimétricos)

Este método de criptografia baseia-se na utilização de chaves distintas: uma para cifragem (E) e outra para a decifragem (D), escolhidas de forma que a derivação de D, a partir de E, seja, em termos práticos, senão impossível, pelo menos difícil de ser realizada [16], por indivíduos não autorizados.

A criptografia de chave pública foi inventada em 1976 por Whitfield Diffie e Martin Hellman [25]. Eles propuseram um novo método de cifragem, chamado cifragem de chave-pública, em que cada usuário possui duas chaves, uma pública e outra privada, e dois usuários podem se comunicar conhecendo as respectivas chaves públicas de cada um.

Neste sistema de chave-pública, cada usuário A tem uma transformação de cifragem pública E_A, que pode ser registrada em um diretório público, e uma transformação de decifragem privada D_A, que é conhecida somente pelo usuário. A transformação privada D_A é descrita pela chave privada e a transformação pública E_A, pela chave pública. Deve ser computacionalmente inviável determinar D_A de E_A (ou mesmo encontrar uma transformação equivalente a D_A), sem o conhecimento de determinados parâmetros do sistema.

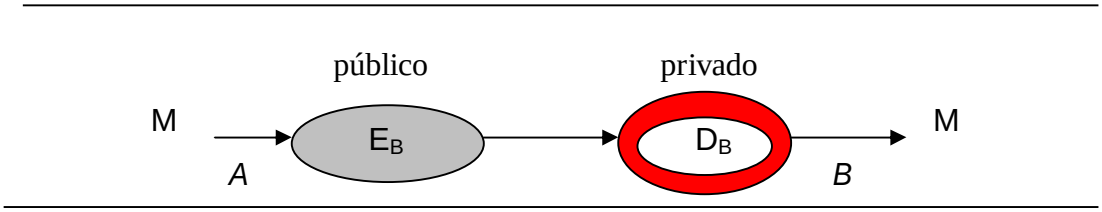
No sistema de chave-pública, são fornecidos sigilo e autenticidade por meio da separação das transformadas. Suponhamos que um usuário A deseja enviar uma mensagem M para outro usuário B. Se A conhece a transformação pública de B, E_B, então A pode transmitir M para B em sigilo,

$$C = E_B (M).$$

O receptor B decifra C usando a transformação privada D_B, obtendo, assim,

$$D_B (C) = D_B (E_B (M)) = M$$

FIGURA 23 - SIGILO NO SISTEMA DE CHAVE-PÚBLICA

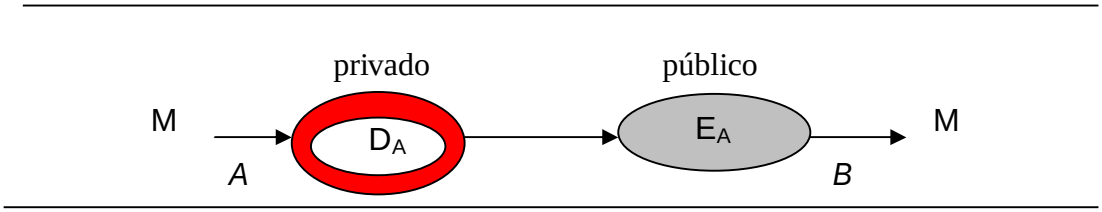


O esquema acima descrito (figura 23) não fornece autenticidade, porque qualquer usuário com acesso à transformação pública de B poderia substituir qualquer outra mensagem M' por M, através da substituição de C por C' = E_B (M').

Para obter autenticidade (figura 24), M deve ser transformado pela própria transformação privada de *A*, D_A . Ignorando o sigilo por um momento, *A* envia C = D_A (M) para B. Na recepção, B usa a transformação pública de A para computar

$$E_A (C) = E_A (D_A (M)) = M$$

FIGURA 24 - AUTENTICIDADE NO SISTEMA DE CHAVE-PÚBLICA



A autenticidade é fornecida no esquema acima porque somente *A* pode aplicar a transformação D_A . O sigilo não é fornecido, porque qualquer usuário com acesso à transformação pública de *A* pode recuperar M.

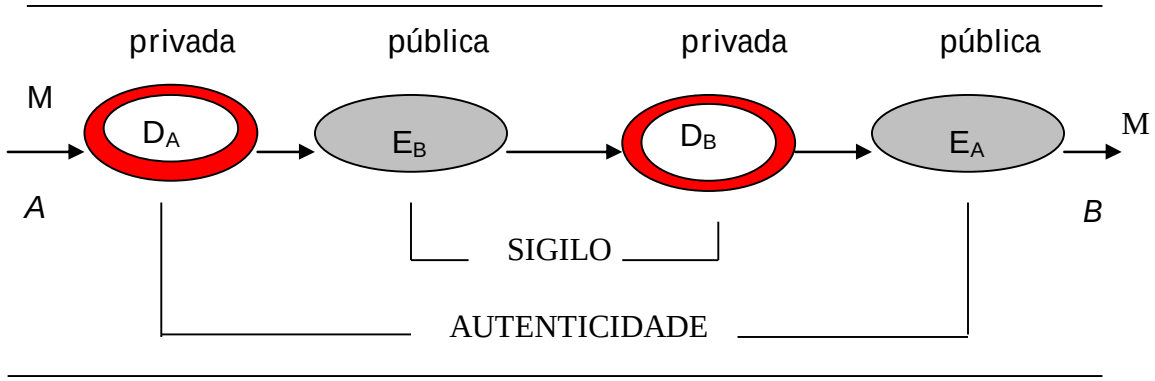
Agora, para utilizar o sistema de chave-pública com autenticidade e sigilo ao mesmo tempo (figura 25), o transmissor *A* e o receptor *B* devem aplicar dois conjuntos de transformações. Suponha-se que *A* deseja enviar uma mensagem M para *B.* Primeiramente, a transformação privada de *A*, D_A é aplicada. Então *A* cifra o resultado, utilizando, agora, a transformação de cifragem pública de *B*, E_B, e transmite a mensagem duplamente transformada

$$C = E_B (D_A (M)) \text{ para } B.$$

O usuário *B* recupera M, primeiramente aplicando a própria transformada de decifragem D_B, e, então, aplica a transformada pública de *A*, E_A para validar a sua autenticidade, obtendo

$$E_A (D_B (C)) = E_A (D_B (E_B (D_A (M))))$$
$$= E_A (D_A (M))$$
$$= M.$$

FIGURA 25 - SIGILO E AUTENTICIDADE NO SISTEMA DE CHAVE-PÚBLICA



O esquema inventado por Rivest, Shamir e Adleman [28], comumente conhecido pelo nome derivado de suas iniciais, RSA, é o mais importante método de criptografia assimétrico. O método RSA baseia-se na dificuldade de fatorar o produto de dois números primos muito grandes. Temos também o esquema baseado na dificuldade em solucionar o "problema da mochila", o qual pode ser usado para sigilo ou autenticidade, porém não ambos, mas que foi quebrado definitivamente.

3.2.4 Assinatura Digital

Nos sistemas com chave pública, qualquer pessoa pode cifrar uma mensagem, mas somente o destinatário da mensagem pode decifrá-la. Como vimos anteriormente em autenticidade e sigilo da chave-pública, invertendo-se o uso das chaves, podemos ter uma que só pode ser cifrada por uma pessoa e decifrada por qualquer um, obtendo-se assim um

efeito de personalização do documento semelhante a uma assinatura. Um sistema desse tipo é dito provedor de assinatura digital.

Assim, para personalizar uma mensagem, um determinado usuário A cifra uma mensagem utilizando sua chave secreta e a envia para o destinatário. Somente a chave pública de A permitirá a decifragem da mensagem, provando, pois, que A enviou a mensagem. A mensagem, assim, pode ser decifrada por qualquer um que tenha a chave pública de A . Para garantir autenticidade e sigilo, deve-se cifrar duas vezes a mensagem: a primeira, utilizando a própria chave secreta (para fazer a assinatura digital) e, a seguir, utilizando a chave pública do destinatário, para que somente este possa ler a mensagem [33].

Propriedades:

- a assinatura é autêntica: quando um usuário usa a chave pública de A para decifrar uma mensagem, confirma que foi A, e somente A, quem enviou a mensagem;
- a assinatura não pode ser forjada: somente A conhece sua chave secreta;
- documento assinado não pode ser alterado: se houver qualquer alteração no texto criptografado, este não poderá ser restaurado com o uso da chave pública de A;
- a assinatura não é reutilizável: a assinatura é uma função do documento e não pode ser transferida para outro documento;
- a assinatura não pode ser repudiada: o usuário B não precisa de nenhuma ajuda de A para reconhecer sua assinatura e A não pode negar ter assinado o documento.

3.2.5 Comércio Eletrônico

Atualmente uma rota de comércio crescente é representada pela Internet e está sendo explorada principalmente por empresas (e não por países) de vanguarda que estão reconhecendo e aproveitando, como questão de rara oportunidade, o potencial econômico dessas transações que de forma ímpar independem das restrições geográficas e, até qualquer posição em contrário, do arcaico registro físico em papel.

Ao invés de levar séculos para percorrer o globo, a Renascença Digital está ocorrendo agora em algum nível em todos os negócios do mundo conectado à Internet, bem como evoluindo para maiores volumes e necessidade crescente de confiabilidade nos meios de pagamento

(http://www.inf.ufrgs.br/pos/SemanaAcademica/Semana98/horch.html).

Dentro da esfera comercial, os dois processos que dependem de confiabilidade e segurança são especificamente o recebimento e o pagamento do bem ou serviço, respectivamente preocupação do comprador e do vendedor.

A tecnologia na qual se obtém confiabilidade e segurança no *e-commerce* é fornecida pela criptografia e certificação, que são, hoje, considerados elementos essenciais nas transações seguras na Internet e são sempre utilizadas em conjunto [34].

A maioria dos *sites* oferece apenas algum tipo de proteção para os dados, que não são suficientes. É preciso certificar-se de que os *sites* nos quais se efetua compras utilizam um dos dois métodos de segurança – Secure Eletronic Transaction (SET) ou Secure Socket Layer (SSL). Tanto o Netscape Navigator como o Microsoft Internet Explorer trabalham com esses padrões (no final da referência http – se existir uma letra “s”- https – significa estar conectado a um servidor seguro). O SSL utiliza criptografia para manter em segurança as informações, como números de cartões de crédito durante transações na Internet. O SET emprega assinaturas digitais para garantir que os usuários e negociantes sejam quem realmente eles dizem que são. A maior vantagem do SET é que o

número de cartão de crédito nunca é armazenado no servidor do fornecedor. Mas o SSL é um pouco mais rápido e fácil para a configuração em sites.
(http://www.solar.com.br/segu/comercio.htm)

3.2.6 Criptoanálise

Criptoanálise é a ciência que estuda métodos de quebrar cifras. Uma cifra é quebrável se é possível determinar o texto claro ou a chave a partir do texto cifrado, ou determinar a chave a partir dos pares de texto claro – texto cifrado.

Um criptosistema deve ser seguro mesmo quando os algoritmos de cifragem e de decifragem forem conhecidos. Por essa razão são usadas chaves.

Uma pessoa não autorizada que tem acesso a alguns dos elementos de um criptosistema, é denominada de atacante. Um atacante passivo somente obtém cópias dos elementos, enquanto um atacante ativo pode alterar alguns desses elementos. Existem três métodos básicos ou classes de ataques aos criptogramas em sistemas convencionais e mais um quarto método adicional, em se tratando de sistemas criptográficos modernos. Todos eles supõem que o criptoanalista possui conhecimento total sobre os métodos de cifragem e decifragem utilizados, mas não sobre as chaves.

Métodos de Ataque ao Criptosistema:

1. *Apenas texto cifrado*: o criptoanalista tem a sua disposição uma grande quantidade de mensagens cifradas, mas desconhece as originais e as chaves utilizadas. Sua tarefa é recuperar as mensagens originais (deduzir as chaves utilizadas).
2. *Texto claro conhecido*: o criptoanalista tem a sua disposição uma grande quantidade de mensagens cifradas e também as mensagens originais

equivalentes. Sua tarefa é deduzir as chaves usadas (ou um método para recuperar mensagens cifradas com a mesma chave).

3. *Texto claro escolhido:* no método anterior, o criptoanalista poderia ser capaz de fornecer somente uma grande quantidade de mensagens de uma só vez; agora, ele pode fornecer um pequeno conjunto, analisar os resultados, fornecer outro conjunto, e assim por diante. Este é o caso mais favorável ao criptoanalista. Sua tarefa é deduzir as chaves utilizadas.
4. *Texto cifrado escolhido*: o criptoanalista não só tem uma grande quantidade de mensagens e seus equivalentes cifrados, mas pode produzir uma mensagem cifrada específica para ser decifrada e obter o resultado produzido. É utilizado quando se tem uma "caixa-preta" que faz decifragem automática. Sua tarefa é deduzir chaves utilizadas.

Supõe-se que, em geral, o criptoanalista dispõe ainda de outras informações:

- método de cifragem;
- alfabeto usado;
- assunto;
- palavras prováveis.

Ao se avaliar a segurança de um criptosistema, devemos levar em conta os recursos do atacante e o fator tempo.

CAPÍTULO 4

A CIFRA SAFER

Neste capítulo, apresentaremos a técnica criptográfica, SAFER *(Secure And Fast Encryption Routine),* como sugestão para implementação na rede de computadores da Polícia Militar, bem como o pressuposto norteador dessa escolha, dando prosseguimento aos detalhes técnicos quanto às suas propriedades, seu funcionamento, suas vantagens e sua disponibilização.

Abordaremos, ainda, o sistema de gerenciamento de chaves, que foi desenvolvido na linguagem Visual Basic para aplicação na Polícia Militar.

Pressupostos Norteadores

Considerando todo o levantamento realizado por meio do estudo exploratório, verificou-se a necessidade da implementação de uma política de segurança na rede corporativa da Polícia Militar de Pernambuco, pois o trânsito de informações, nos mais diversos setores da polícia é, em grande parte, classificado como sigiloso, devido à peculiaridade da atividade policial militar.

Para resguardar essas informações ao acesso do pessoas não autorizadas, apresentamos como sugestão a utilização da técnica criptográfica denominada SAFER.

A escolha do SAFER, entre tantas outras técnicas criptográficas, foi motivada pelo fato desta cifra ter um rastro de segurança desde seu lançamento em 1993, tendo recentemente participado do “rol” das técnicas concorrentes à substituição do DES (*Data Encryption Standard*), que atualmente é padrão americano de cifragem, e por ser consignada ao domínio público.

O DES tem uma história fascinante, que começou no final dos anos 60, quando a IBM criou um algoritmo simétrico de nome Lucifer. Doutra parte, o NBS (National Bureau of Standards), órgão do governo norte-americano, vinha estudando, desde 1968, como proteger informação sensível em aplicações civis e governamentais (as militares eram, e ainda são, de competência da NSA, a poderosa National Security Agency).

A conclusão do NBS foi que era necessário padronizar um único algoritmo criptográfico. Nesse meio tempo, a NSA tomou conhecimento do Lucifer e propôs uma mudança em sua arquitetura, reduzindo o tamanho da chave para 56 bits.

A NSA, cuja missão fundamental é propor ou aprovar algoritmos para o uso do governo dos EUA e quebrar sistemas criptográficos de outras nações, apresentou uma versão do Lucifer ao NBS. Nascia, em 1977, pelas mãos do NBS e com o aval da NSA, o DES.

Desde então, o “mandato” do DES tem sido periodicamente renovado, inicialmente pelo NBS e, depois, por seu sucessor, o NIST (National Institute of Standards and Technology). Em agosto de 1993, Michael Wiener, pesquisador canadense, construiu uma máquina dedicada a quebrar senhas DES, tornando-o vulnerável.

No ano de 1998, o NIST iniciou o processo de escolha do sucessor do DES, especificando que os candidatos deveriam operar com tamanho de chave variável e blocos de 128 bits. Ficaram excluídos, automaticamente, além do DES, vários outros algoritmos que trabalham com tamanho de chave fixo.

Encerrado o prazo para registro dos candidatos (em junho de 1998), já era possível identificar alguns candidatos com ótimo "pedigree", dentre os quais o SAFER, objeto do nosso estudo e da aplicação no projeto de segurança em redes .

4.1 Proteção Criptográfica (SAFER)

SAFER K-64 (Secure And Fast Encryption Routine)

Descreveremos, a seguir, a cifra não proprietária SAFER (Secure And Fast Encryption Routine), criada pelo Prof. James Massey para a Cylink Corporation (Sunnyvale, CA, USA), em 1993. O SAFER K-64 é um algoritmo de cifragem de bloco orientado a byte. Nele, tanto o texto claro quanto o texto cifrado têm comprimento de 64 bits (8 bytes). A chave selecionada pelo usuário também tem comprimento de 64 bits (8 bytes). Uma diferença importante entre o SAFER e outras cifras de bloco é o fato da cifragem e da decifragem não diferirem apenas na reversão da construção da chave [36].

São usadas apenas operações com bytes nos processos de cifragem e de decifragem. Esta propriedade é particularmente útil em aplicações como *"smart cards"*, em que o poder de processamento disponível é muito limitado.

A fim de alcançar a segurança desejada, são explorados dois novos conceitos criptográficos. São eles:

- uso de Transformação Linear Não-Ortodoxa (Transformada Pseudo-Hadamard), para realizar difusão tanto do texto claro como da chave, sobre o texto cifrado;
- uso de Polarização Aditiva de Chaves, para eliminar *chaves fracas*.

Cifragem no SAFER K-64

SAFER K-64 é uma cifra *iterativa*, isto é, a cifragem é realizada aplicando-se 6 rodadas (valor recomendado) numa mesma transformação, e então aplicando uma Transformação de Saída. Cada iteração usa duas subchaves (8 bytes cada), derivadas da chave secreta pelo algoritmo de chaves.

A estrutura de cifragem do texto pode ser observada na figura 26.

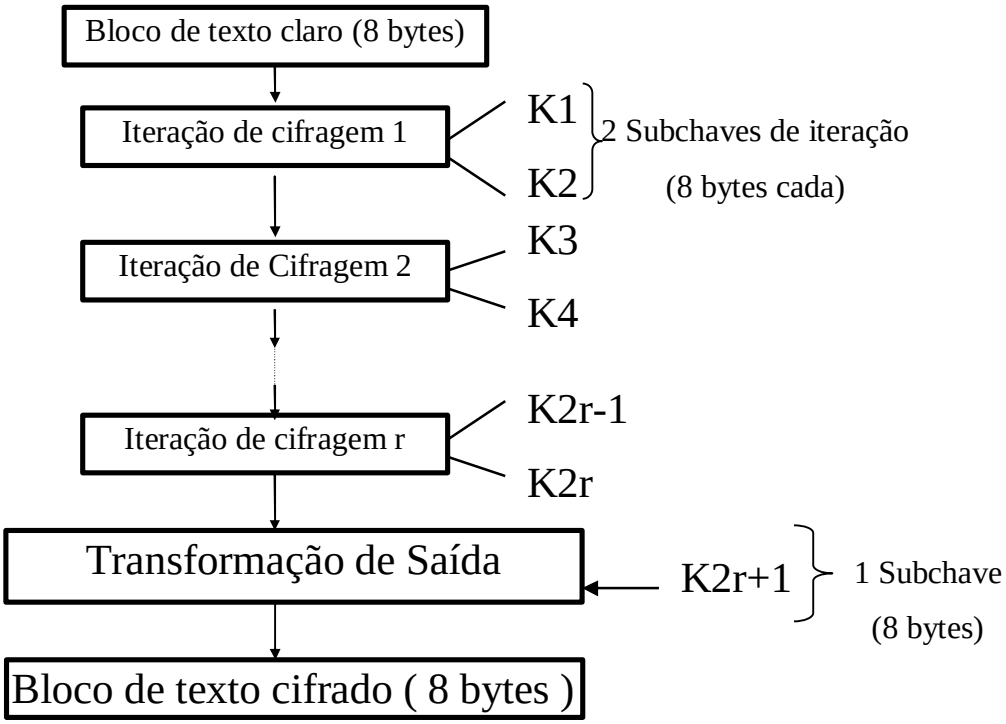


FIGURA 26 - CIFRAGEM DO SAFER K-64

A transformação de saída é formada pelas seguintes operações de ou-exclusivo (xor) e adição módulo 256 (*byte addition*) (“Ou-Exclusivo/Adição de Bytes Mista), feitas entre os bytes da última subchave (2r+1) e os respectivos bytes da saída da *r*-ésima rodada. A operação de ou-exclusivo é aplicada, bit a bit, nos bytes 1,4,5 e 8; já a operação de adição módulo 256, é aplicada, byte a byte, nos bytes 2,3,6 e7. A estrutura de cada rodada pode ser observada na figura 27.

A primeira camada da i-ésima rodada é uma camada linear na qual são feitas operações de ou-exclusivo(xor) e de adição módulo-256 (add). A operação de xor é aplicada, bit a bit, nos bytes 1,4,5 e 8; já a operação de adição módulo-256, é feita, byte a byte, nos bytes restantes (2,3,6 e 7).

A seguir, temos uma camada não linear na qual os bytes resultantes da camada anterior são sujeitos, individualmente, a transformações “altamente não lineares”, rotuladas:

- “ $45^{(.)}$” - significa que, se o byte de entrada é o inteiro j, então se obtém como saída 45^{j} módulo 257 (o resultado desta operação é tomado 0 para j = 128).
- “$\log_{45}$” - sugere que, se o byte de entrada é o inteiro j, então se obtém como byte de saída $\log_{45}(j)$ (a saída é considerada 128 se j = 0).

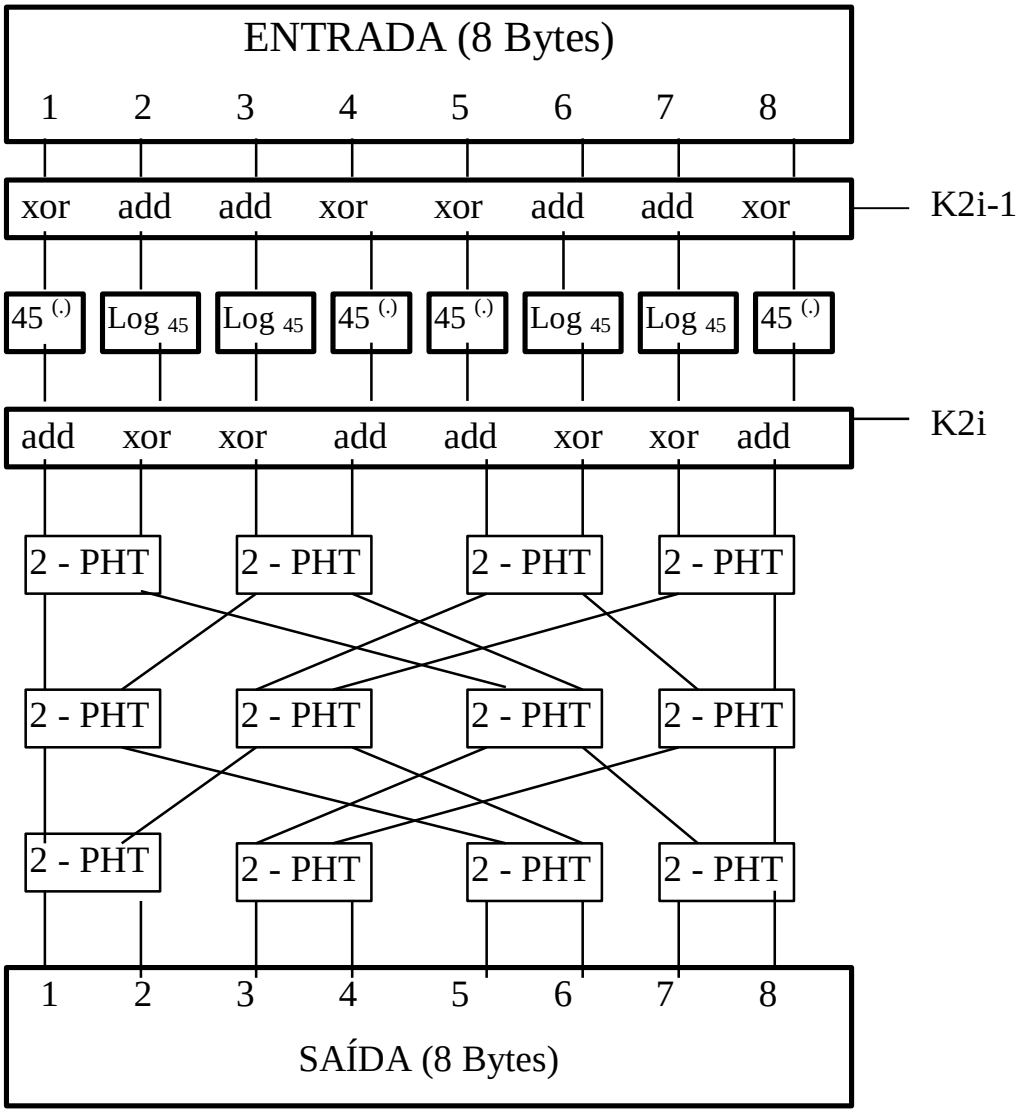


FIGURA 27 - CIFRAGEM DO SAFER K-64, ESTRUTURA ITERATIVA

A saída das oito camadas não lineares passa por uma nova camada linear semelhante à primeira. Pode-se observar pela figura que a diferença entre elas é que os bytes que sofriam a operação de xor (ou-exclusivo) agora sofrem a operação de add (adição módulo 256), e vice-versa. Essas operações são feitas entre os bytes resultantes da camada não linear com os respectivos bytes da chave K2i (Adição de Bytes Mista/Ou-exclusivo). Os bytes obtidos da segunda camada linear passam por uma nova camada linear formada por três níveis de Pseudo-Transformadas de Hadamard, com duas entradas

("2- PHT"), operando sobre um par de entradas (a_1, a_2), gerando o par de saídas (b_1, b_2), em que

$$b_1 = 2\,a_1 + a_2$$
$$b_2 = a_1 + a_2 \qquad (1)$$

com aritmética módulo 256, operando sobre bytes. A Pseudo-Transformada de Hadamard é a responsável pela difusão no SAFER K-64 [36].

Decifragem no SAFER K-64

A estrutura de decifragem do SAFER K-64 pode ser vista na figura 28.

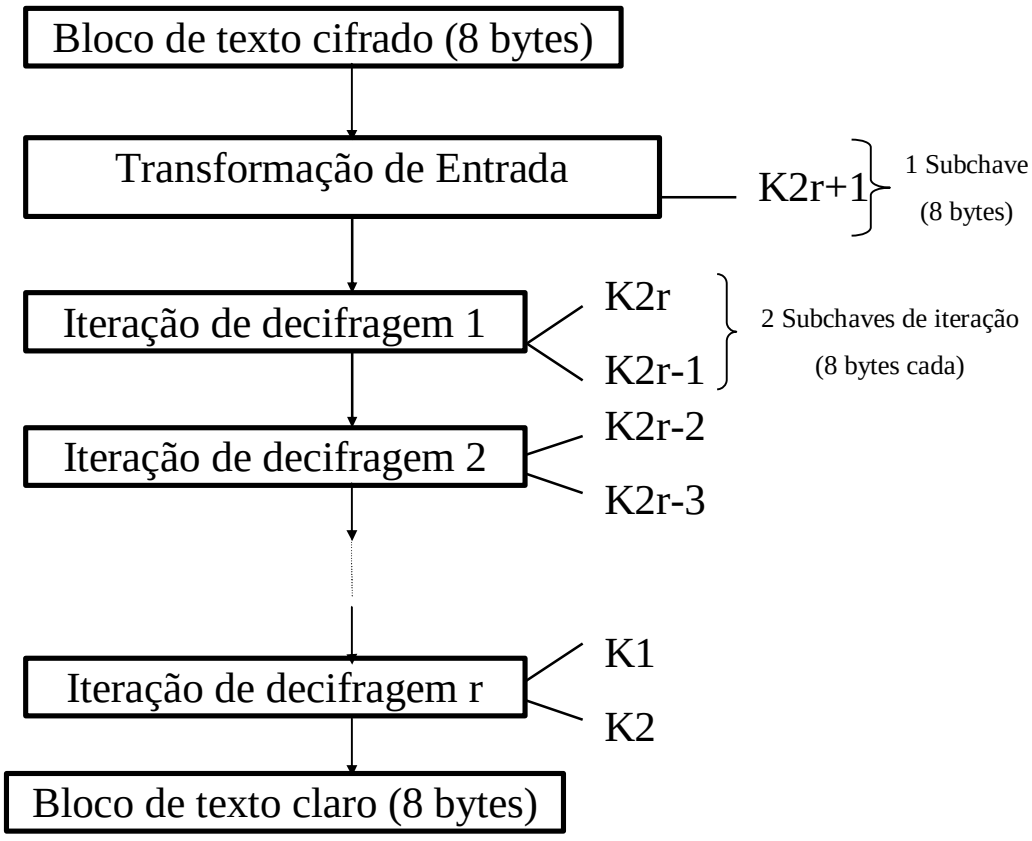


FIGURA 28 - DECIFRAGEM DO SAFER K-64

A decifragem no SAFER K-64 consiste numa transformação de entrada, seguida por r rodadas idênticas. A transformação de entrada é formada por operações de

ou-exclusivo e subtração módulo 256. A operação de ou-exclusivo é feita, bit a bit, entre os bytes 1,4,5 e 8 da subchave (K2r+1) e do texto cifrado, enquanto a operação de subtração módulo 256 é feita dos bytes 2,3,6 e 7 da subchave K2r+1 pelos respectivos bytes do texto cifrado.

A estrutura da rodada de decifragem do SAFER K-64 é mostrada na figura 29.

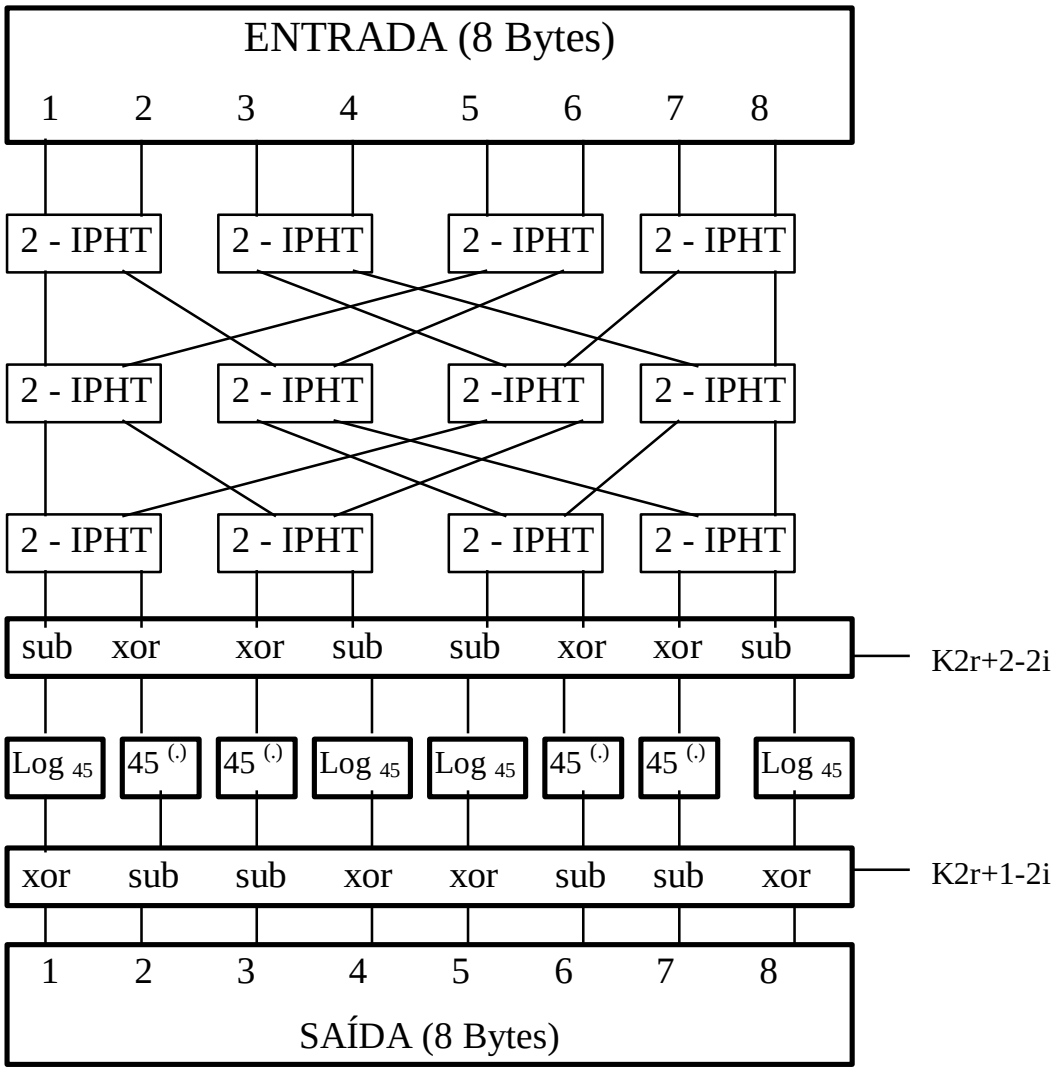


FIGURA 29 - DECIFRAGEM DO SAFER K-64, ESTRUTURA ITERATIVA

Inicialmente, os 8 bytes que entram na i-ésima rodada, passam pela camada linear formada por três níveis da Pseudo Transformada Hadamard Inversa, com duas entradas (2-PHT), que opera sobre um par de entradas (b_1, b_2), gerando o par de saídas (a_1, a_2), onde

$$a1 = b1 - b2 \qquad (2)$$
$$a2 = -b1 + 2b2$$

e a aritmética é módulo 256.

A próxima camada desta i-ésima rodada é a camada linear, constituída de operações de ou-exclusivo (xor) e subtração módulo 256 (sub). Nela, os bytes 1,4,5 e 8 da subchave 2r+2-2i são subtraídos módulo 256 dos respectivos bytes de saída da camada anterior, e os bytes restantes (2,3, 6 e 7) da subchave sofrem a operação de ou-exclusivo, bit a bit, com os respectivos bytes de saída da camada anterior (OU-Exclusivo/Subtração de Bytes Mista).

O próximo passo é a aplicação das operações não lineares "$\log_{45}(.)$" nos bytes 1,4,5 e 8 e "$45^{(.)}$" nos bytes 2,3,6 e 7 da saída da camada linear precedente. O resultado da camada não linear vai, então, para a última camada, que é uma camada linear muito semelhante à segunda. Nesta camada, os bytes 1,4,5 e 8 da subchave K2r+1-2i sofrem, juntamente com os respectivos bytes obtidos da camada não linear, a operação de ou-exclusivo, a qual é feita bit a bit. Já os bytes 2,3,6 e 7 da subchave são subtraídos módulo 256 dos respectivos bytes de saída da camada não linear. Obtêm-se, assim, os 8 bytes que formam a saída da i-ésima rodada [36].

Como o SAFER K-64 funciona e por quê.

Para ver que o SAFER K-64 decifra corretamente, nota-se, inicialmente, que a Transformação de Entrada da decifragem (Ou-Exclusivo/Subtração de Bytes Mista), feita entre os bytes da subchave K2r+1 e os respectivos bytes do texto cifrado (figura 28), desfaz a Transformação de Saída da cifragem, a qual é feita entre os bytes da subchave K2r+1 e os respectivos bytes de saída da r-ésima rodada. (fig.26).

Então, a camada linear inversa da 1ª rodada de decifragem (fig. 29) desfaz a transformação provocada pela camada linear da última rodada de cifragem (fig. 27). A seguir, a camada linear da 1ª rodada de decifragem, constituída de operações de subtração módulo 256 e ou-exclusivo (Ou-Exclusivo/Subtração de Bytes Mista) aplicadas entre os

bytes da subchave K2r e os respectivos bytes da camada linear anterior (fig. 29), desfaz a transformação provocada pela camada linear (Ou-Exclusivo/Adição de Bytes Mista) aplicada entre os bytes da subchave K2r e os respectivos bytes da camada não linear (fig. 27). Nesse caso, a camada não linear inversa da 1ª rodada de decifragem (fig. 29) desfaz o efeito da transformação praticada pela camada não linear da última rodada de cifragem (fig. 27). Finalmente, a camada linear da 1ª rodada de decifragem, formada por operações de ou-exclusivo e subtração módulo 256 (Ou-Exclusivo/Subtração de Bytes Mista) aplicadas entre os bytes da subchave K2r-1 e os respectivos bytes provenientes da camada não linear inversa, desfaz a transformação provocada pela camada linear da última rodada de cifragem (Ou-Exclusivo/Adição de Bytes Mista) ocorrida entre os bytes da subchave K2r-1 e os respectivos bytes do texto claro. Da mesma forma, a transformação da i-ésima rodada desfaz a transformação ocorrida na rodada r+1-i para i=2,3,...,r. Assim, observa-se que a decifragem recupera de fato o texto claro original.

O *design* do SAFER K-64 foi feito obedecendo aos princípios de Shannon, da confusão e da difusão, para a obtenção de segurança nas cifras de chave privada. As operações de transformações empregadas provocam a confusão necessária para fazer com que a estatística do texto cifrado dependa, de maneira complicada, da estatística do texto claro, dado que pequenas mudanças se difundem rapidamente através da cifra. A fim de garantir a difusão desejada foi criada uma nova e não ortodoxa transformada, a Pseudo-Transformada de Hadamard, que faz com que o byte de entrada afete cada byte de saída, isto é, a Pseudo-Transformada de Hadamard proporciona completa e garantida difusão em cada camada linear.

A rápida difusão proporcionada pela Pseudo-Transformada de Hadamard é a principal razão pela qual a escolha de r = 6 é suficiente para fazer com que o SAFER K-64 seja resistente a “quebras” [36].

Processo de obtenção das subchaves

O procedimento para a geração das subchaves K2,K3, ..., K2r+1, a partir da subchave selecionada para o usuário K1, é mostrado na figura 30. As quantidades B1,B2,B3,...,B2r+1 são as chaves de polarização, de 8 bytes cada, responsáveis por proporcionar às subchaves um caráter "aleatório" e também por evitar que mais de uma das subchaves de rodada seja nula. Seja b[i, j] o *j*-ésimo byte de B_i, o mesmo é expresso por

$$b[i, j] = 45^{45^{(9i+j+1) \bmod 257} \bmod 257} \bmod 257 \quad (3)$$

Tal equação fornece as chaves de polarização utilizadas no SAFER K-64. Uma tabela com os valores precisos das chaves de polarização é fornecido para o SAFER K-64. Pode ser observado em tal tabela que os valores das chaves de polarização se mostram "aleatórios". O uso de uma chave de polarização para a geração das subchaves, além de novo, é, em geral, uma boa idéia para cifras iterativas, pois evita as "chaves fracas". Nota-se, no processo de geração das subchaves, que a chave selecionada pelo usuário é rotacionada 3 bits à esquerda, entre as adições módulo 256, com cada uma das novas polarizações. Idealiza-se que todas as subchaves (K1,K2,...,K2r+1) tenham o caráter de uma seqüência de subchaves independentemente escolhidas e uniformemente aleatórias. Claro que isto não é alcançado, já que todas foram determinadas tomando como base a subchave K1 (selecionada para o usuário). Assim, a grande meta do *design* do processo de geração das subchaves é fazer com que a relação das mesmas com a subchave geradora K1 seja tão complicada, que não possa ser usada pelo criptoanalista; este, aliás, é o propósito tanto das rotações quanto das adições da subchave com a chave de polarização.

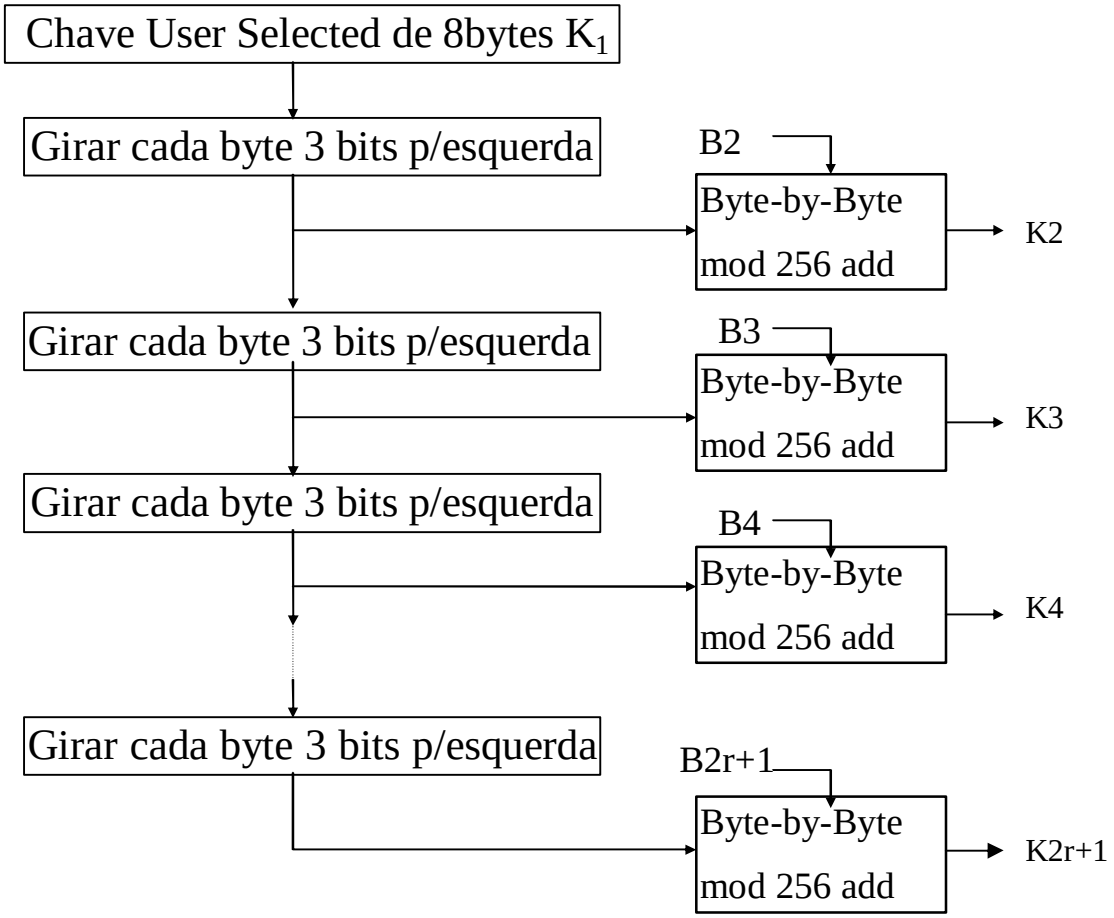


FIGURA 30 – ALGORITMO PARA A GERAÇÃO DAS SUBCHAVES

Considerações sobre segurança no SAFER K-64

Mostrou-se como o SAFER K-64 alcança tanto boa confusão quanto boa difusão, as quais são dois atributos básicos para que uma cifra de bloco iterativa seja segura. Atualmente, a melhor ferramenta para a medição da segurança proporcionada por uma cifra iterativa é a resistência da mesma à criptoanálise diferencial. Muitos testes foram feitos por criptoanalistas contratados pela Cylink Corporation, os quais não tinham nenhum vínculo com o projeto. Além dos testes envolvendo ataques por criptoanálise diferencial, também foram feitos extensivos estudos estatísticos da cifra, com o objetivo de encontrar alguma fraqueza na mesma. Concluiu-se, com tais testes, que o SAFER K-64 com 6 rodadas é resistente à criptoanálise diferencial e também que nenhum tipo de fraqueza foi encontrada no mesmo [36].

SAFER K-128

Logo após o anúncio do SAFER K-64, surgiram pedidos para uma versão com chave selecionada pelo usuário de comprimento 128 bits. Tal pedido foi atendido pelo Special Projects Team of Ministry of Home Affair, Singapura, que tomou a iniciativa de criar um novo algoritmo para a geração das subchaves, a partir da chave selecionada pelo usuário, agora com 128 bits [37].

O SAFER K-128 é uma cifra com estrutura de rodada, transformação de saída e chaves de polarização idênticas ao SAFER K-64, porém o comprimento da chave selecionada pelo usuário é de 128 bits. Recomenda-se, nesta implementação, a utilização de 10 rodadas (r = 10) sem, no entanto, utilizar mais que 12 rodadas. O algoritmo criado pelo Special Projects Team of Ministry of Home Affair, Singapura, pode ser visto na figura 31 . A chave selecionada pelo usuário de 128 bits é dividida em duas metades: a metade esquerda é denominada Ka, enquanto que a metade direita é denominada Kb. Comparando os dois processos de geração de subchaves do K-64 e K-128, nota-se que se Kb = K1 (chave de 64 bits selecionada pelo usuário), as subchaves K3, K5, K7, ..., K2r+1 serão as mesmas geradas por ambos os algoritmos. Similarmente, se Ka = K1, as subchaves K2, K4, K6, ..., K2r são geradas por ambos os algoritmos.

Concluímos, assim, que, se tanto Ka quanto Kb forem idênticas a K1, todas as subchaves produzidas são as mesmas para ambos os algoritmos. Esta é a característica que tornou tal algoritmo tão atrativo, já que o mesmo mantém a compatibilidade entre as duas versões (SAFER K-64 e SAFER K-128). O usuário que possua a implementação SAFER K-128 pode utilizá-la como SAFER K-64, para cifrar ou decifrar quando desejado. O SAFER K-128 também é uma cifra não proprietária.

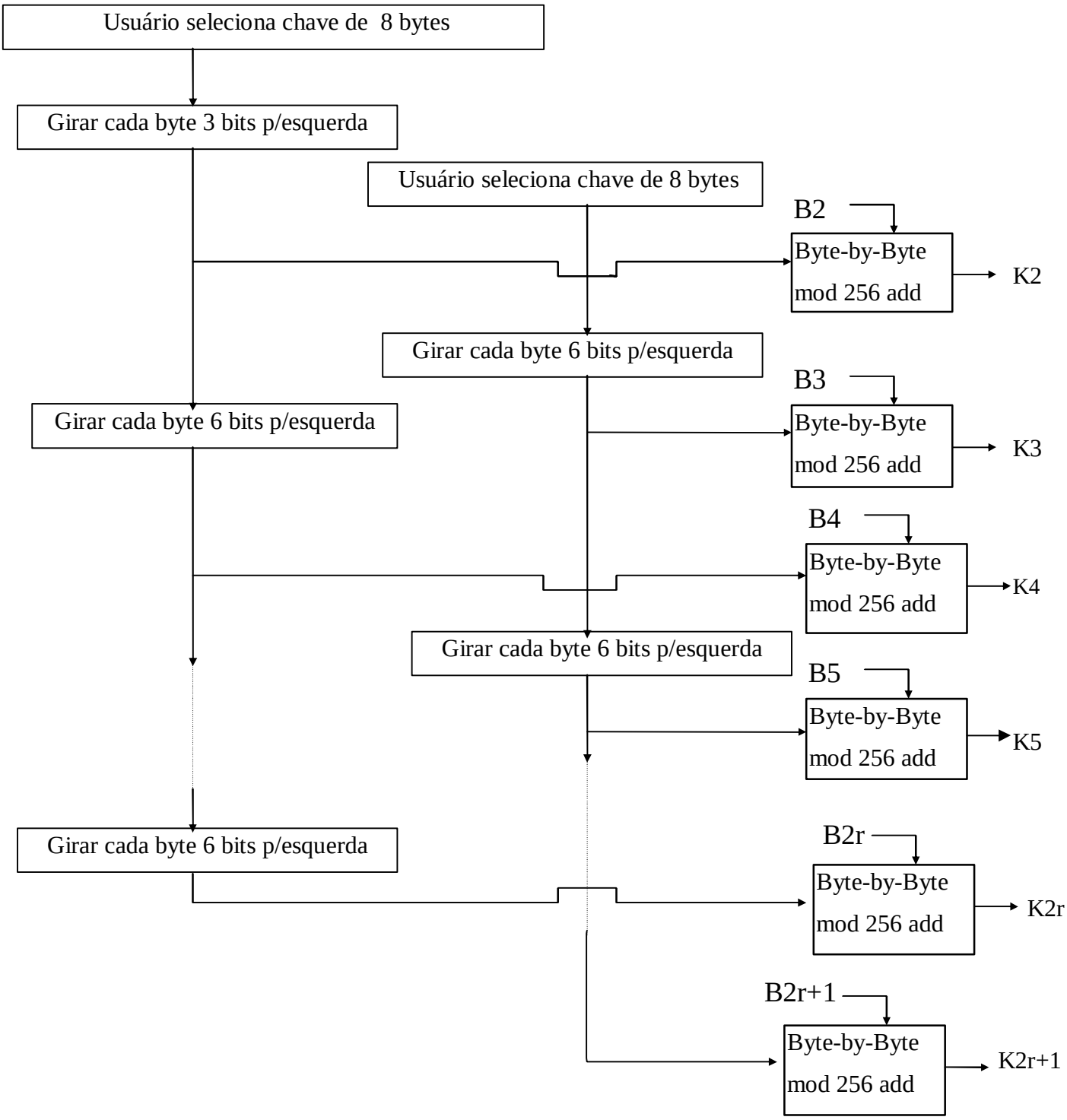


FIGURA 31 – ALGORITMO PARA GERAÇÃO DAS SUBCHAVES PARA O SAFER K-128

4.2 SAFER +

Faremos uma breve exposição sobre o SAFER +, esclarecendo que a princípio, o sistema que será adotado na Polícia Militar será o SAFER K-128.

O SAFER + é a proposta da Cylink Corporation, para o Advanced Encryption Standard. Os inventores do algoritmo são o Prof. James L. Massey, Gurgen H. Khachatrian e Melsik K. Kuregian.

A Cylink abre mão de todos seus direitos proprietários sobre o SAFER + e consigna este algoritmo ao domínio público.

Fundamentos do SAFER +

O SAFER + é baseado na família de cifras SAFER existentes, a qual compreende as cifras SAFER K-64, SAFER K-128, SAFER SK-64, SAFER SK-128, e SAFER SK-40.

O comprimento de bloco de todas as cifras da atual família SAFER é de 64 bits, enquanto que o comprimento da chave é de 40 ou 64 ou 128 bits conforme indicado no nome da cifra, ou seja diferem da SAFER K-64 apenas nos esquemas da chave e no número de iterações usado.

A estrutura de Cifragem do SAFER + utiliza um comprimento de bloco de 16 bytes (128 bits), com comprimento da chave de 128 ou 192 ou 256 bits, devendo ter um número de 8, 12 e 16 iterações respectivamente, conforme o comprimento das chaves utilizadas.

A estrutura de decifragem é muito semelhante à de cifragem, com os mesmos comprimentos de bloco e de chaves, porém não é idêntica.

SAFER + não é nem uma cifra de Feistel nem uma cifra de permutação-substituição, mas sim uma cifra de substituição/transformação linear.

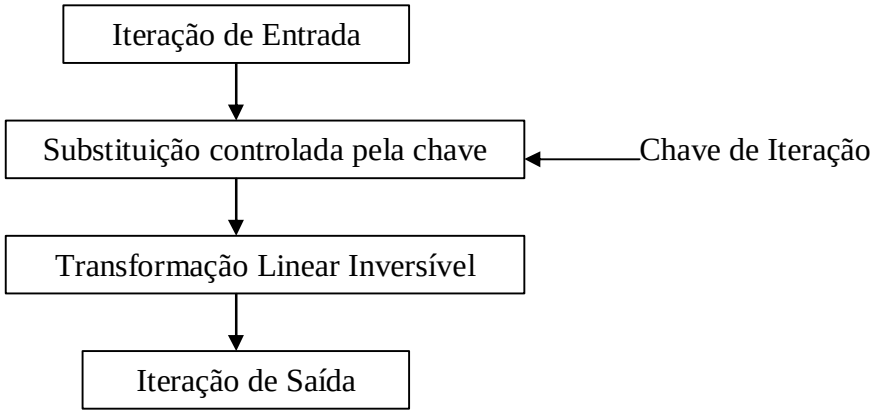


FIGURA 32 – ESTRUTURA DE ITERAÇÕES DO SAFER +

A substituição Controlada-pela-Chave provê *confusão* e a Transformação Linear Inversível provê *difusão*. Na substituição controlada por chave, realizam-se operações "xor" que denota a adição de bytes modulo 2 bit-a-bit; "add" denota a adição de bytes modulo 256; "exp" denota a função exponencial 45^{x} modulo 257, sendo que para x = 128, a exponencial é igual à zero; "log" denota a função logarítmica $\log_{45}(x)$, com a convenção que $\log_{45}(0) = 128$.

Observa-se que a estrutura de iterações do SAFER + obedece a mesma estrutura da família SAFER K-64, na qual a Transformação Linear Inversível é baseada também, na Transformada Pseudo-Handamard (PHT), sendo empregado um "Embaralhamento Armênio" que consiste na permutação de coordenadas.

Um estudo exaustivo do SAFER + mostrou que todas as características de 5 iterações têm probabilidade significativamente menor que 2^{-128} (porém este não é o caso para apenas 4 iterações) contra um ataque.

SAFER + com seis ou mais iterações (mas nunca menos) é seguro contra criptoanálise diferencial, mas para uma margem de segurança desejável, escolheu-se 8 iterações para o SAFER + com o esquema de chave de 128 bits, que provêem uma enorme margem de segurança, inclusive contra a criptoanálise linear [40].

Esquema de Chave (128 bits) do SAFER +

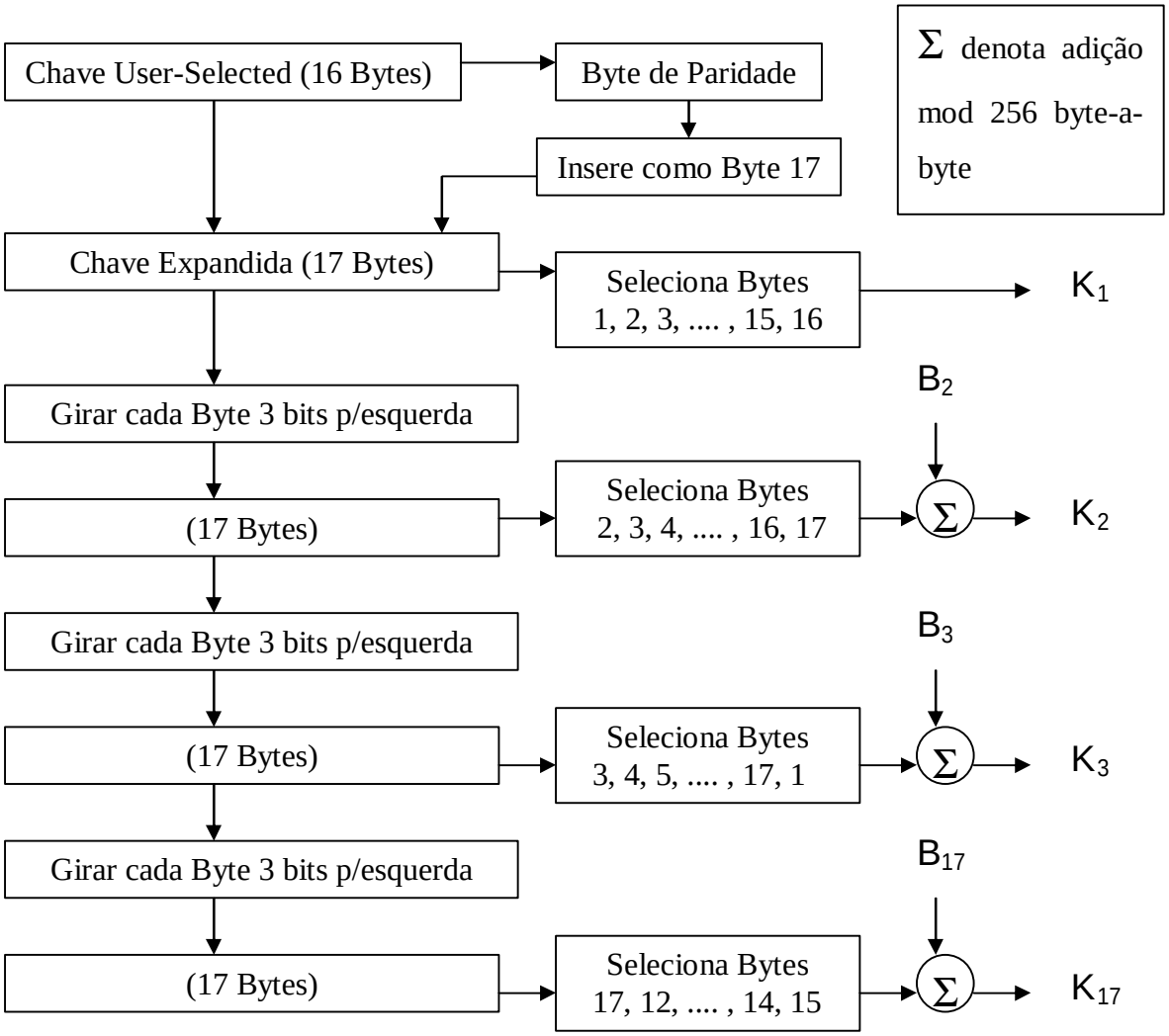


FIGURA 33 – ESQUEMA DE CHAVE (128 BITS) DO SAFER +

O uso do Byte de paridade e da rotação progressiva na seleção dos Bytes foi sugerida pelo Dr. Lars Knudsen (Univ. of Bergen, Norway). As chaves de polarização (B_2, B_3, ... B_{17}) são computadas por "dupla exponenciação"com a função 45^x mod 257 [41].

4.3 Como usar o SAFER

Este pacote de *software* é uma implementação do algoritmo de cifra de bloco orientado a byte. Quatro versões do algoritmo são implementadas, a saber: SAFER K-64, SAFER K-128, SAFER SK-64 e SAFER SK-128. Os numerais 64 e 128 correspondem ao comprimento da chave selecionada pelo usuário. O padrão "K" é associado à palavra chave em inglês "key", e as novas versões implementadas "SK" significam um fortalecimento da chave. A interface para o usuário é também fornecida para os sistemas UNIX, MS-DOS, VMS e outros.

Qualquer compilador baseado em ANSI C ou C++ pode ser usado para compilar o código fonte. Além disso, o comportamento da entrada-saída dos programas executáveis é idêntico, isto é, o comando do usuário da função 'safer' é compatível em qualquer computador [38].

Como já foi referenciado, o código fonte pertence ao domínio público.

Arquivos do SAFER

- README – esta documentação
- safer.c – códigos fonte do SAFER K-64, K-128, SK-64 e SK-128
- safer.h – cabeçalho do arquivo 'safer.c'
- safercmd.c – código fonte para o comando do usuário 'safer'
- makefile – descreve como gerar o comando do usuário 'safer'
- safer.exe - comando do usuário 'safer' executável para MS-DOS
- mani/safer.1 – manual do comando do usuário 'safer' para o UNIX
- SAFERCMD.TXT – descrição do comando do usuário 'safer'
- SAFER_SK.TXT – paper: Massey, J. L., "Announcement of a Strengthened Key Schedule for the Cipher SAFER", Sep. 9, 1995.
- SAFER_40.TXT – paper: Massey, J. L., "Announcement of a 40-Bit Key Schedule for the Cipher SAFER", Oct. 22, 1995.

- SCHNEIER.TXT – paper: Massey, J. L., “Comment on the insinuation about SAFER and the NSA written in the book, Applied Cryptography, Bruce Schneir”, Nov. 20, 1995.
- check.c – código fonte do programa usado para checar a forma correta do comando ‘safer’. Tipo ‘make checking’ executa o teste que produz ‘check.out’como saída do arquivo. Se o comando da função ‘safer’ estiver correto, o arquivo ‘check.out’ e ‘check.ref’ são idênticos.
- check.ref – exemplos de dados cifrados

Servidor – FTP

Este software pode ser obtido, executando-se os seguintes comandos:

[ ] ftp:// ftp.isi.ee.ethz.ch

Name: anonymous

Password: your_e-mail_address

cd pub/simpl

binary

get safer.V1.2.tar.Z

quit

[ ]uncompress safer.V1.2.tar.Z

[ ]tar –xf safer.V1.2.tar

[ ]cd safer

Instalação

1. Modificar a linha “CC ...” no “*makefile*” (e nenhuma outra linha!!)
2. Execute os comandos que se seguem:

   [ ] man –M . safer mostra o manual do safer (somente UNIX)

   [ ] make produz o comando do usuário ‘safer’

   [ ] make checking verifica a correção do comando ‘safer’

Maiores informações são descritas no Apêndice C.

4.4 Sistema de Gerenciamento de Chaves (SGC) aplicado ao SAFER

Apresentaremos nesta seção, o Sistema de Gerenciamento de Chaves (SGC), desenvolvido para a Polícia Militar de Pernambuco, cujo objetivo principal é o armazenamento de chaves públicas em um banco de dados denominado Catálogo Público Custodiado (CPC), o qual fornece subsídio para o cálculo de uma chave secreta comum, necessária para efetivar a troca de mensagens entre usuários de uma rede, por meio da utilização da cifra SAFER.

Para tanto, daremos um enfoque inicial nas definições de funções unidirecionais e sistema de distribuição pública de chaves de Diffie-Hellman, que foram os princípios utilizados para a criação do SGC.

4.4.1 Funções Unidirecionais

Em 1976, dois pesquisadores da Universidade de Stanford, Whitfield Diffie e Martin E. Hellman, publicaram um artigo intitulado "*New Directions in Cryptography*", o qual abalou as fundações do pensamento criptográfico [25]. Diffie e Hellman sugeriram que é possível conceber sistemas criptográficos computacionalmente seguros, que não necessitam de nenhum canal seguro para a troca de chaves secretas [13].

A contribuição fundamental do artigo de Diffie e Hellman consistiu da definição de funções unidirecionais e como tais funções poderiam eliminar a necessidade de trocar chaves secretas em sistemas computacionalmente seguros.

Uma função unidirecional, como definida por Diffie e Hellman é uma função f tal que,

(1) para todo x no domínio de f é fácil computar $f(x)$, mas

(2) para virtualmente todo y no contra-domínio de f é computacionalmente inviável encontrar um x tal que $f(x) = y$.

A maior aplicação de funções unidirecionais é em sistemas de senhas, em que a cifragem e decifragem não se fazem necessárias.

Suponhamos, por exemplo, que o usuário A escolhe uma senha x_A e apresenta esta senha inicialmente ao sistema, então é computado $y_A = f(x_A)$, este valor é armazenado no arquivo de senhas como o par (A, y_A).

Quando mais tarde um outro usuário B, fingindo ser o usuário A, apresentar a senha x_B ao sistema, o sistema irá computar $y_B = f(x_B)$ e então verificar se (A, y_B) encontra-se no arquivo de senhas. Caso seja encontrado o par (A, y_B), ou seja, tal que o $y_B = y_A$, o sistema permitirá o acesso do usuário B; caso contrário, ele terá seu acesso negado [13].

A possibilidade do usuário B, acima, conseguir encontrar um x_B tal que $y_A = f(x_B)$, é computacionalmente inviável. Esta é a vantagem deste sistema, o qual não exige que o arquivo de senhas seja mantido em sigilo.

4.4.2 Exponenciação Discreta e a Função Unidirecional de Diffie-Hellman-Pohlig

Seja $< G, * >$ um grupo cíclico de ordem n e seja a um gerador deste grupo. Então, usaremos a expressão exponenciação discreta na base a em $< G, * >$ para significar a função $f: Z_n \longrightarrow G$ tal que

$$f(x) = a^x.$$

Os valores de f (x), resultantes de x tomando valores em Z_n = { 0, 1, ..., n – 1}, são todos distintos.

A conjectura de Diffie-Hellman-Pohlig é de que a exponenciação discreta no grupo multiplicativo de GF (p), é uma função unidirecional desde que p seja um número primo grande (com pelo menos 100 dígitos decimais), tal que $n = p - 1$ também tenha um fator primo grande, pois a segurança do sistema é baseada na dificuldade de fatorar números primos grandes. O número

$$p' = \frac{p-1}{2}$$, também deve ser um primo [13, 27].

4.4.3 O Sistema de Distribuição Pública de Chaves de Diffie-Hellman

O sistema sugerido por Diffie-Hellman para a criação de uma chave secreta comum, foi um bastante original e inteligente, pois por intermédio desse sistema pode-se trocar chaves secretas sem a necessidade de um canal seguro e, baseia-se na propriedade unidirecional da exponenciação discreta [13].

Sua segurança consiste na dificuldade de se calcular logaritmos discretos em corpos finitos, comparado com a facilidade de se calcular exponenciais no mesmo corpo finito [33].

Suponhamos que $f (x) = a^x \mod n$, tal que n é um número primo (p) grande (com pelo menos 100 dígitos decimais), e a um elemento primitivo de p em GF (p), sendo, então, $f (x) = y$, uma função verdadeiramente unidirecional e de conhecimento de todos os usuários autorizados a acessar o sistema.

Diffie e Hellman postularam a existência de um Catálogo Público Custodiado (CPC), contendo um banco de informações autênticas, com os valores de $f(x) = y$ não-confidencial, que está disponível para todos os usuários cadastrados [13].

Digamos que temos dois usuários, A e B, autorizados a acessar o sistema. Após a identificação ao Sistema de Gerenciamento de Chaves (SGC) e o respectivo "*login*" , os

$$\text{Usuário A:} \qquad y_A = a^{x_A} \mod p$$
$$\text{Usuário B:} \qquad y_B = a^{x_B} \mod p$$

usuários escolhem uma chave privada x_A e x_B para computar o valor da chave pública, da seguinte maneira:

Estas chaves públicas são, então, armazenadas no Catálogo Público Custodiado (CPC), como os pares (A, y_A) e (B, y_B), correspondendo respectivamente aos usuários A e B.

Suponhamos, ainda, que os usuários A e B desejam se comunicar secretamente, então

(1) Os usuários A e B consultam no CPC, as chaves públicas de A, y_A e de B, y_B , respectivamente.

(2) A irá computar a chave secreta comum, elevando a chave pública de B, y_B, a um expoente igual a sua chave privada x_A, seguindo da mesma forma o usuário B:

$$A \Rightarrow K = (y_B)^{x_A} \mod p = (a^{x_B})^{x_A} \mod p$$
$$B \Rightarrow K' = (y_A)^{x_B} \mod p = (a^{x_A})^{x_B} \mod p$$

Este número computado da chave K,

$$K = K' = a^{x_A \, x_B} \mod p$$

que tanto o usuário A como o usuário B podem calcular, é a chave secreta comum de ambos, a qual eles podem, agora, usar como a chave secreta num criptosistema convencional de chave secreta (SAFER).

O que o esquema de Diffie-Hellman fornece é portanto um modo público de distribuir chaves secretas, e este esquema é usualmente chamado de *Sistema de Diffie-Hellman de Distribuição Pública de Chaves.*

Um ataque a este sistema é inviável, se a exponenciação discreta for realmente unidirecional. Ainda não se tem notícia de um ataque ao sistema de Diffie-Hellman que não fosse computacionalmente equivalente a computar logaritmos discretos, nem alguém conseguiu ainda provar que todos ataques a este sistema são computacionalmente equivalentes a computar logaritmos discretos. Este é, portanto, geralmente considerado como um dos melhores sistemas [13].

4.4.4 Resultado Experimental

Apresentaremos, agora, por meio de um exemplo, a funcionalidade do *Sistema de Gerenciamento de Chaves* (SGC), que foi desenvolvido, por ocasião desta dissertação, para aplicabilidade na Polícia Militar de Pernambuco.

O SGC, tem por função a criação de uma chave pública dos usuários cadastrados e autorizados a acessar o sistema, gerando conseqüentemente um banco de dados no qual são armazenados os pares (Usuário e Chave-Pública), o qual denominamos *Catálogo Público Custodiado* (CPC).

A chave pública, gerada e armazenada no CPC, deve sofrer atualizações periódicas, conforme determinação do gerente do sistema (administrador).

O administrador do sistema é o único responsável pelo cadastro (inclusão e exclusão) dos usuários, e pela alteração dos valores das constantes a e p para o cálculo da função $f(x) = Y$.

No CPC estão disponíveis as chaves públicas, $Y = a^x \bmod n$, dos usuários. Se um usuário A deseja se comunicar com o usuário B, A precisa recuperar a chave pública de B e gerar a sua chave secreta comum. A pode, então, cifrar uma mensagem,usando o SAFER, com a chave secreta comum e enviá-la para B. B pode recuperar a chave pública de A no CPC, e gerar a sua chave secreta comum. Cada usuário deve ter uma única chave privada, e nenhuma comunicação prévia é necessária para o envio da mensagem [33].

Temos, ainda, no SGC, um "*link*" com o SAFER, facilitando o transporte dos dados obtidos da chave secreta comum para o SAFER.

O SGC, obedece aos princípios formulados por Diffie e Hellman em seu sistema de distribuição pública de chaves. Utilizamos a função,

$f(x) = Y = a^x \bmod n$, em que

- x é o valor da chave-privada escolhida secretamente pelo usuário;
- Y é o valor da chave-pública obtida pelo cálculo de $f(x)$;
- a é a raiz primitiva, geradora de todos os elementos não-nulos do corpo finito GF(p);
- n é um número primo grande p (com pelo menos 100 dígitos decimais) e (p – 1) / 2 também é um número primo.

Os valores de a e p são fixos, variando apenas o valor da chave privada (x) do usuário. No entanto, em períodos predeterminados, os valores de a e p devem ser alterados pelo administrador do sistema dificultando, assim, possíveis ataques.

Exemplificação do Sistema de Gerenciamento de Chaves

(1) Inicialmente, solicitamos a entrada no SGC, na qual visualizamos a seguinte tela de entrada,

FIGURA 34 – TELA DE ENTRADA NO SISTEMA DE GERENCIAMENTO DE CHAVES (SGC)

(2) Em seguida , temos a solicitação da matrícula e senha do usuário, que sendo cadastrado no sistema, será executado o "*login*". No caso do nosso exemplo, o acesso será do próprio administrador a fim de visualizarmos os recursos disponíveis.

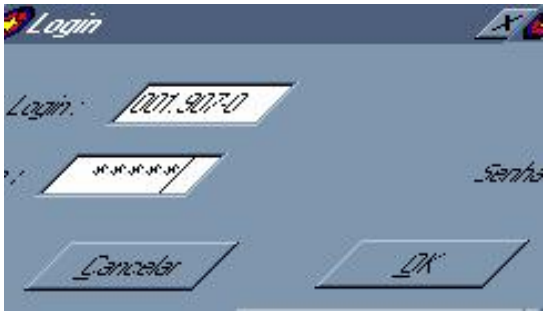


FIGURA 35 – "LOGIN" DO SISTEMA DE GERENCIAMENTO DE CHAVES

(3) Após efetuar o "*login*" do SGC, entramos na tela principal do sistema, onde, conforme podemos visualizar na figura 36, temos disponíveis as opções de Cadastro, CPC, Chave Secreta, SAFER, MAPLE e Administração.

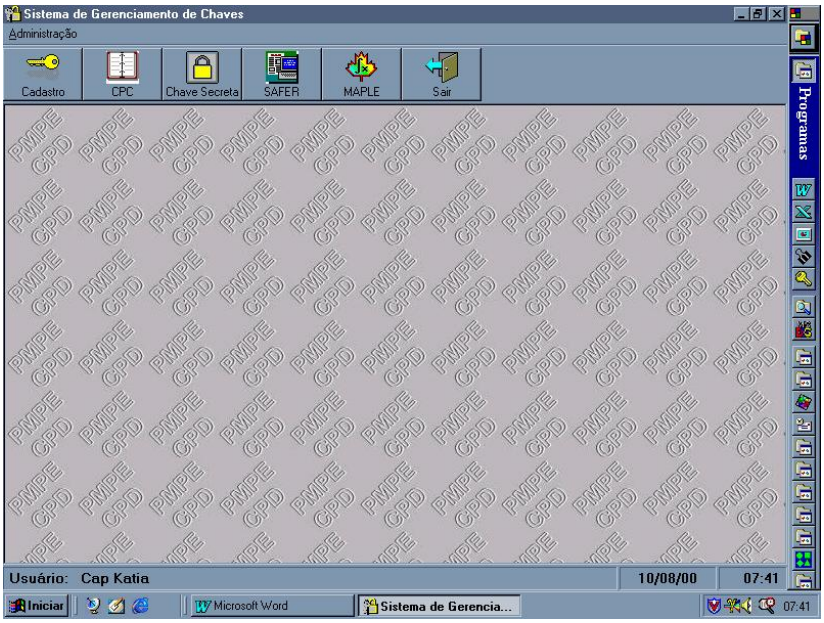


FIGURA 36 – TELA PRINCIPAL DO SISTEMA DE GERENCIAMENTO DE CHAVES

(4) Os recursos disponíveis do administrador são cadastro de usuários e alteração das constantes a e p da função $f(x) = a^x \mod p$. Na figura 37, verificamos o cadastro do usuário – Cap Roberto – realizado pelo administrador – Cap Kátia -.

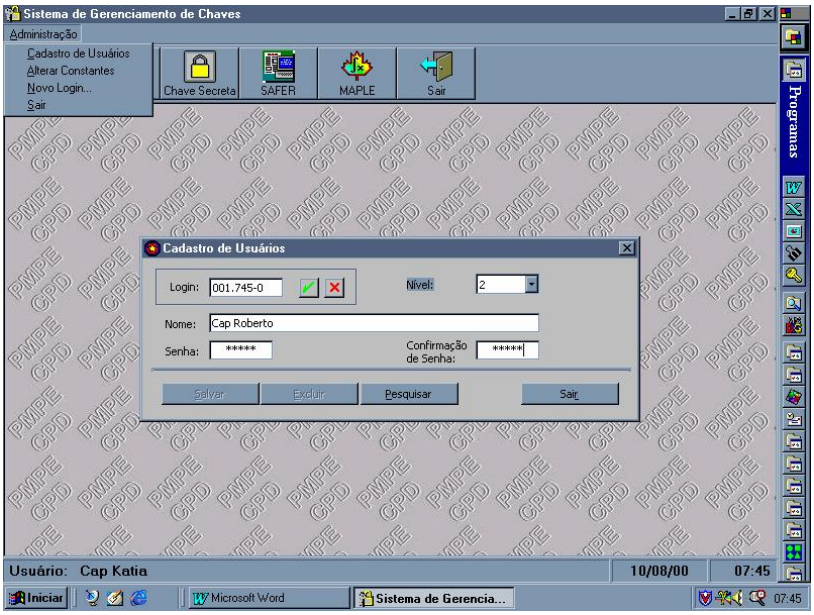


FIGURA 37 – CADASTRO DE USUÁRIOS DO SGC

(5) Temos ainda no cadastro de usuários um campo de pesquisa para controle do administrador, com as opções de exclusão, inclusão e alteração, o nível estipulado no cadastro para o administrador é 1, enquanto para os demais usuários é 2.

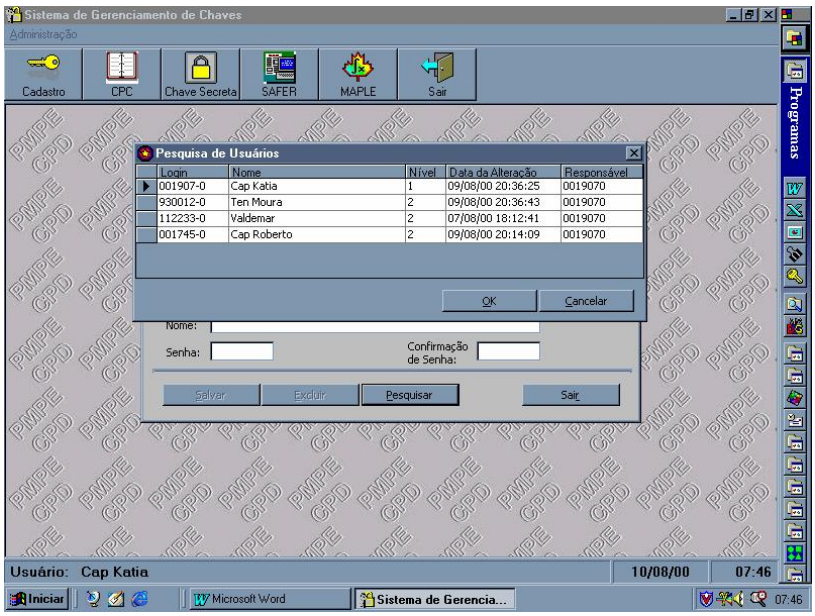


FIGURA 38 – PESQUISA DOS USUÁRIOS CADASTRADOS NO SGC

(6) Na figura 39, abaixo, visualizamos o outro recurso disponível do administrador que é a alteração das constantes, sendo que, no nosso exemplo, adotamos para a o valor 2 e para p um número primo com 101 dígitos decimais.

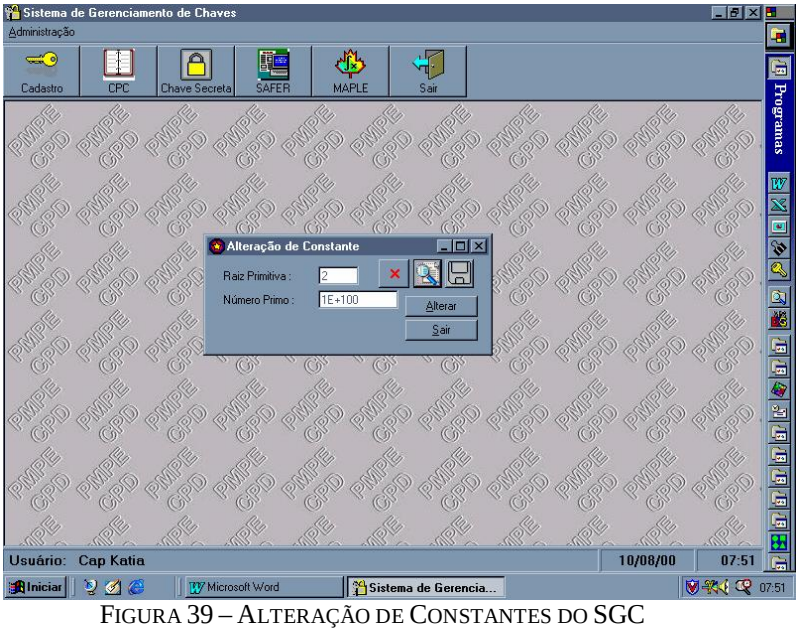


FIGURA 39 – ALTERAÇÃO DE CONSTANTES DO SGC

(7) Na figura 40, visualizamos o cadastro da chave pública do usuário – Cap Kátia -, no qual é solicitado o valor da chave privada, em seguida, clicando-se em calcular, obtém-se o valor da chave pública que é automaticamente armazenado no CPC – Catálogo Público Custodiado (figura 41).

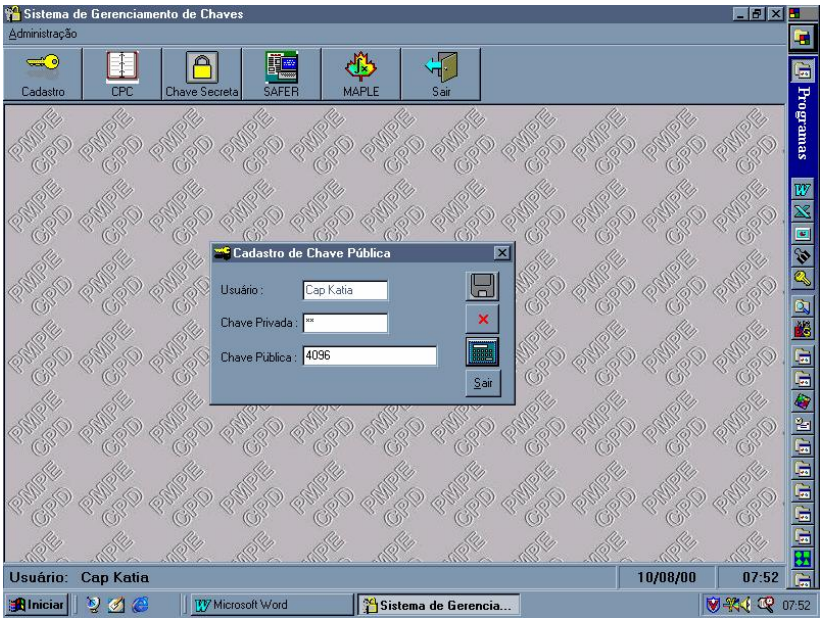


FIGURA 40 – CADASTRO DA CHAVE PÚBLICA DO SGC

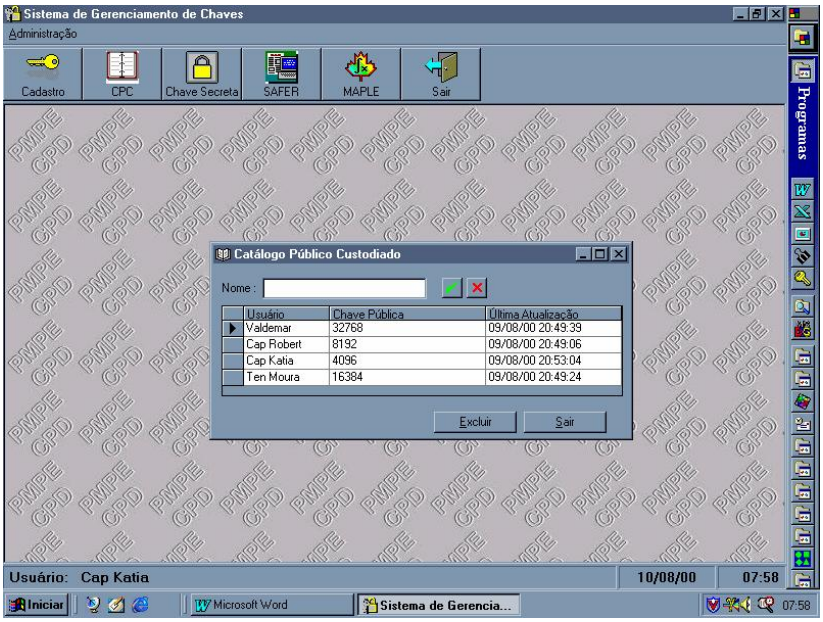


FIGURA 41 – CATÁLOGO PÚBLICO CUSTODIADO (CPC) DO SGC

(8) Suponhamos agora, que a Cap Kátia deseje enviar uma mensagem cifrada ao Cap Roberto. O procedimento inicial é buscar no CPC, o valor da chave pública do Cap Roberto. Assim que for localizado a chave pública, clica-se exatamente no nome desejado, o sistema executa um "*link*" para a tela da chave secreta, ou se preferir, pode-se clicar no menu, campo chave secreta (figura 42).

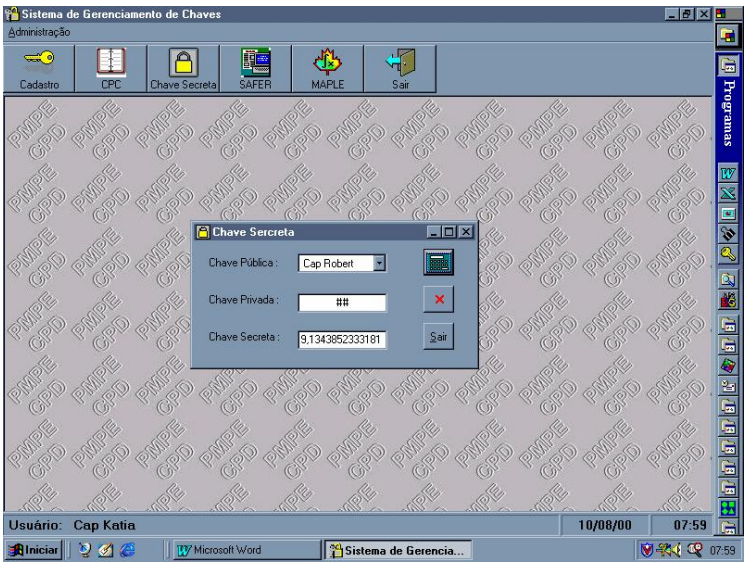
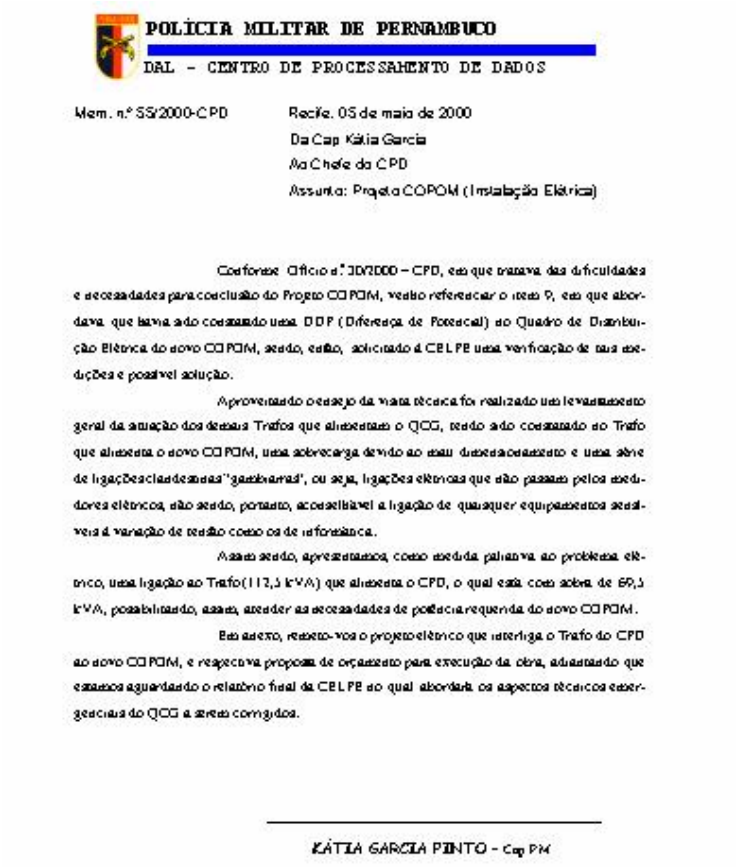


FIGURA 42 – GERAÇÃO DA CHAVE SECRETA COMUM DO SGC

Caso o nome do Cap Roberto seja acionado no CPC é realizado um "*link*" para a Chave Secreta, com o transporte respectivo do nome; a Cap Kátia, digita, então, a sua chave privada e em seguida clica em calcular, obtendo, assim, o valor da chave secreta comum, necessária para cifrar a mensagem no SAFER. No exemplo, o resultado obtido da chave secreta foi 9,13438523331814E+46.

(9) Já tendo selecionado o arquivo da mensagem confidencial COPOM.doc (figura 43), na qual a Cap Kátia deseja cifrar e enviar ao Cap Roberto, conforme podemos observar no arquivo do texto claro selecionado:



POLÍCIA MILITAR DE PERNAMBUCO
DAL - CENTRO DE PROCESSAMENTO DE DADOS

Mem. n.º SS/2000-CPD Recife, 05 de maio de 2000
Da Cap Kátia Garcia
Ao Chefe do CPD
Assunto: Projeto COPOM (Instalação Elétrica)

Conforme Ofício nº 30/2000 – CPD, em que tratava das dificuldades e necessidades para conclusão do Projeto COPOM, venho referenciar o item 9, em que abordava que havia sido constatado uma DDP (Diferença de Potencial) no Quadro de Distribuição Elétrica do novo COPOM, sendo, então, solicitado à CELPE uma verificação de tais condições e possível solução.

Aproveitando o ensejo da visita técnica foi realizado um levantamento geral da situação dos dezesseis Trafos que alimentam o QCG, tendo sido constatado no Trafo que alimenta o novo COPOM, uma sobrecarga devido ao mau dimensionamento e uma série de ligações clandestinas "gambiarras", ou seja, ligações elétricas que não passam pelos medidores elétricos, não sendo, portanto, aconselhável a ligação de quaisquer equipamentos sensíveis à variação de tensão como os de informática.

Assim sendo, apresentamos, como medida paliativa ao problema elétrico, uma ligação ao Trafo(112,5 kVA) que alimenta o CPD, o qual está com sobra de 69,5 kVA, possibilitando, assim, atender as necessidades de potência requerida do novo COPOM.

Em anexo, remeto-vos o projeto elétrico que interliga o Trafo do CPD ao novo COPOM, e respectiva proposta de orçamento para execução da obra, salientando que estamos aguardando o relatório final da CELPE no qual abordará os aspectos técnicos emergenciais do QCG a serem corrigidos.

KÁTIA GARCIA PINTO - Cap PM

FIGURA 43 – ARQUIVO DO TEXTO CLARO (COPOM.DOC)

Fazendo um "*link*" para o SAFER, e executando o comando necessário para cifrar a mensagem (figura 44):

C:\ SAFER> safer –e –ecb –k 9,13438523331814E+46 COPOM.doc COPOM.cry

Chave secreta comum calculada no SGC

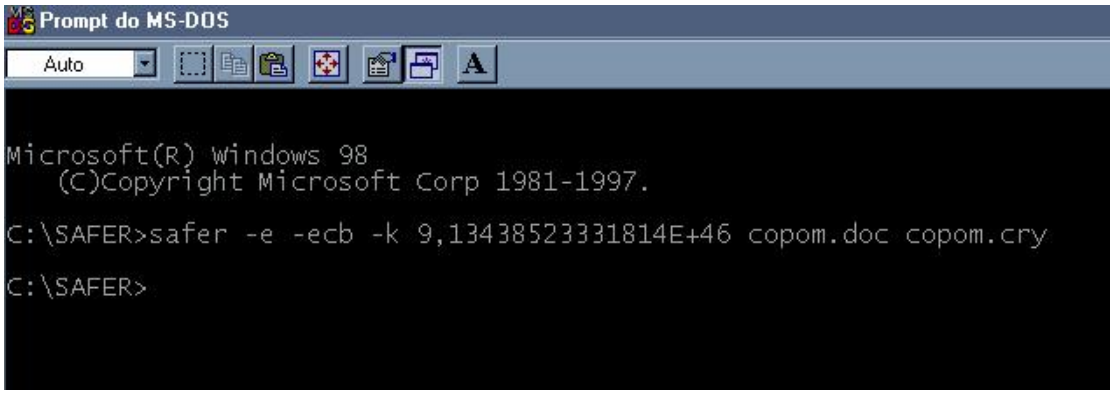


FIGURA 44 – EXECUÇÃO DO COMANDO DE CIFRAGEM NO SAFER

Obtemos, assim, o arquivo cifrado COPOM.cry, o qual apresentamos na figura 45.

FIGURA 45 - TRECHO DO TEXTO CIFRADO OBTIDO DO ARQUIVO COPOM.CRY

Para decifrar o arquivo COPOM.cry e recuperar o texto original, executamos o seguinte comando no SAFER:

C:\ SAFER> safer –d – k 9,13438523331814E+46 COPOM.cry COPOM.ori

Chave secreta comum calculada no SGC

No qual a chave secreta comum para decifragem do texto foi calculada pelo receptor da mensagem – Cap Roberto -, ao verificar no CPC, o valor da chave pública da -Cap Kátia-, e computando a chave secreta por meio da exponenciação discreta realizada pelo sistema, em que a chave pública da -Cap Kátia- é elevada à chave privada do -Cap Roberto-

Apresentamos no próximo capítulo, o Teleprocessamento na Polícia Militar de Pernambuco com sua fundamentação jurídica.

CAPÍTULO 5

TELEPROCESSAMENTO NA POLÍCIA MILITAR DE PERNAMBUCO

O Centro de Processamento de Dados (CPD) foi criado através da Lei nº 6.772, de 03 de outubro de 1974, porém sua ativação se deu em 24 de setembro de 1985 com a implantação do sistema COPOM (Central de Operações da Polícia Militar).

A Polícia Militar de Pernambuco, visando atender à demanda social pela eficiência e eficácia dos serviços públicos de segurança, fomentando respostas mais objetivas em atendimento a sua destinação constitucional, investiu no setor de informática, consubstanciado na interligação do sistema de comunicação da rede corporativa.

Atualmente, temos o interligamento de alguns serviços através das nossas redes de comunicação, seja na Diretoria de Pessoal, no COPOM, na 2ª Seção ou no CPD, com previsão de expansão em todos os setores do Comando Geral e dos Batalhões da capital e do interior, conforme Projeto da Rede Corporativa em implementação.

No entanto, a utilização desta rede, sem considerar as ameaças relacionadas à segurança, pode causar prejuízos imensos em diversos níveis organizacionais, com a atuação dos "*gangsters virtuais*", conhecidos como "*hackers*", tentando se infiltrar nas redes corporativas para piratear informações. Já testemunhamos centenas de ataques a redes de computadores. Qualquer acesso não-autorizado à rede de uma corporação, por parte de um intruso, geralmente causa mais transtornos, perdas morais e financeiras do que

as pessoas imaginam. Os principais incidentes envolvem a violação de sistemas financeiros para conduzir transações monetárias não-autorizadas; a modificação, a destruição ou o roubo de dados valiosos para organizações governamentais; a interrupção de valiosos serviços; e outros tipos de atividade não-autorizada. No caso da Polícia Militar de Pernambuco, tais incidentes podem se manifestar, por exemplo:

- através da espionagem no Centro de Recrutamento e Seleção de Pessoal, a fim de obter dados sobre concursos ou alterar gabaritos de provas para ingresso ou exclusão de candidatos;
- no Serviço de Informações e Inteligência da Polícia, uma vez que transitam na rede dados das identidades dos agentes de inteligência, banco de dados da criminalidade, cadastro de informantes, credenciamento de instituições públicas/privadas às quais prestam auxílio com os serviços policiais, relações de mandados de prisão, entre outros, os quais, nas mãos de pessoas não qualificadas, trariam grandes prejuízos operacionais à corporação, com conseqüente reflexo na comunidade;
- as senhas podem ser roubadas, quando conectados a outros sistemas; depois disso, a conta poderá ser falsificada;
- as linhas de comunicação podem ser grampeadas e informações secretas da Corporação poderão ser comprometidas;
- sistema poderá ser violado, sendo operações sigilosas seqüestradas;
- a rede poderá ser inundada com informações e acabar entrando em pane.

Várias Corporações e empresas já passaram por incidentes de segurança com muito mais freqüência do que se imagina. As mais cuidadosas não distribuem informações sobre incidentes relacionados à segurança, para evitar sua exposição negativa à imprensa e a acionistas pouco amigáveis: basta, apenas, um incidente sério chegar aos jornais para destruir a boa reputação de um serviço. Portanto, pouquíssimas pessoas ficam sabendo desses incidentes, porque o conhecimento da vulnerabilidade do sistema pode ser mais nefasto do que o próprio crime.

Entendemos ser ainda muito restrita a visão do nosso País acerca destes crimes. Tem se dado alto destaque ao direito à intimidade, deixando-se de lado questões muito

mais complexas. De fato, estamos diante de “crimes” que, a rigor não têm doutrina nem lei, e muito menos jurisprudência, mas que podem lesar interesses sociais muito relevantes.

Assim sendo, a possibilidade de os violadores não serem detectados, processados ou condenados não só destaca a necessidade de a maioria das corporações proteger melhor suas redes contra o acesso não-autorizado , como mostra o alto custo das conseqüências por não fazê-lo.

Neste capítulo, apresentaremos uma fundamentação jurídica da proposta oferecida nesta dissertação, quanto à implantação de um sistema de segurança na rede de computadores da Polícia Militar, tendo em vista a necessidade de um embasamento legal à validade de qualquer projeto na área do serviço público.

5.1 Fundamentação Jurídica

Mesmo para os que não militam na atividade jurídica, é fácil compreender a realidade fática dividida naquilo que é público e no que é privado. Também constitui ponto inarredável a associação do que é público com as coisas do Estado, concebido em suas três esferas, e do que é privado com a iniciativa do particular, aquele que se preocupa com interesses próprios.

O registro inicial dessa clássica dicotomia, ainda atual e, não obstante o passar dos anos, indelével, teve sua gênese formal no Direito Romano. O Jurisconsulto Ulpiano, instado certa feita a pronunciar uma sentença, pontificou que o Direito Público era aquele concernente ao estado dos negócios romanos e o Direito Privado, o que disciplinava os interesses particulares. Apesar dessa assertiva de aparente simplicidade, o enunciado constituiu um marco. Desde então, é possível estabelecer a divisão de vários aspectos de nossa vida em duas vertentes: o público e o privado [2].

Ao chegarmos ao terceiro milênio, e percebendo a velocidade em que as coisas acontecem, não é difícil constatar que muitos problemas enfrentados pelo serviço público

tendem a desaguar na adoção do chamado Estado Mínimo. A rudeza da cibernética e do capitalismo exacerbado anuncia uma redução extrema na máquina estatal. As privatizações são uma realidade. O Estado há de ser eficiente, econômico e moderno, sob pena de ceder, a cada dia, seu campo de atuação ao particular; no entanto, não é possível vislumbrar a falência total do Estado. Com efeito, alguns entes de natureza Jurídica de Direito Público hão de ficar. Mesmo assim, há de se tornar ágil, leve e eficiente. É o Estado Mínimo, formado apenas pelas atividades indisponíveis ao ente privado: a distribuição da justiça "*lato sensu*", o serviço de saúde pública e a segurança pública, dentre outros. Por conseguinte, a Polícia Militar, como parte do Estado, em trabalhos realizados por seus integrantes, ao desempenharem este papel, deverá estar imbuída da necessidade de produtividade, sob pena de nova revisão em sua atuação.

5.1.1 Aspectos Legais do Direito à Informação

Antes de abordar os aspectos legais do direito à informação, necessário se faz esclarecer qual a diferença entre um *hacker* e um *cracker.*

Hackers e Crackers

Os entusiastas da Internet discutiram a diferença entre *hackers* e *crackers* durante muitos anos. A melhor definição do termos *hacker* e *cracker* que encontramos é:

- Um *hacker* é uma pessoa intensamente interessada nas funções misteriosas de qualquer sistema operacional de computador. Os *hackers* são mais freqüentemente programadores. Como tal, os *hackers* têm conhecimento avançado de sistemas operacionais e linguagens de programação. Eles podem descobrir brechas dentro de sistemas e as razões para tais brechas. Os *hackers* constantemente buscam mais conhecimento, compartilham livremente o que eles descobriram e nunca corrompem dados intencionalmente.

§ Um *cracker* é alguém que domina ou de outro modo viola a integridade de um sistema de máquinas remotas com intenção maliciosa. Os *crackers*, tendo adquirido acesso não autorizado, destroem os dados vitais, negam serviço a usuários legítimos ou basicamente causam problemas para seus alvos. Os *crackers* podem ser facilmente identificados porque suas ações são maliciosas [12].

Mens rea

Mens rea é um termo em Latim que se refere a "mente culpada". Ele descreve a condição mental em que existe intenção criminal. Aplicar *mens rea* à equação *hacker-cracker* parece suficientemente simples. Se o suspeito penetrou inconscientemente em um sistema de computador – e fez isso por métodos que qualquer cidadão obediente à lei empregaria naquele momento – não há *mens rea* e portanto nenhum crime. Entretanto, se o suspeito estava ciente de que uma brecha de segurança estava aberta – e ele conscientemente empregou métodos sofisticados para exploração dessa brecha – existe *mens rea* e um crime foi cometido. Por essa medida, pelo menos de um ponto de vista jurídico, o primeiro é usuário de computador inconsciente (possivelmente um *hacker*) e o último é um *cracker*.

Na mente de um promotor, o teste de *mens rea* parece claro, definitivo e infalível e desde que a intenção é freqüentemente um requisito para um indiciamento criminal, é nisso que eles geralmente se baseiam [12].

Nossa "*lex fundamentalis*" de 1988, revela uma indisfarçável preocupação com o acesso às informações. É um reflexo natural da situação excepcional dos governos pós-revolução de 1964, os chamados Governos Militares, em que o acesso às mais simples informações era negado. Via de regra, a Carta Política atual trouxe, em suas cláusulas pétreas, dispositivos garantidores desse acesso [6]. A que vem gizada no inciso XXXIII, art. 5º, é a seguinte:

"Art. 5º - Todos são iguais perante a lei, sem distinção de qualquer natureza, garantindo-se aos brasileiros e aos estrangeiros residentes no País a inviolabilidade do

direito à vida, à liberdade, à igualdade, à segurança e à propriedade, nos termos seguintes:

.....

XXXIII – *Todos têm direito a receber dos órgãos públicos informações de seu interesse particular, ou de interesse coletivo ou geral, que serão prestadas no prazo da lei, sob pena de responsabilidade, ressalvadas aquelas cujo sigilo seja imprescindível à segurança da sociedade e do Estado."*

O acesso é garantido, mas não sem um limitador. Calha, então, a ressalva da referida cláusula. A *segurança da sociedade* é o campo de atuação da Polícia Militar. Resguardada está, em tema de altitude constitucional, *a possibilidade de conferir proteção a algumas informações sem desrespeitar as garantias fundamentais de quaisquer interessados.* Quando a informação requerida possibilitar o comprometimento da segurança da sociedade ou do Estado, será defeso o seu acesso [4].

Caso o interessado persista no objetivo de obter as informações, a Carta Política atual trouxe, como *um dos remédios constitucionais, o "Habeas data"*[6]. É o seguinte o texto:

"Art.5º

.....

LXXII – *Conceder-se-á habeas data:*

a) Para assegurar o conhecimento de informações relativas à pessoa do impetrante, constantes de registros ou bancos de dados de entidades governamentais ou de caráter público;

b) Para retificação de dados, quando não se prefira fazê-lo por processo sigiloso, judicial ou administrativo."

Garante-se, neste particular, o acesso à informação de caráter pessoal. Não encontramos, também, razão pela qual não se devam proteger tais informações para que o acesso a elas seja exclusivo do interessado [3].

Por outro lado, há outro tipo de interessado nas informações dos bancos de dados da Corporação. Diferente do profissional que usa meios lícitos para acessar as informações, existem os já definidos *hackers* e *crackers*, também denominados intrusos ou violadores de rede. São, estes intrusos, pessoas que conseguem acesso não-autorizado e clandestino a sistemas de computadores, especialmente aqueles que fazem isso remotamente [7].

Os órgãos de segurança norte-americanos identificam essas pessoas com características específicas, tais como: solidão, desilusões ou grandes decepções.

Os motivos que revelam para a sua atividade danosa são: ganhos financeiros, vingança, idealismo, curiosidade, busca de emoção, anarquia, aprendizado, ignorância e espionagem.

Naturalmente, uma rede corporativa de informações na Polícia Militar constituiria uma grande tentação para esses intrusos. Dados referentes a movimentos estatísticos, locais de valor operacional, pontos críticos e sensíveis, operações secretas, investigação, segurança de dignitários, dentre outros, constituiriam rico material para suas investidas.

No Brasil, a advogada Maria Helena Junqueira Reis (s.d.) , em trabalho denominado “Crime informático” [5], constata não haver tipificação para esta modalidade de crime. Ainda em seu trabalho, faz referência à revista Exame Informática (14 de junho de 1993, p. 56), cita vários casos de violação de sistemas nas faculdades da Universidade de São Paulo e da PUC-RJ. Outros juristas citam idêntica contaminação na Petrobrás, no Banco Real, na American Express, na Embratel, na White Martins, entre outros, e pontificam que não há interesse de que sejam publicados todos os casos, porque o conhecimento da vulnerabilidade dos sistemas pode ser mais danoso do que o próprio crime [5].

A lei pode não ser o instrumento mais preciso para reverter os prejuízos causados por *hackers* e *crackers*, mas muitas vezes ela pode fazer um serviço adequado. Existe uma necessidade real de alteração do nosso código penal brasileiro face às novas

formas de crimes informáticos, sendo a alternativa atual: obter confissão do invasor; tentar enquadrá-lo em algum crime previsto no código vigente (estelionato, calúnia, violação de correspondência, etc.). A dificuldade está na definição do que é legal, ou não.

A Polícia Militar não pode ficar divorciada do avanço tecnológico mas, ao mesmo tempo, precisa defender-se das ações que possam, comprometer seus sistemas de informações e bancos de dados.

5.1.2 Princípios Constitucionais da Administração Pública e Seus Aplicativos na Segurança das Informações na Rede

Alguns dos Princípios Constitucionais consagrados na ordem jurídica brasileira servem de lastro para a necessidade de um sistema codificado de segurança. Os princípios, de um modo mais genérico, são preceitos de nível supra constitucional que informam o próprio direito objetivo que dela (a Constituição) emanará. Acha-se, então, acima das normas, porque uma norma só será verdadeiramente legítima quando submissa aos princípios gerais do direito. Os princípios administrativos, por sua vez, são postulados fundamentais que inspiram todo o modo de agir da administração pública. Representam cânones pré-normativos, norteando a conduta do Estado no exercício de atividades administrativas.

A Constituição da República enunciou alguns princípios básicos para a Administração Pública. Em face de tal constatação, e diante de outros princípios extra constitucionais que também informam o serviço público, utilizamos a sugestão do professor José dos Santos Carvalho Filho e dividimos os princípios em dois grupos [4]:

Princípios Expressos: aqueles que vêm estampados no art. 37, “caput”, do texto constitucional, a saber :

Princípio da Legalidade;

Princípio da Impessoalidade (ou da Finalidade);

Princípio da Moralidade;

Princípio da Publicidade; e

Princípio da Eficiência (este ausente no texto original da Lei Fundamental e acrescentado pela Emenda Constitucional nº. 19/98).

Princípios Reconhecidos: aqueles amplamente consagrados pelos estudiosos do Direito e percebidos nas intenções de diversas passagens constitucionais ou até fulcrados na doutrina mais autorizada. São eles:

Princípio da Supremacia do Interesse Público sobre o Interesse Privado;

Princípio da Indisponibilidade, pela administração, dos interesses e bens públicos;

Princípio da Razoabilidade;

Princípio da Proporcionalidade;

Princípio da Motivação;

Princípio da Economicidade;

Princípio do Controle Judicial dos Atos Administrativos;

Princípio do Devido Processo Legal e da Ampla Defesa;

Princípio da Responsabilidade do Estado por Atos Administrativos;

Princípio da Autotutela; e

Princípio da Continuidade dos Serviços Públicos.

Do elenco supra, pinçamos alguns princípios que darão sustentáculo jurídico à proposta de implantação do sistema de segurança na rede corporativa da Polícia Militar.

Princípio da Legalidade

O tradicional Princípio da Legalidade informa que o administrador público somente poderá fazer o que estiver expressamente autorizado em lei e nas demais espécies normativas, inexistindo, pois, incidência de sua vontade subjetiva, pois na Administração Pública só é permitido fazer o que a lei autoriza, diferentemente da esfera particular, na qual será permitida a realização de tudo o que a lei não proíba. Com efeito, se a atividade do administrador não for autorizada por lei, é ilícita. Sustenta, neste particular, José dos Santos Carvalho Filho [4] que tal postulado, consagrado após séculos de evolução política,

tem por origem mais próxima a criação do Estado de Direito, ou seja, o Estado que deve respeitar as próprias leis que edita. No dizer de Hely Lopes Meirelles [1], *"enquanto os indivíduos no campo privado podem fazer tudo o que a lei não veda, o administrador público só pode atuar onde a lei autoriza."* É corrente em Direito que a interpretação no ramo administrativo se faz declarativamente, por força do Princípio da Legalidade.

Estabelecer um sistema de proteção à rede corporativa pode significar uma medida crucial na preservação do Princípio da Legalidade, que é imperativo para o administrador.

Princípio da Publicidade versus Princípio da Supremacia do Interesse Público sobre o Interesse Privado

A Corporação Policial pertence à Administração Direta, sendo, portanto, Pessoa Jurídica de Direito Público, regendo-se incontestavelmente pelo Princípio da Publicidade. Por este princípio, os atos da administração devem merecer a mais ampla divulgação possível entre os administrados, para propiciar o exercício do controle sobre a legitimidade da conduta dos agentes administrativos. É a necessidade de transparência no serviço público. Só assim é possível aquilatar a legalidade dos referidos atos. Somente este princípio, salienta Alexandre de Morais [3], evita os dissabores de processos arbitrariamente sigilosos, permitindo os recursos administrativos e judiciais. É assim que se dá com as publicações levadas a efeito pelo Diário Oficial. Ora, se estamos falando em publicidade, falar em informações seladas seria uma incongruência. Desse modo, seria defeso escudar a proteção da rede corporativa em algumas informações de interesse público. Já ficou sobejamente provado que o legislador constituinte excepcionou quanto à publicidade de algumas informações, desde que visem à segurança da sociedade e do Estado. Trata-se de realidade palpável, e não de mera suposição. Ademais, surge, na contramão desta possibilidade de ostensividade de todos os atos geradores das informações, o Princípio da Supremacia do Interesse Público sobre o Interesse Privado. Revela este princípio que as atividades administrativas são desenvolvidas pelo Estado para o benefício da coletividade. Não havendo a supremacia do interesse público, a atuação de qualquer funcionário estará inquinada de desvio de finalidade. O destinatário da atividade

administrativa não é o indivíduo em si, nem determinada classe privilegiada, mas o grupo social como um todo.

O risco de informações privilegiadas caírem em mãos inescrupulosas, o que afetaria o interesse público, legitima a adoção da rede cifrada. José Afonso da Silva [2], em seu Curso de Direito Constitucional Positivo, faz alusão à necessidade de se resguardar informações sigilosas. Alerta, no entanto, Hely Lopes Meirelles que se trata de situação excepcional. Somente as informações que possam causar dano grave, se divulgadas, serão alvo de proteção. Além disso, sempre é possível buscar a tutela judicial pelo Mandado de Segurança ou "Habeas data", nos moldes do Art. 5º da Lei Maior. Ainda assim, pelo *Princípio da Autotutela,* a própria administração pode rever seus próprios atos, sanando qualquer irregularidade ou má interpretação nessa tarefa de proteger certas informações. A tarefa policial, aliada a outras, é pública por excelência. Com efeito, os bens e interesses públicos não pertencem à administração nem aos seus gerentes. É o que informa o *Princípio da Indisponibilidade*. O caráter deste princípio não é só patrimonial, mas também da órbita dos interesses. Entender o interesse público como bem supremo, neste campo, é compreender a impossibilidade que o administrador tem de dispor de certas informações, porque não faz a sua própria vontade. O princípio parte da premissa de que todos os cuidados exigidos para os bens e interesses públicos trazem benefício para a própria coletividade. Enfatizamos o aspecto da indisponibilidade dos interesses porque, neste princípio, há sempre uma preocupação excessiva em se estudar os diversos tipos de bens públicos e suas vedações.

Descreveremos no próximo capítulo a Rede Corporativa da Polícia Militar de Pernambuco.

CAPÍTULO 6

REDE CORPORATIVA DA POLÍCIA MILITAR DE PERNAMBUCO

A Polícia Militar, como parte do Estado, em trabalhos apresentados por seus integrantes, tem procurado otimizar suas ações de apoio para a consecução de um melhor rendimento em sua destinação constitucional. Trata-se de apoio, e não de disfunção. Na seara da tecnologia informática, muitos avanços foram percebidos dentro da Corporação. A implantação do serviço de atendimento a ocorrências no sistema COPOM (Central de Operações da Polícia Militar) foi a gênese. O sistema de telefonia em discagem direta a ramal e a aquisição de computadores modernos para acesso à rede mundial são outra realidade.

Atualmente, mais dois projetos estão em fase de implantação:

- a *rede corporativa*, que consiste em interligar as Unidades Administrativas e Operacionais, viabilizando o acesso mais rápido e preciso às informações;
- as *patrulhas comunitárias*, que consistem na informatização e o monitoramento das viaturas policiais, de modo que se obtenham, com precisão, dados gerais de interesse policial que vão desde a origem de uma arma de fogo até o dossiê completo de um elemento considerado suspeito ou de um meliante.

A rede corporativa, que é o tema de nosso campo de interesse, é um projeto ousado e de utilidade comprovada. Significa construir um banco de dados completo e

atualizado de informações vitais em todas as áreas de trabalho da instituição. Este banco de dados seria suprido a todo instante e teria o escopo de colocar os usuários em contato com a informação em questão de segundos.

Um sistema dessa envergadura, no entanto, carece de um esquema de proteção que dê segurança às informações que devam circular nessa rede. Hoje, quando se deseja obter uma informação na Polícia Militar, utiliza-se o sistema de expediente normal, com atrasos inadmissíveis para a amplitude de certas missões. Também a tramitação de requerimentos administrativos é penosa e, por vezes, insolúvel. Os prazos quase nunca são cumpridos, estabelecendo percalços dos mais diversos níveis. A rede corporativa proporcionará uma fluidez significativa no acesso às informações. Este acesso facilitado contribuirá, sobremaneira, para as atividades na vanguarda do combate à criminalidade. Todos os que estão neste "front", seja a Polícia Militar, a Polícia Civil, o Ministério Público, os Magistrados, serão beneficiados, uma vez que o rápido acesso aos bancos de dados estabelecerá um campo de cooperação que levará à eficiência no processo de combate sistemático ao delito.

Há de se convir, no entanto, que nem todos terão acesso a todas as informações da rede corporativa. O grande desafio será selecionar as informações que podem ser ostensivas e as que devem ser sigilosas, protegendo-as com sistemas de segurança eficazes. Esta proteção, num primeiro momento, dar-se-ía nos pontos infra:

- transmissão da dados;
- acesso aos arquivos;
- leitura e gravação de dados; e
- compartilhamento de periféricos e *softwares*.

A Polícia Militar não pode ficar divorciada do avanço tecnológico, e tem necessidade de defender-se das ações que possam comprometer seus sistemas de informações e bancos de dados.

A proposta formulada nesta dissertação, através da introdução de técnicas criptográficas como uma das medidas adotadas na política de segurança na rede de

computadores da PMPE, exigiu, além do estudo teórico, a coleta de informações no contexto prático, para o qual converge o projeto.

Nesse sentido, foram feitos levantamentos "*in loco*" do estado atual da rede de comunicações da Polícia Militar no âmbito do Quartel do Comando Geral, ou seja, foi verificado, através de observação direta, quantos micros e pontos de rede existiam, quais as aplicações utilizadas, sua arquitetura e o sistema operacional empregado, os quais serão ilustrados adiante através de tabelas e mapas.

Realizamos também um levantamento no Centro de Processamento de Dados acerca do projeto da rede corporativa, com os pontos previstos, a arquitetura e o sistema operacional.

E, por fim, foram colhidas informações, por meio de questionário, junto aos setores do Quartel do Comando Geral, as quais serviram de subsídios na justificativa na implantação de uma política de segurança (Apêndice B).

Entre os instrumentos utilizados, as entrevistas com os Comandantes, Chefes e Diretores de seções foram, de fato, o que desempenhou um papel fundamental em nível de conscientização da existência de violações em ambientes computacionais e seus conseqüentes prejuízos, podendo repercutir operacional, financeira e moralmente na Corporação, afetando sobremaneira os serviços desempenhados pela Polícia Militar.

Será apresentada, no apêndice B, toda a coleta das informações, por intermédio da ilustração de gráficos, com as respectivas observações.

6.1 Estado Atual da Rede de Comunicações da PMPE

A atual rede de computadores da PMPE utiliza três servidores, os quais atendem ao CPD (Centro de Processamento de Dados), à 2 ª seção, à Diretoria de Finanças, à Diretoria de Pessoal e ao COPOM, conforme podemos visualizar da figura 46.

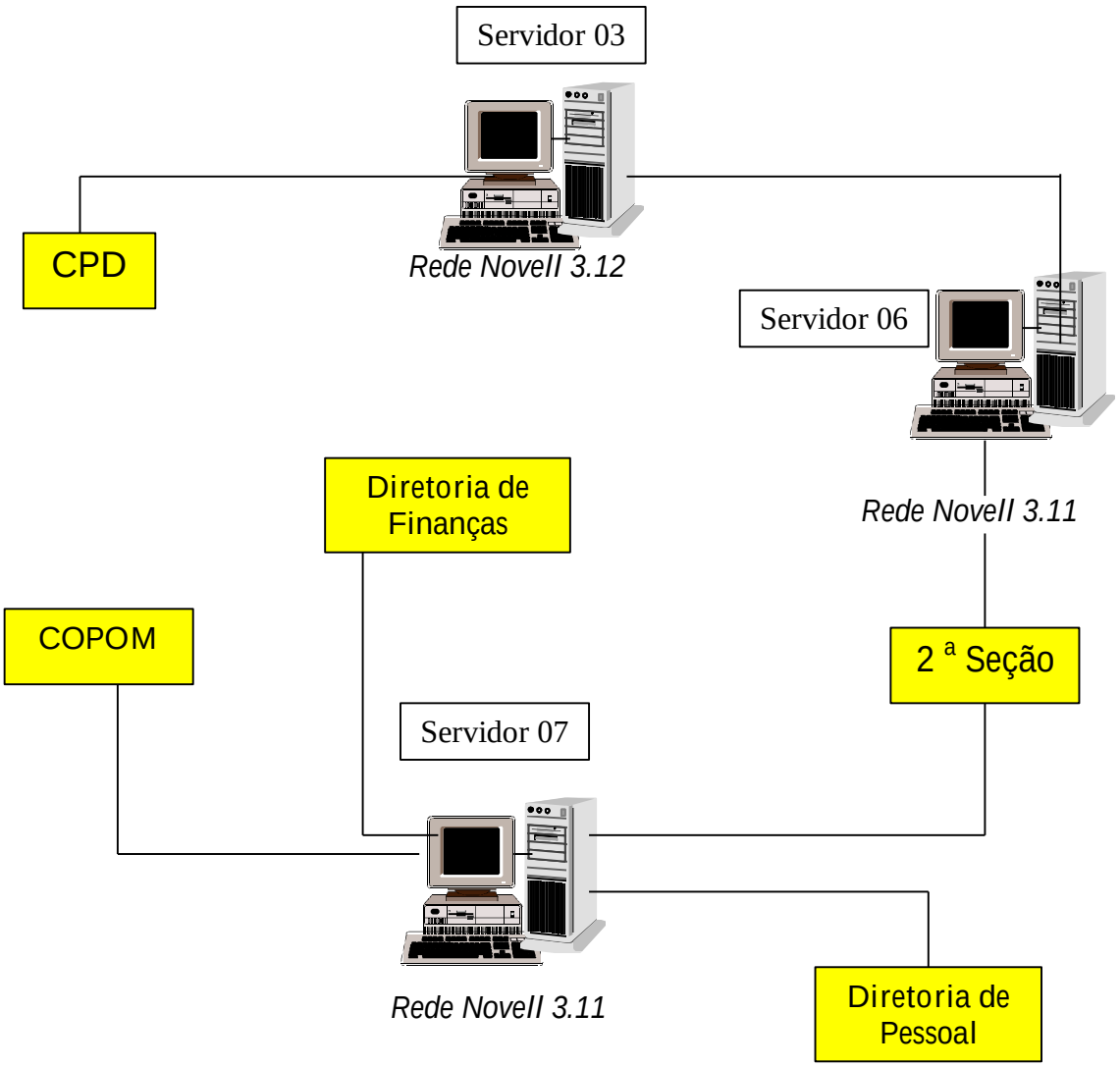


FIGURA 46 - ESTADO ATUAL DA REDE DE COMUNICAÇÕES DA PMPE

6.1.1 Aplicações Utilizadas

Realizamos um levantamento das aplicações utilizadas no ambiente de rede de computadores:

- no COPOM – São desenvolvidas Operações Policiais;
- em Recursos Humanos – Controle de Pessoal;
- na 2 ª Seção (PM-2) – Investigações e Estatística;
- em Telefonia – Controle de Ligações;
- na Diretoria de Finanças – Folha de Pagamento;
- em Controle de Armas – Armamento;
- no Hospital – Marcação de Consultas.

6.1.2 Arquitetura da Rede

Rede ETHERNET, com topologia em barra, caracterizando-se pela implementação do protocolo IPX/SPX, utilizando servidores NetWare, os quais oferecem suporte às estações *diskless* e às estações com Windows.

O protocolo IPX (*Internetwork Packet Exchange*) é o protocolo usado pela Novell para o nível de rede. O IPX fornece um serviço datagrama não confiável a seus usuários (normalmente o SPX), isto é, seus pacotes são transmitidos sem que seja necessário estabelecer conexões e não são reconhecidos pelo destinatário. O IPX implementa um esquema de roteamento inter-redes (todas elas usando o IPX), baseado em tabelas de rotas localizadas nos *gateways.*

O SPX (Sequenced Packet Protocol) é o protocolo usado pela Novell para o nível de transporte do RM-OSI. O SPX implementa um serviço de circuito virtual, ou seja, mecanismos de controle de erro, de fluxo e de seqüenciação [8].

6.2 Projeto da Rede Corporativa da PMPE

A Rede Corporativa a ser implementada terá tecnologia ETHERNET, utilizando como protocolo padrão o TCP/IP, com topologia em estrela baseada em *switches* de alta velocidade, permitindo um "*throughput*" (fluxo) de 10 Mbps "*full*" por porta, na qual o cascateamento entre os *switches* será de 100 Mbps.

A tecnologia ETHERNET se caracteriza pela implementação do protocolo de acesso ao meio CSMA/CD *(Carrier Sense Multiple Acces/Collision Detection), contensivo e não determinístico.*

O Sistema Operacional de Rede (SOR) a ser empregado será o Windows NT. Este SO se caracteriza pela facilidade de operação e pela implementação, bem como pela diversidade de clientes permitida no ambiente de rede (DOS; Windows 3.11, 95, 98, NT Workstation e 2000).

A rede corporativa será conectada ao *backbone* do Estado por meio de um link de 10 Mbps, o que possibilitará a conexão e a troca de informações com todos os Batalhões que tiverem acesso ao mesmo *backbone*.

A rede do QCG contemplará 170 pontos e será formada por um Switch Central de 12 portas a 100 Mbps *full,* instalado no CSM/TEL, o qual permitirá a interconexão de 10 Switches de 24 portas a 10 Mbps *full*, que servirá para conexão das estações de trabalho nos mais diversos setores da PM (figura 47).

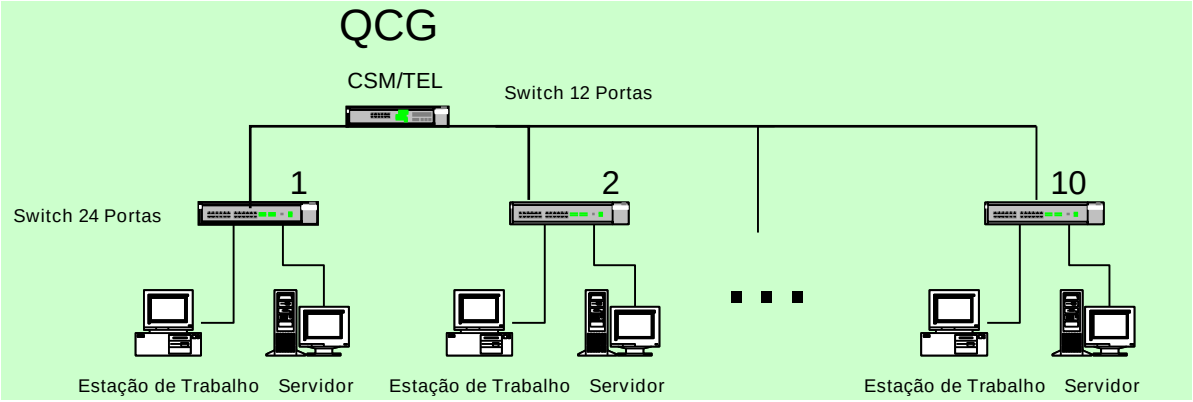


FIGURA 47 – DISTRIBUIÇÃO DA REDE NO QCG

A rede corporativa do QCG será atendida por cinco servidores distribuídos conforme a ilustração da figura 48, onde apresentamos as funções de cada servidor.

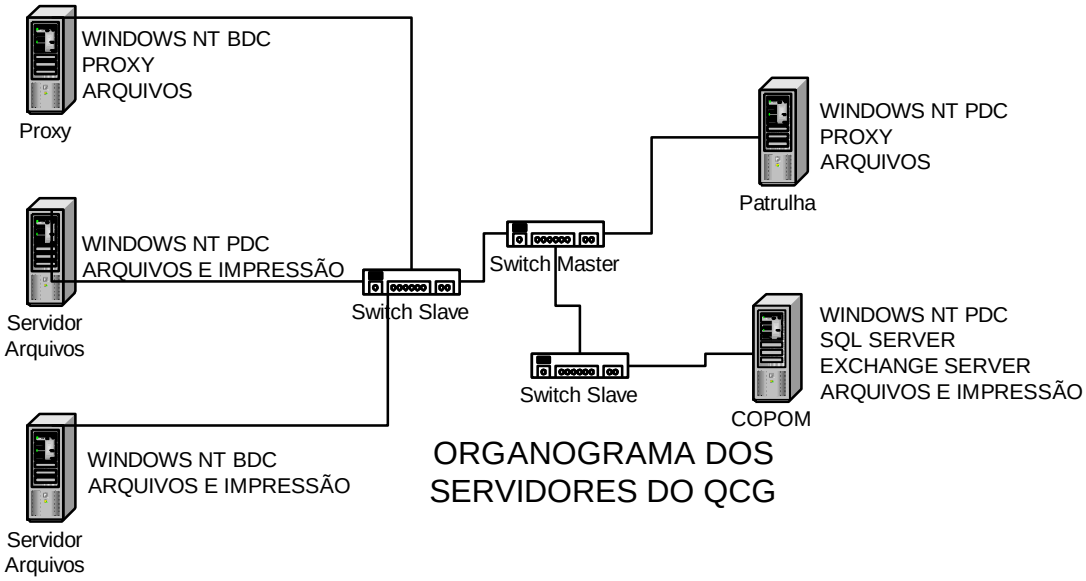
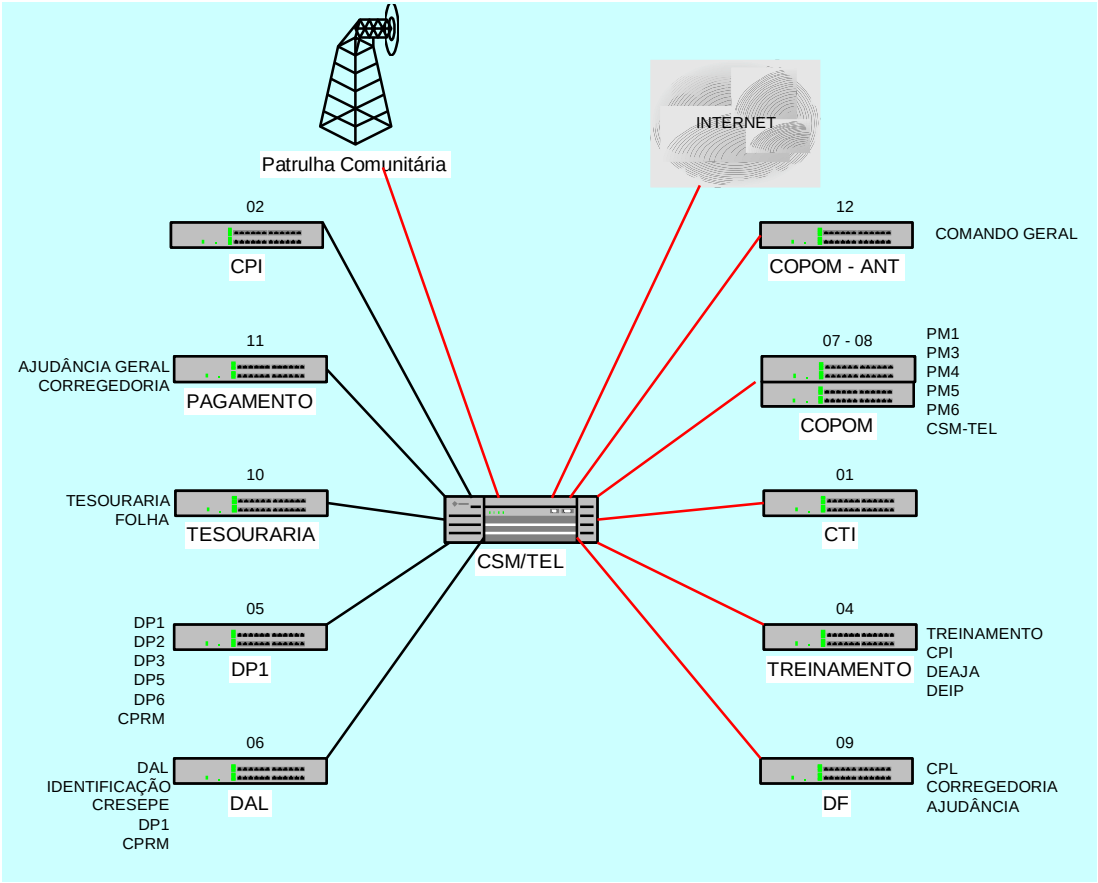


FIGURA 48 – ORGANOGRAMA DOS SERVIDORES DO QCG

6.2.1 Mapa Demonstrativo da Rede

Na figura 49, apresentamos da distribuição geral da rede no Quartel do Comando Geral (QCG), a qual é composta por *switches* cascateados e um *link* formado com as viaturas através das ondas de rádio.

FIGURA 49 – DISTRIBUIÇÃO GERAL DA REDE NO QCG



Na figura 50, apresentamos o esquema formado pelas Patrulhas Comunitárias.

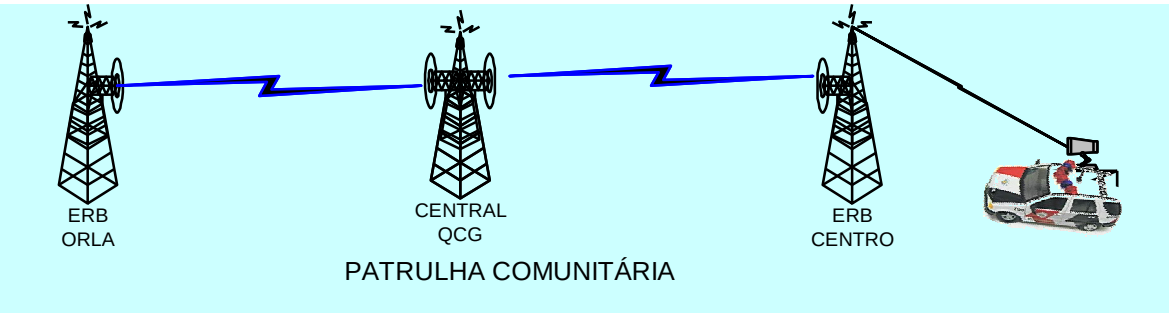


FIGURA 50 – ESQUEMA DA PATRULHA COMUNITÁRIA

O *link* formado entre o QCG e as Patrulhas Comunitárias, conforme apresentado na figura 50, é realizado com equipamentos da Lucent, os quais funcionam por meio de módulos (figura 51), com placas PCMCIA inseridas nos mesmos. Cada placa possui duas entradas, podendo ser Ethernet ou RCA (Antenas).

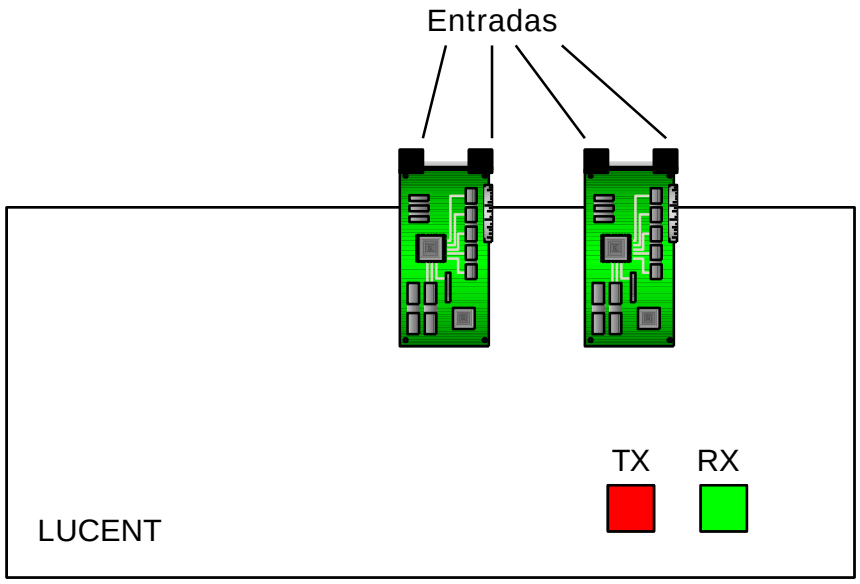


FIGURA 51 – MÓDULO LUCENT COM DUAS PLACAS PCMCIA

A comunicação entre a viatura e uma Estação Rádio Base (ERB) é feita por meio da antena localizada na viatura, a qual deve estar direcionada para a antena ominidirecional da ERB. Esta antena ominidirecional funciona como uma repetidora da antena central localizada no QCG (figura 50).

A comunicação entre uma ERB e a Central do QCG, é realizada por antenas unidirecionais (figura 50).

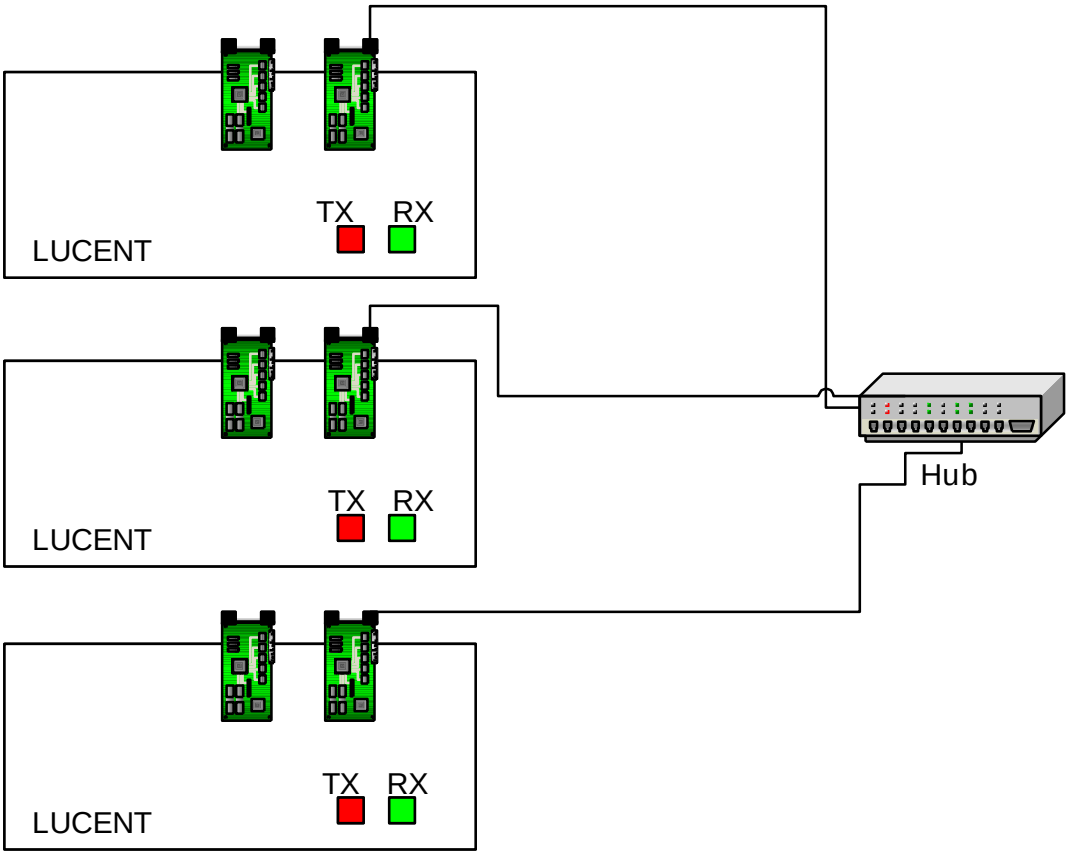
O *link* entre as ERB's e a Central é de 2 Mbps (podendo chegar a 10 Mbps, caso haja *upgrade* (atualização)); e, entre a viatura e a ERB, caso a antena da viatura esteja bem direcionada, o link será de 2 Mbps.

Em cada ERB, há um módulo Lucent, podendo conter uma ou duas placas PCMCIA, o fator que determina a utilização de duas placas, é se a ERB irá funcionar como repetidora para uma outra ERB.

Cada viatura possui um módulo com a placa, na qual uma saída é Ethernet para o microcomputador e a outra é RCA (antena).

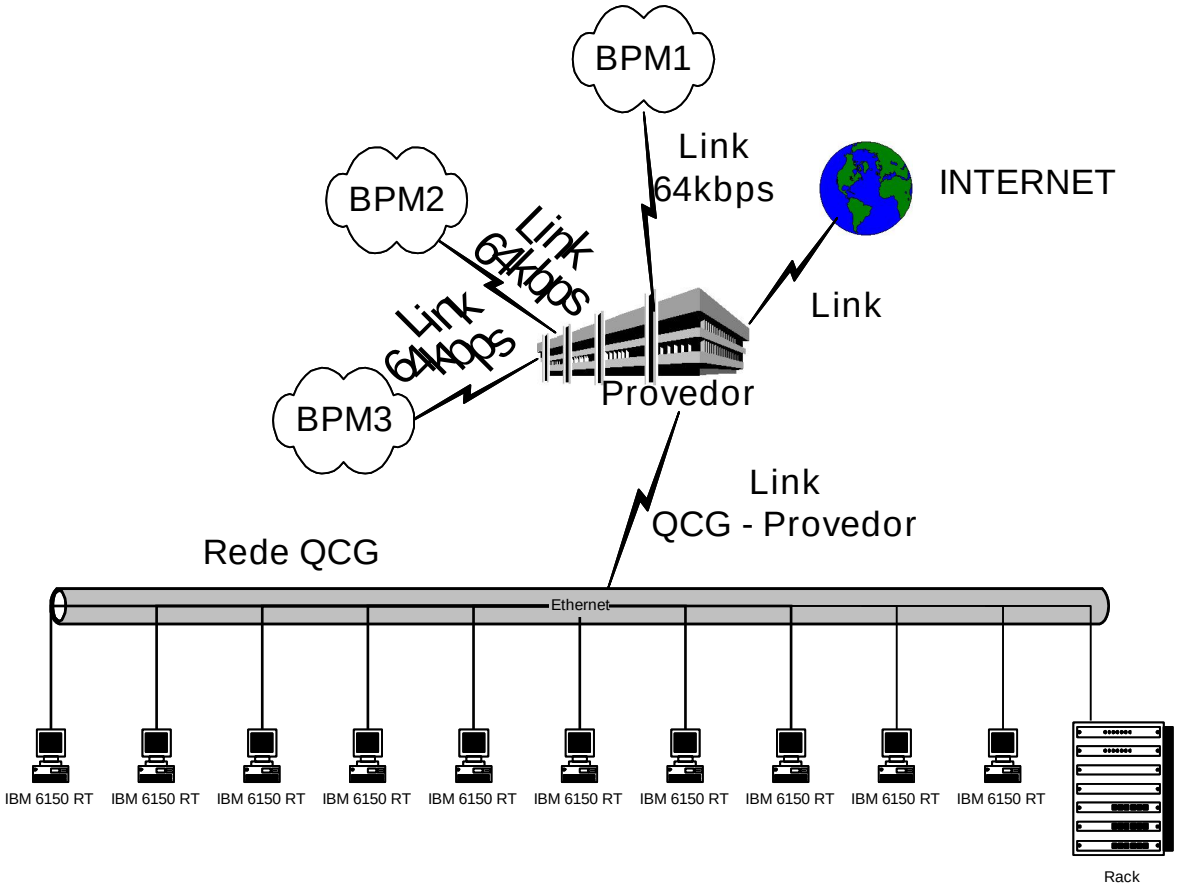
Na Central do QCG, temos o seguinte dispositivo: três módulos cascateados por um *hub* o qual é interligado ao *switch master* do CSM/Tel (figura 52).

FIGURA 52 – REPRESENTAÇÃO DOS MÓDULOS LOCALIZADOS NA CENTRAL DO QCG



Na figura 53, ilustramos o *link* de 10 Mbps em fibra óptica, do QCG ao Provedor do Estado (FISEPE), bem como o *link* do Provedor (FISEPE) aos Batalhões da Polícia Militar.

FIGURA 53 – LINK DO PROVEDOR AO QCG E AOS BATALHÕES



Na figura 54, apresentamos a conexão ao *backbone* dos Batalhões no interior do Estado.

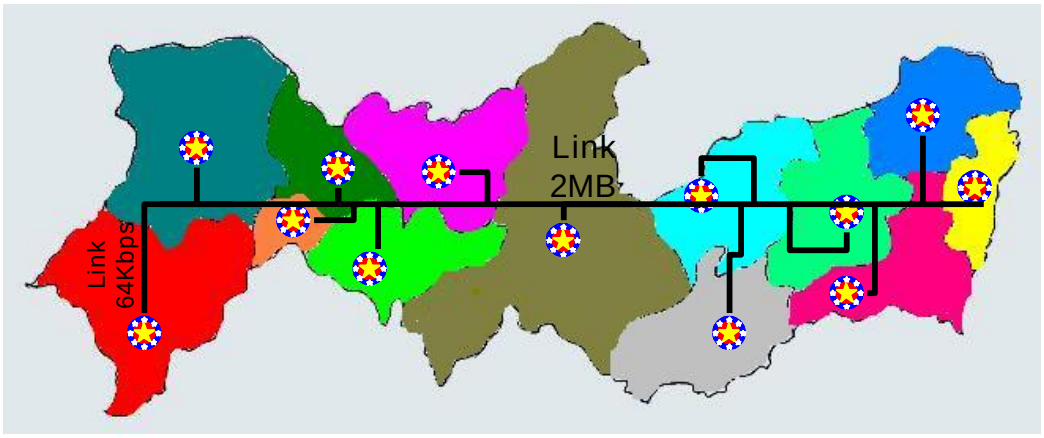


FIGURA 54 – BACKBONE DOS BATALHÕES NO INTERIOR DE PE

Na figura 55, apresentamos a conexão ao backbone dos Batalhões da RMR (Região Metropolitana do Recife).

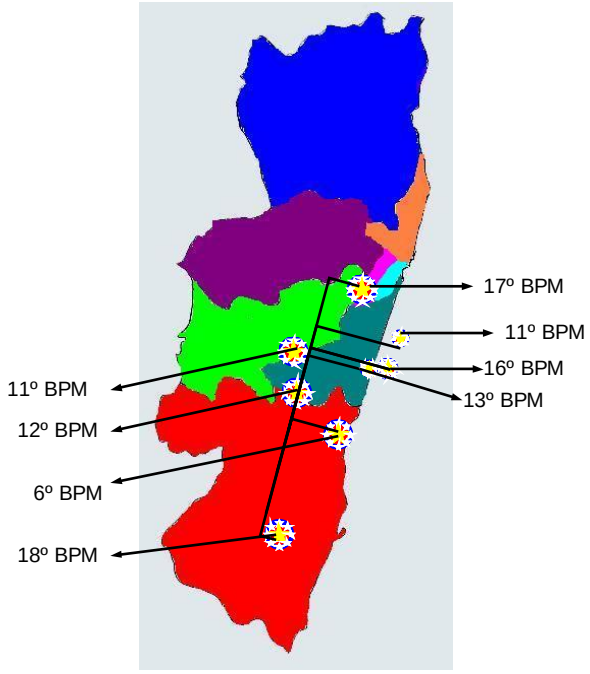


FIGURA 55– BACKBONE DOS BATALHÕES DA RMR

6.3 Análise de Risco

A análise de risco de violação de dados sigilosos os quais transitam na rede, abrange a identificação de como é possível comprometer a confidencialidade dos dados e sistemas e a possibilidade de acesso a eles. Outro objetivo dessa análise é a identificação de outras perdas e de seu provável impacto (em termos de perdas esperadas e da possibilidade de cada uma delas acontecer). Nosso objetivo final é priorizar o uso de recursos na implementação de controles de segurança. Portanto, a análise de risco é primordial nos programas de segurança de informações.

A Rede Corporativa não é exceção a essa regra. Para gerenciar a segurança na rede de forma inteligente, primeiro precisamos avaliar o que poderemos encontrar. Desta feita, foi realizada uma pesquisa direcionada aos gestores da polícia no âmbito do QCG, colhendo informações, a fim de avaliar o nível de conhecimento a respeito da problemática da violação do sigilo e dos crimes informáticos, verificando a incidência e a probabilidade de ocorrência de quebra de sigilo nos mais diversos setores e serviços executados pela polícia, quais as informações que deveriam ser salvaguardadas, que tipos de informações,

que pessoas poderiam se interessar por essas informações e qual a repercussão do vazamento desses dados na Corporação.

O resultado obtido dessa análise foi de natureza qualitativa, ou seja, não expressamos os prejuízos causados pela quebra de sigilo em termos financeiros. A análise obtida correspondeu à expectativa, ou seja, verificamos a existência do conhecimento de grande parte dos entrevistados acerca da problemática da quebra de sigilo em ambientes computacionais, havendo uma conscientização da necessidade de uma política de segurança, principalmente após a implementação da rede corporativa que possibilitará o acesso a informações em todos os setores da PM, e houve um consenso geral quanto à repercussão negativa sobre a imagem da Corporação.

Na apêndice B, apresentamos por meio de gráficos, a tabulação da pesquisa realizada, confirmando as deduções acima explanadas, com os correspondentes comentários sobre os resultados obtidos.

CONCLUSÃO

A segurança em uma rede de computadores está relacionada à necessidade de proteção dos dados contra a leitura escrita ou qualquer tipo de manipulação, intencional ou não, confidencial ou não, e a utilização não autorizada do computador e de seus periféricos.

A utilização de computadores é hoje essencial em qualquer atividade, não só pela sua versatilidade que permite o seu uso em um sem-número de tarefas, mas também por poderem ser utilizados para transferir informação (sejam eles voz, dados ou mesmo imagem) a longa distância com rapidez e eficiência. O aparecimento das redes corporativas e, em especial da Internet, surgiu como resposta a esta crescente necessidade de interligação e de intercâmbio.

E como não poderia deixar de ser, a Polícia Militar de Pernambuco investiu neste setor visando acompanhar o desenvolvimento tecnológico. Com o projeto da rede corporativa em andamento, vislumbra-se uma verdadeira mudança em termos de facilidade e fluidez nos serviços executados pelos diversos escalões da polícia. Por outro lado, surge um novo problema: a proteção dos dados sigilosos que tramitam pela rede. É preciso não se descuidar das questões relacionadas com a negligência do sigilo das informações e a falta de um controle do acesso a arquivos dentro da rede. A implementação de garantias e controles de segurança adequados pode dificultar extremamente a ação de pessoas inescrupulosas, logo garantindo a salvaguarda de assuntos altamente confidenciais e sigilosos.

A pretensão de qualificar a rede de informações da Polícia Militar com um sistema de proteção tem viabilidade. Os motivos são mais que justificáveis em face dos riscos oferecidos na ausência de uma proteção das informações disponíveis em rede. Mas uma das preocupações do administrador deve ser o atendimento aos princípios e normas constitucionais, bem como à legislação subalterna. O exercício da cidadania implica a

constante busca do cidadão pelos seus direitos, o que tem praticado o povo brasileiro, antes tão divorciado da tutela jurisdicional, em virtude da desinformação. O que queremos dizer é que a atividade pública deve ser realizada com toda a segurança jurídica, para não violar o direito alheio. Quando se fazem restrições, esta vigília à preservação da legalidade recrudesce ainda mais. O sistema de segurança na rede corporativa tem apoio no ordenamento jurídico vigente, porque, nesse âmbito, antes de qualquer enunciado prescritivo, está o interesse público, como declara a Carta Constitucional: "todo poder emana do povo."

Por ser a Polícia Militar um órgão destinado à segurança da integridade das pessoas e de seus patrimônios, somos candidatos adequados para implementar serviços que garantam confiabilidade por parte da comunidade. Um desses novos serviços é a proteção em rede por meio de técnicas criptográficas, sendo esta uma das medidas de segurança sugeridas nesta dissertação, com a implementação do SAFER, que é um algoritmo de chave-secreta, desenvolvido para a Cylink Corporation (Sunnyvale, CA,USA) pelo Prof. J. L. Massey, a qual abre mão de todos os seus direitos proprietários e consigna este algoritmo ao domínio público.

Aliada a esta medida de segurança, por meio da implementação do SAFER, desenvolvemos um Sistema de Gerenciamento de Chaves em Visual Basic, baseado na técnica de Diffie-Hellman, que possibilita o compartilhamento de chaves secretas entre o transmissor e o receptor em uma rede, sem a necessidade de um canal seguro para a troca de chaves secretas. O sistema fornece um modo público de distribuir chaves secretas.

Vale salientar, contudo, que o trabalho não se completa com a implantação da técnica criptográfica sugerida. É preciso observar outros procedimentos de segurança em redes de computadores, tais como: firewall, assinatura digital, autenticação, segurança física e de pessoal, controle de roteamento, registro de eventos etc.

A implementação de um programa de segurança requer o uso de uma equipe maior do que a necessária para empregar a ferramenta de criptografia, justificando-se a contratação de mais especialistas nessa área. O pessoal extra necessário dependerá muito da extensão das necessidades levantadas para administrar as soluções de segurança, assim

como a utilização de consultores externos para preencher necessidades temporárias e assegurar a existência de características atualizadas no programa de segurança, caso se torne necessário.

O aspecto criptográfico abordado neste trabalho deve ser sempre atualizado devido às constantes quebras de cifras, aos avanços científicos e tecnológicos.

Tudo isso não deve se tornar um trabalho isolado. É muito importante que a Corporação assuma um compromisso de continuidade e dinamismo na implementação de ferramentas, treinamentos e avanços tecnológicos, a fim de não recair no marasmo tecnológico causado por acomodação e deficiência em atualizações.

Todo esse processo de controle é importante para a segurança. O sucesso dos serviços desempenhados pela Polícia Militar, na verdade, pode ser medido por meio da eficiência e da efetividade dos recursos de que dispõe e como são empregados. Esperamos, assim, contribuir para a conscientização de uma Polícia cada vez melhor.

APÊNDICE A

ASPECTOS JURÍDICOS DA SEGURANÇA EM INFORMÁTICA EM OUTROS PAÍSES

Estados Unidos da América X Robert Tappan Morris

O incidente do *worm* da Internet (ou, como veio a ser conhecido, o Morris Worm) mudou para sempre as atitudes para com ataques na Internet. Em 1988 Robert T. Morris, por ironia filho de um especialista em segurança do M.I.T., escreveu um pequeno programa ("*worm*", ou verme, um tipo de programa que vai se multiplicando) que ia se espalhando por toda a rede, nó a nó, por meio de bugs nos serviços finger e mail. Como a cada máquina infectada, todas as que estavam ligadas a ela eram (re)infectadas, o que levou a um eventual colapso da rede em poucas horas. Como resposta a futuros problemas, foi criado o CERT (Computer Emergency Response Team, ou Equipe de resposta a Emergência de Computadores), uma equipe distribuída de especialistas acionados em caso de problemas de segurança. O CERT também distribui alertas em caso de falhas gerais de segurança e mantém um histórico dos casos apresentados. Essa mudança não foi gradual. Para a comunidade Internet preocupada com segurança, houve uma espécie de vingança na condenação de Morris. Contudo, a decisão final naquele caso teve algumas implicações perturbadoras para *hackers* e *crackers*.

O governo tomou a posição de que Morris tinha violado a seção 2 (d) do *Computer Fraud and Abuse Act* de 1986. Essa lei visava uma certa classe de indivíduos:

...qualquer pessoa que intencionalmente acesse sem autorização uma categoria de computadores conhecida como “computadores de interesse federal”, e cause danos ou evite a utilização autorizada das informações em tais computadores, causando perdas de U$ 1.000 ou mais ...

EUA - Califórnia

A Califórnia é a capital do crime e da fraude de computador no mundo. Com isso, esse estado institui algumas leis muito definidas com relação à invasão de computador.

O estatuto é abrangente. Basicamente identifica uma lista de atividades envolvidas por sua definição, incluindo, mas não limitado a, qualquer ação não autorizada que implique invasão, exclusão, alteração, roubo, cópia, visualização ou outro tipo de manipulação dos dados. O estatuto chega até mesmo a diretamente abordar a questão da negação de serviço.

As penas são as seguintes:

- Para acesso não autorizado simples que não implique danos superiores a US$ 400, uma multa de US$ 5.000, um ano de prisão ou ambos.
- Para acesso não autorizado que implique danos reais superiores a US$ 400, uma multa de US$ 5.000 e/ou prisão por 16 meses, 2 anos, ou 3 anos em prisão estadual, ou 1 ano em uma prisão municipal.

Ainda prevê que o pai de uma criança *cracker,* no estado da Califórnia, é quem sofrerá as penalidades civis.

Outros Estados (EUA)

A maioria dos outros estados têm leis quase idênticas. As informações sobre estatutos de crime informatizado podem ser obtidos na Electronic Frontier Foundation (EFF). A EFF mantém uma lista de leis relacionadas com o crime informatizado para cada estado. O site da Web da EFF encontra-se em http://www.eff.org/pub/Privacy/Security/Hacking_cracking_phreaking/Legal/comp_crime_us_state.laws.

A geração anterior de *crackers* não era destrutiva. De fato, eles representavam aborrecimentos e, ademais, os serviços telefônicos freqüentemente eram roubados, mas danos eram um resultado raro. Ao contrário, a nova geração de *crackers* é altamente destrutiva. Essa mudança no caráter do *cracker* moderno sem dúvida desencadeará sentenças mais rígidas no futuro. Forças sociais e econômicas também contribuirão para essa mudança. Como a rede está sendo utilizada para operações bancárias, provavelmente o judiciário adotará uma política mais dura contra *crackers*. Contudo, algo diz que as sentenças norte-americanas sempre permanecerão mais brandas que a de outros países – como a China, por exemplo.

China

A China tem uma atitude bem mais severa para com *hackers* e *crackers.* Por exemplo, em 1992 a Associated Press relatou que o chinês Shi Biao tinha conseguido "craquear" um banco. Ele fugiu com cerca de US$ 192.000 mas foi posteriormente preso e condenado. Sua sentença? A morte. O Sr. Biao foi executado em abril de 1993.

Os recursos mais interessantes sobre legislação chinesa expressamente relacionada com a Internet podem ser encontrados no *Provisional Regulation on the Global Connection via Computer Informationa Network by the People's Republic of China* (Regulamento Provisório sobre a Conexão Global via Rede de Informações de Computador pelo Povo da República da China), com site na Web http://www.smn.co.jp/topics/0087p01e.html.

Neste regulamento, os chineses pretendem controlar todo o tráfego de comunicações com o exterior e colocar certas restrições sobre como as empresas podem se conectar à Internet. O governo chinês afirma que pretende erigir uma nova Grande Muralha da China para deter a Internet ocidental. A China não está só em sua aplicação da política totalitária à Internet e para computadores em geral. Examinemos a visão da Rússia.

Rússia

O presidente Yeltsin promulgou o Decreto 334 de 3 de abril de 1995. Esse decreto concedeu poderes extraordinários ao Órgão Federal de Comunicações e Informações do Governo (OFCIG). O decreto proíbe

... dentro dos sistemas de telecomunicações e informações de órgãos governamentais e de empresas a utilização de dispositivos de codificação, incluindo métodos de criptografia para assegurar a autenticidade das informações (assinatura eletrônica) e meios seguros para armazenar, tratar e transmitir informações ...

A única maneira como esses dispositivos podem ser utilizados é sob revisão, recomendação e aprovação do OFCIG. O decreto também proíbe que

... pessoas jurídicas e físicas projetem, fabriquem, vendam e utilizem meios de informações e também meios seguros de armazenar, tratar e transmitir informações e que realizem serviços na área de codificação de informações, sem licença do OFCIG.

Nos termos mais estritos, portanto, nenhum cidadão russo pode projetar ou vender *software* sem licença desse órgão federal, que age como uma política de informações. Fontes norte-americanas de inteligência identificaram semelhanças entre OFCIG e a NSA (*National Security Agency*).

Comunidade Econômica Européia (CEE)

É interessante notar que *crackers* e *hackers* europeus freqüentemente têm motivações diferentes para suas atividades. Especificamente, *crackers* e *hackers* europeus

tendem a ser politicamente motivados. Uma análise interessante desse fenômeno foi feita por Kent Anderson em seu artigo "*International Intrusions: Motives and Patterns*:"

Um exame detalhado da motivação por trás das invasões revela várias diferenças internacionais importantes: na Europa, grupos organizados freqüentemente têm um motivo político ou ambiental, enquanto nos Estados Unidos uma atitude mais "*anti-establishment*" é comum, bem como simples vandalismo. Nos últimos anos, parece haver um crescimento da espionagem industrial na Europa enquanto os Estados Unidos estão assistindo a um aumento nos motivos criminosos (fraude).

Este artigo pode ser encontrado na Web em
http://www.aracnet.com/~kea/Papers/paper.shtml.

Por essas razões, o tratamento das atividades de *crackers* e *hackers* de Internet na Europa é bem diferente daquele nos Estados Unidos. Um caso recente na Itália demonstra claramente que enquanto a liberdade de expressão é um fato estabelecido nos Estados unidos, nem sempre esse é o caso na Europa.

Foi noticiado que um BBS na Itália, que fornecia acesso de *gateway* à Internet, foi atacado em fevereiro de 1995. Os proprietários e os operadores desse serviço foram subseqüentemente indiciados por alguns crimes relativamente sérios, como discutido por Stanton McCandlish em "*ScotIsand and Italy Crack Down on Anarchy Files*":

... os indivíduos que sofreram o ataque foram formalmente indiciados pelos crimes de subversão terrorista, o que é punido severamente: 7-15 anos de prisão ... BITS BBS [o alvo] carregou um índice de arquivos de materiais disponíveis no repositório *Spunk* (um BBS *underground*), (embora não os próprios arquivos), bem como exemplares passados de *Computer Underground Digest* (do qual o próprio EFF é o principal *site* repositório de arquivos) e outros materiais (não *software*) na forma de textos políticos e não-políticos.

Este artigo pode ser encontrado em

http://www.eff.org/pub/Legal/Foreign_and_local/UK/Cases/BITS-A-t-E_Spunk/eff_raids.article

Reino Unido

Embora o Reino Unido seja de fato membro da União Européia, essa nação será tratada separadamente. O Reino Unido também já teve sua cota de *crackers* e *hackers* de computador; o corpo principal da legislação britânica que proíbe a atividade de *cracker* (amplamente com base na seção 3 (1) do *Computer Misuse Act* de 1990) é de fato bem conciso. Ainda assim, abrange quase todos os atos imagináveis que possam ser empreendidos por um *cracker*. A redação diz o seguinte (o texto foi extraído de um artigo de Yaman Akdeniz):

> Uma pessoa é culpada de uma ofensa se
> (a) cometer qualquer ato que cause uma modificação não autorizada do conteúdo de qualquer computador e;
> (b) no momento em que cometeu o ato ele tinha a intenção e o conhecimento como requisitos.

Observa-se que a intenção é um requisito aqui. Portanto, realizar uma modificação não autorizada deve ser acompanhado da intenção. Um caso é citado sob esse ato contra um indivíduo chamado Christopher Pile (também conhecido como Black Baron), que supostamente lançou um vírus sobre uma série de redes. Pile foi indiciado (e por fim condenado) por ilegalmente acessar e danificar sistemas e dados de computador. A sentença foi de 18 meses, proferida em novembro de 1995. Pile é o primeiro autor de vírus condenado precisamente por esse ato que se tem notícia.

Finlândia

A Finlândia é tradicionalmente conhecida como muito democrática em sua aplicação de leis de informática. A Finlândia fez tentativas para manter uma posição liberal ou quase neutra no que diz respeito à espionagem não autorizada e à atividade de *crackers*

e *hackers*. Mas não mais. Considere essa declaração, extraída do artigo "*Finland Considering Computer Virus Bill"* de Sami Kuusela":

Os legisladores finlandeses aprovarão uma lei nas próximas duas semanas que tornará crime a disseminação de vírus de computador – apesar do fato de tantos vírus serem distribuídos acidentalmente – o que significa que se alguém na Finlândia trouxer um disquete contaminado para seu local de trabalho e não o verificar com um programa antivírus e o vírus se espalhar na rede, a pessoa terá cometido um crime. Também será considerado crime se o vírus se espalhar a partir de um arquivo descarregado da Internet. http://www.wired.com/news/politics/story/2315.html

Neste ponto, observa-se que a tendência (em todos os países e jurisdições) caminha principalmente na direção da proteção dos dados. Recentemente, esboços de leis nesse sentido foram transformados em propostas na Suíça, Reino Unido e EUA.

Acredita-se que essa tendência continuará e isso indica que a legislação da informática amadureceu. Agora que se confrontam com *hackers* e *crackers* por todo o globo, esses governos formaram um tipo de triagem com relação à Internet e às leis de informática em geral. Neste momento, quase todas as novas leis parecem ser projetadas para proteger dados[12].

APÊNDICE B

QUESTIONÁRIO PARA LEVANTAMENTO DE DADOS SOBRE SEGURANÇA NO QCG

Este questionário, foi destinado apenas aos Comandantes, Diretores e Chefes de Seções, que serviu de subsídio à análise dos dados, que versa sobre segurança em rede de computadores na Polícia Militar de Pernambuco.

Questionário

Mesmo que sua seção não tenha ainda, acesso à rede de computadores, de muito servirá sua participação, tendo em vista futuro projeto em implementação, servindo de auxílio na avaliação dos riscos pertinentes às informações transitadas em rede na polícia militar.

Favor informar sua função e seção correspondente:

_____

1) Tem conhecimento de alguma invasão ou quebra de sigilo em ambiente de computadores?

☐ SIM ☐ NÃO

2) Em seu comando, diretoria ou seção, transita alguma informação que poderia ser desejada por pessoas não autorizadas?

☐ SIM ☐ NÃO

3) Favor citar que tipo de informações seria objeto de interesse de pessoas não autorizadas?

_____

_____

4) Qual o nível de importância do sigilo das informações em sua seção?

☐ ALTO ☐ MÉDIO ☐ BAIXO

5) Sua seção já foi vítima de alguma invasão ou violação do sigilo das informações transitadas?

☐ SIM ☐ NÃO

6) Ao seu ver, qual seria a repercussão à Corporação, em caso de violação da integridade das informações sigilosas?

☐ DESCRÉDITO DE OPERAÇÕES

☐ EXPOSIÇÃO NEGATIVA À IMPRENSA

☐ PERDA DE REPUTAÇÃO E CONFIANÇA

☐ OUTROS. CITAR: _____

7) Como seriam classificados os possíveis intrusos interessados nas informações de sua seção?

☐ OS PRÓPRIOS FUNCIONÁRIOS.

☐ PROFISSIONAIS DE OUTRAS ÁREAS

☐ AGENTES EXTERNOS

☐ OUTROS.CITAR: _____

8) Que motivos levariam os intrusos a violarem informações na PM?

- [ ] GANHOS FINANCEIROS
- [ ] VINGANÇA
- [ ] NECESSIDADE DE ACEITAÇÃO OU RESPEITO
- [ ] IDEALISMO
- [ ] CURIOSIDADE OU BUSCA DE EMOÇÃO
- [ ] ESPIONAGEM
- [ ] OUTROS. CITAR:_____

9) Qual o seu procedimento para resguardar informações privadas e sigilosas?

- [ ] Relaciona pessoas autorizadas ao acesso de informações.
- [ ] Controla a saída e entrada de documentos.
- [ ] Utiliza arquivos devidamente fechados.
- [ ] Leva para casa.
- [ ] Outros. Citar: _____

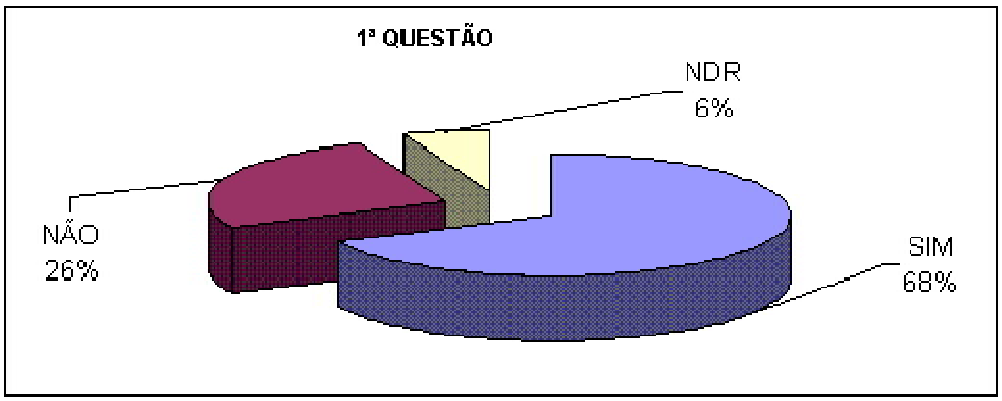
10) Caso utilize rede de computadores em sua seção, que recursos utiliza para evitar a quebra do sigilo?

- [ ] CRIPTOGRAFIA E ASSINATURA DIGITAL
- [ ] SEGURANÇA DE HOST
- [ ] AUTENTICAÇÃO
- [ ] OUTROS. CITAR: _____

Gráfico 1

CONHECIMENTO DE QUEBRA DE SIGILO EM AMBIENTE DE COMPUTADORES PELOS GESTORES NO QCG

SIM	NÃO	NDR	TOTAL
23	9	2	34

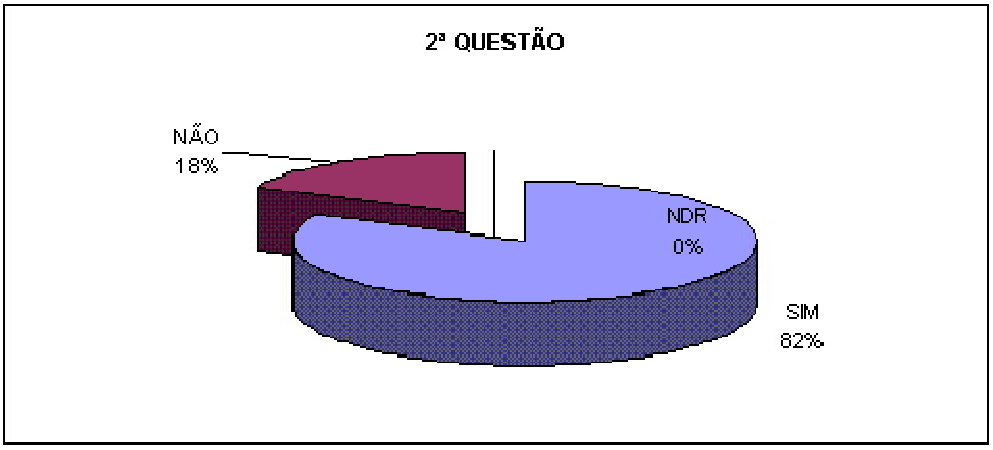


O gráfico mostra o nível de conscientização dos gestores a respeito da problemática quebra de sigilo em ambiente de computadores.

Gráfico 2

TRÂNSITO DE INFORMAÇÕES SIGILOSAS NO ÂMBITO DO QCG

SIM	NÃO	NDR	TOTAL
28	6	0	34

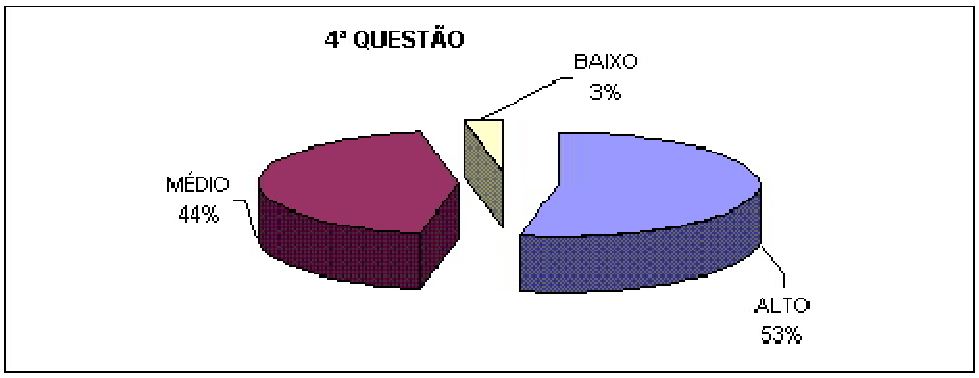


Instados a responder sobre o trânsito de informações sigilosas, as colocações dos pesquisados confirmam as nossas preocupações em salvaguardar os dados ou as informações manipuladas pelos diversos escalões da PMPE.

Gráfico 3

NÍVEL DO SIGILO DAS INFORMAÇÕES TRANSITADAS NO QCG

ALTO	MÉDIO	BAIXO	TOTAL
18	15	1	34

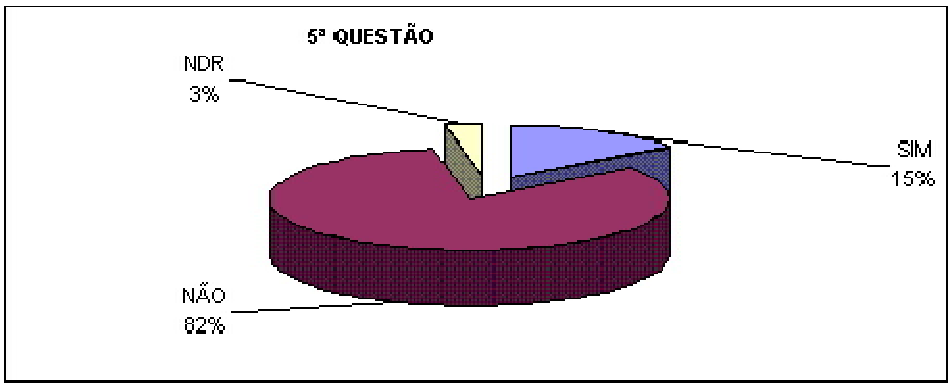


Percebe-se, pelo gráfico, a existência de um percentual elevado no nível de sigilo das informações transitadas no QCG.

Gráfico 4

OCORRÊNCIA DE VIOLAÇÃO DO SIGILO DAS INFORMAÇÕES

SIM	NÃO	NDR	TOTAL
5	28	1	34

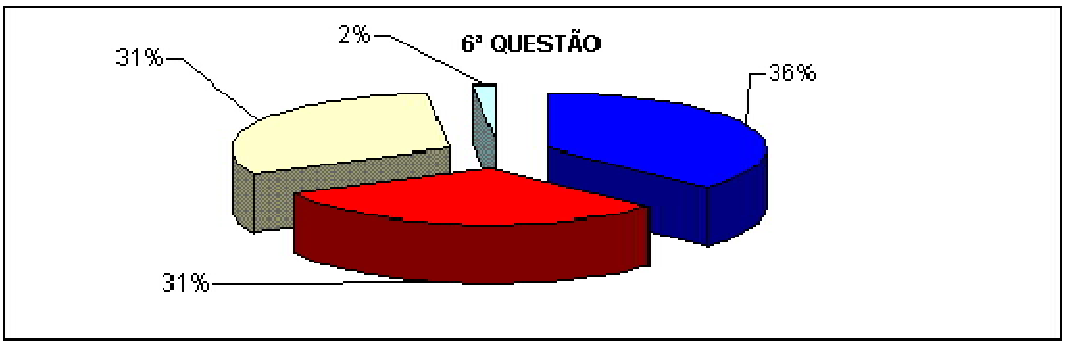


Pelo levantamento realizado através do gráfico se constata que, apesar de muitos setores não serem ainda contemplados pela rede de computadores, já há registros de violação das informações. Esse fato é preocupante, tendo em vista que, mesmo com a implementação da Rede Corporativa, o que em muito viabilizará os serviços, o sistema ainda estará vulnerável a invasões eletrônicas de *hackers*.

Gráfico 5

REPERCUSSÃO NA CORPORAÇÃO NOS CASOS DE VIOLAÇÃO DE INFORMAÇÕES SIGILOSAS

D.Op	Exp.Neg.Imp	Ped.Rep.Conf.	Outros
24	20	20	1



Legenda:

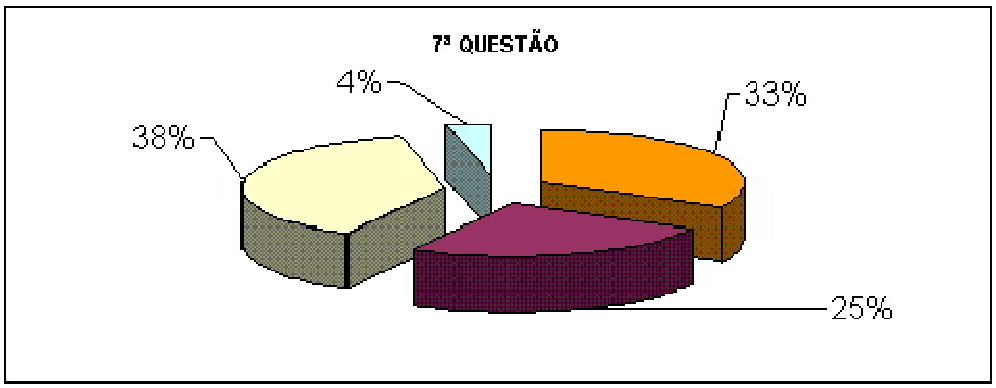
- Descrédito de Operações
- Exposição Negativa à Imprensa
- Perda de Reputação e Confiança
- Outros

Pelo levantamento realizado através da pesquisa, ficou clara a relevância da grande repercussão negativa sobre a imagem da Corporação mediante a possibilidade de vazamento de dados e informações sigilosas.

Gráfico 6

CLASSIFICAÇÃO DE INTRUSOS

O P F	P O A	A E	OT
18	14	21	2



Legenda:

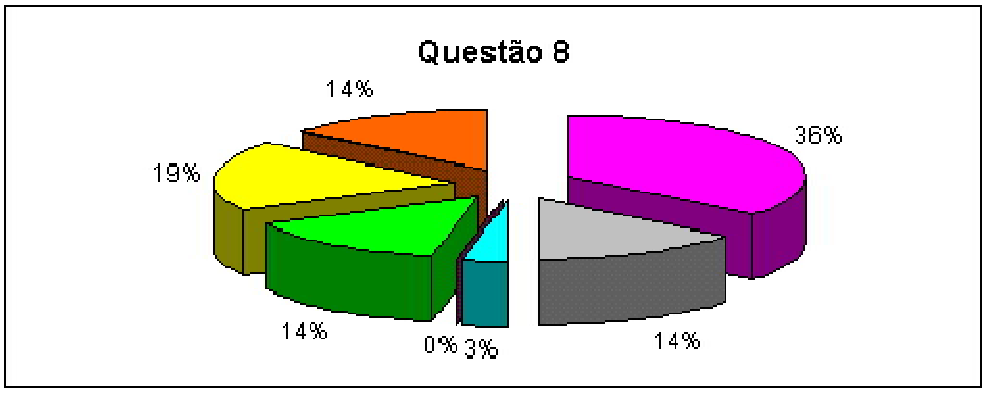
- Os Próprios Funcionários
- Profissionais de Outras Áreas
- Agentes Externos
- Outros

Na análise do resultado da pesquisa, vimos a diversidade potencial dos tipos de pessoas que poderiam se interessar pelos assuntos que circulam pelas seções do QCG, demonstrando claramente a necessidade de resguardar das informações confidenciais.

Gráfico 7

MOTIVOS DE VIOLAÇÕES DE INFORMAÇÕES

GF	VING.	NEC.	IDEAL.	CURIOS.	ESP.	OUTROS
23	9	2	0	9	12	9



Legenda:

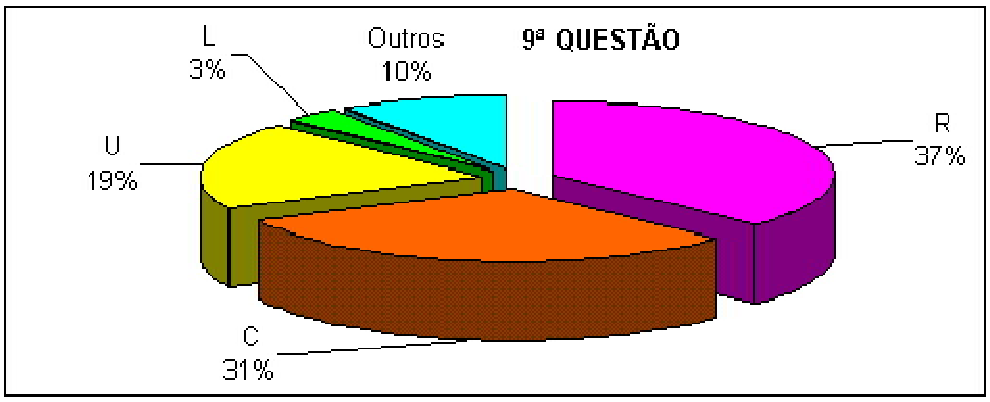
- Ganhos Financeiros
- Vingança
- Necessidade de Aceitação ou Respeito
- Idealismo
- Curiosidade ou Busca de Emoção
- Espionagem
- Outros

O gráfico ilustra um campo bastante diversificado de motivos que levariam as pessoas a violar o sistema de informações da PM, dentre os quais destacamos os ganhos financeiros como o principal motivo de interesse da quebra de sigilo.

Gráfico 8

MEDIDAS ADOTADAS NA SALVAGUARDA DE DOCUMENTOS CONFIDENCIAIS

R	C	U	L	Outros
23	19	12	2	6



Legenda :

- Relaciona Pessoas Autorizadas ao acesso a informações
- Controla a saída e a entrada de documentos
- Utiliza arquivos devidamente fechados
- Leva documentos para casa
- Outros

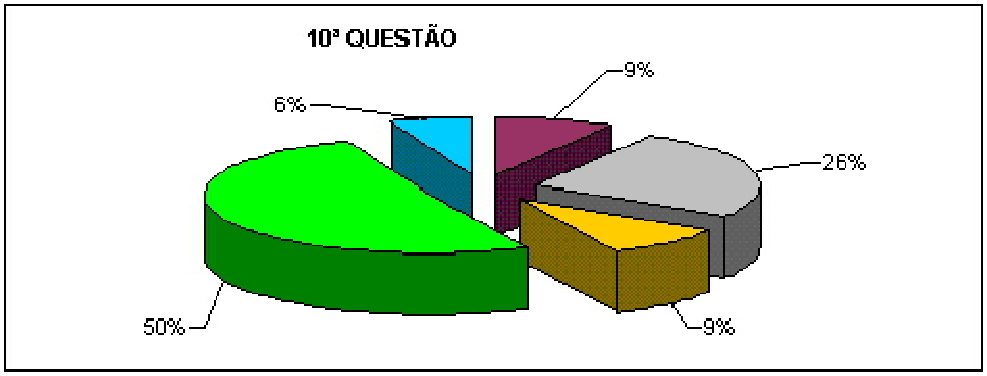
Apesar de a rede informatizada não contemplar todos os setores do QCG, vimos, pelo levantamento dos pesquisados, que já existe uma preocupação premente em resguardar informações sigilosas através de medidas de segurança física.

Entretanto, com a implementação da rede corporativa, essa segurança física adotada não será suficiente para conter atos delituosos quanto à quebra do sigilo.

Gráfico 9

RECURSOS UTILIZADOS EM SEGURANÇA NA REDE DE COMPUTADORES

CRIPTO	SEG.	AUT	NDR	OUTROS
3	9	3	18	2



Legenda:

- Criptografia e assinatura digital
- Segurança de Host
- Autenticação
- NDR
- Outros

Por ocasião das entrevistas e das respectivas respostas ao questionário, ficou evidenciado um percentual de 50% dos pesquisados que não utilizam método algum de proteção em redes de computadores. Ainda assim, os casos registrados de prevenção da segurança ocorreram de forma particular, em residências, não se aplicando à Corporação.

APÊNDICE C

MANUAL DE INSTRUÇÕES DA CIFRA SAFER

Revisão de 15 de setembro de 1995

{NOTA: Esta revisão difere do texto original, apenas no fato em que, nos programas em TURBO PASCAL, as instruções:

```
logtab[1]:= 0;
  exptab[0]:= 1;
  FOR i:= 1 TO 255 DO
  BEGIN
    exptab[i]:= (45 * exptab[i - 1]) mod 257;
    logtab[exptab[i]]:= i;
  END;
  exptab[128]:= 0; logtab[0]:= 128; exptab[0]:= 1;
```

são modificadas para:

```
 logtab[1]:= 0;
  exptab[0]:= 1;
  FOR i:= 1 TO 255 DO
  BEGIN
    exptab[i]:= (45 * exptab[i - 1]) mod 257;
    IF exptab[i] < 256 THEN logtab[exptab[i]]:= i;
  END;
  exptab[128]:= 0; logtab[0]:= 128;
```

Existem mudanças apenas na sexta e na última linha. Estas mudanças corrigem um “defeito” nestes programas em TURBO PASCAL (e em todas as implementações do SAFER em TURBO PASCAL). Estas modificações foram gentilmente sugeridas por Eli

Biham. O problema é que a variável “logtab” é definida como uma variável do tipo “ARRAY[0..255] de inteiro”. Mas exptab[128]= 256 [-1 em GF(257)] de modo que as instruções originais faziam com que fosse armazenado o valor 128 na localização mal definida logtab[256]. No programa compilado, este resultado armazena o valor 128 na primeira localização do próximo arranjo, exptab, isto é na localização exptab[0], a qual foi corrigida pela instrução “exptab[0]:=1;” Esta mudança de fato não seria necessária, ocorre porém que com outros compiladores, a compilação das instruções originais podiam resultar em uma implementação errada do SAFER.

Publicação
De James L. Massey
De um Fortalecimento do Programa da Chave
Para a cifra SAFER (Secure and Fast Encryption Routine)), para chaves com os comprimentos de 64 e 128 bits.

A versão original do “SAFER”, que é uma cifra não proprietária, liberado para uso por qualquer um sem violação dos direitos autorais, foi chamado SAFER K-64 para enfatizar que a chave selecionada pelo usuário tem comprimento 64 bits. Esta cifra foi introduzida no artigo seguinte:

J. L. Massey, "SAFER K-64: A Byte-Oriented Block-Ciphering Algorithm," pp. 1-17 in Fast Software Encryption (Ed. R. Anderson), Lecture Notes in Computer Science No. 809. New York: Springer, 1994.

Quase que imediatamente após a publicação desta cifra, nós começamos a receber solicitações para modificações que permitissem o uso de chaves com 128 bits. O comprimento 128 é natural porque o cifrador usa um programa de chave em cada rodada de cifragem. Um programa de chave de 128 bits para o SAFER, tem uma propriedade desejável, que é: quando as duas metades da chave são idênticas, as subchaves produzidas por este programa de chave coincidem com aquelas produzidas pelo programa de chave para o SAFER-K64 . O programa de chaves de 128 bits foi proposta pelo *Special Projects Team of the Ministry of Home Affairs*, Singapura. Os projetistas deste programa de chave cederam todos os diretos autorais da mesma, assim nós então a aceitamos como o

programa de geração de chaves para o SAFER K-128. Exceto pelas diferentes chaves produzidas pelo programa de chave, as cifragens feitas pelo SAFER K-64 e K-128 são exatamente as mesmas. O SAFER K-128 foi publicado no seguinte artigo:

J. L. Massey, "SAFER K-64: One Year Later," presented at the K. U. Leuven Workshop on Algorithms, Leuven, Belgium, 14-16 December, 1994, and to appear in Fast Software Encryption II (Ed. B. Preneel), to be published by Springer-Verlag in the Lecture Notes in Computer Science Series, 1995.

Na publicação do SAFER K-64, foi recomendado o uso de 6 rodadas de cifragem e especificado que no máximo 10 rodadas deveriam ser usadas. Para o SAFER K-128, 10 rodadas foram recomendadas e especificado no máximo o uso de 12 rodadas.

Tendo publicado uma cifra não proprietária e disponível de graça, achamos que é de nossa responsabilidade informar qualquer "fraqueza" encontrada na mesma. A primeira "fraqueza" substancial, à qual nós atentamos, foi descoberta por Lars Knudsen em fevereiro deste ano e diz respeito ao uso do SAFER para *hashing*. Não é incomum usar cifras de chave secreta dentro de um esquema de *hashing* público. A força do cifrador para *hashing* depende da dificuldade de produzir colisões, isto é encontrar dois textos claros/ pares de chaves distintos que resultem em um mesmo texto cifrado. Quando o texto claro e o texto cifrado tem 64 bits de *string*, o número médio de pares de texto claro/chaves que deve ser escolhido uniformemente e aleatoriamente antes de uma colisão ser encontrada é próximo de 2^{-32}. Um talentoso criptoanalista, Knudsen projetou um método para produzir colisões para o SAFER-K64 com 6 rodadas depois de escolher apenas cerca de 2^{24} pares de textos claros/chaves, isto é, cerca de 256 vezes tão rápido quanto se supusesse aleatório. (Como o SAFER K-128 reduz-se ao SAFER K-64 quando as duas metades da chave de 128 bits coincidem, o ataque de Knudsen também é aplicável ao SAFER K-128.) Na semana passada no artigo "*A Key-Schedule Weakness in SAFER K-64*," apresentado em *Crypto '95* em Santa Barbara, Califórnia, Knudsen descreveu este ataque. Este ataque é explicado pelo fato de que , para a rodada "r" no SAFER K-64 e no SAFER K-128, na mudança de um byte na chave secreta há a mudança de apenas 1 byte na mesma posição em todas as "2r + 1" chaves rodadas . Duas rodadas de chaves diferindo em apenas um byte algumas vezes cifrarão uma rodada de entrada para a mesma rodada de saída.

Knudsen foi capaz de selecionar duas chaves secretas diferindo em apenas um byte de forma que ambas as chaves cifraram entre 2^{22} e 2^{28} textos claros da mesma forma por seis rodadas. Este é então o fenômeno explicado para produzir colisões cerca de 256 vezes mais rápidas do que se supõe aleatório quando o SAFER K-64 com seis rodadas é usado dentro do padrão de esquemas de *hashing.* Ele também encontrou pares de chaves secretas que cifram cerca de 2^{15} textos claros da mesma forma para oito rodadas, mas isto não é bastante para dar uma vantagem sobre o que se supõe aleatório em produzir colisões. Ele também determinou que não há pares de chaves secretas que cifram muitos textos claros da mesma forma para 10 ou mais rodadas. Knudsen também mostrou que esta "fraqueza" do programa de chave podia ser explorada em um ataque de texto claro escolhido em um SAFER K-64 com seis rodadas da então chamada *related-key* que, indesejavelmente, geralmente recupera 8 bits da chave utilizando cerca de 2^{45} textos claros escolhidos – este ataque não é muito realista mas parece significativo que ele resulte da mesma fraqueza do programa de chave como o mais realista ataque de *hashing*.

Há cerca de três meses, nós recebemos uma cópia de um artigo "*An Analysis of SAFER*," de Sean Murphy, o qual foi submetido a publicação para o *Journal of Cryptology*. Em uma análise muito elegante, Murphy mostrou que existe um *rank –4* (isto é, 4 bytes*) submodule* que é invariante sob a transformação de Pseudo-Hadamard (PHT) [que é uma transformação linear em oito bytes usada dentro do SAFER como um mecanismo primário para obter difusão dentro do cifrador] e que este *rank-4 submodule* depende apenas de seis dos oito bytes de entrada para PHT. Como para a rodada "r", o SAFER K-64 e o SAFER K-128, mudando um byte da chave secreta, mudam somente o byte na mesma posição em todas as 2r+1 chaves das rodadas, segue que existe uma função linear de quatro bytes do texto claro e uma função linear de quatro bytes do texto cifrado, dos quais as junções estatísticas dependem em apenas seis dos oito bytes das chaves no SAFER K-64 e somente em doze dos dezesseis bytes das chaves no SAFER K-128. Esta observação não aparece para guiar qualquer ataque prático em ambos o SAFER-K64 e SAFER K-128, mas ela novamente sugere que é uma má idéia usar o programa de chaves do SAFER original. Para limitar a influência de cada byte da chave selecionada do usuário para os bytes da subchave na mesma posição. Esta é a mesma fraqueza que Knudsen explorou.

Como o raio caiu duas vezes no mesmo local decidimos que seria aconselhável não adotar um novo programa de geração de chaves para o SAFER, com nossas desculpas para aqueles usuários que já tenham implementado o SAFER K-64 e/ou o SAFER K-128 em hardware ou software, apesar de nenhum ataque prático ter sido encontrado para o SAFER K-64 com oito ou mais rodadas ou para o SAFER K-128. No artigo mencionado acima, Knudsen sugeriu uma modificação mais recente do programa de chaves para o SAFER, que neutraliza os ataques de *hashing* e os ataques de *key-related* que ele formulou. Estas modificações de 64 bits e 128 bits no programa de chaves também removeram a fraqueza encontrada por Murphy nas quais as estatísticas conjuntas agora dependem de todos os bytes da chave. Knudsen nos assegurou que ele cede todos os direitos autorais para estes programas de chaves modificados e nós, então, as utilizaremos para uso com o SAFER. Para minimizar as confusões com as chaves programadas anteriores, nós vamos nos referir às cifras com os novos programas de chaves por:

SAFER SK-64 e SAFER SK-128

Em que SK significa "*Strengthened Key schedule*". Nós estamos também mudando para 8 o número recomendado de rodadas para o SAFER SK-64, com um mínimo de 6 rodadas e um máximo de 10 rodadas. Para o SAFER K-128, nós mantemos a recomendação anterior de 10 rodadas com um máximo de 12 rodadas.

Uma inovação no programa de chaves de Knudsen é a adição de um nono "byte de paridade" para os 8 bytes selecionados pelo usuário no SAFER SK-64 e para cada metade dos 16 bytes da chave do usuário selecionada no SAFER SK-128. Isto tem o efeito desejável de originar duas chaves diferentes selecionadas pelo usuário diferindo no mínimo em dois bytes depois da expansão com o byte de paridade. Como no programa de chave original, existe uma rotação adicional dentro dos bytes da chave selecionada pelo usuário, mas antes ela é adicionada a uma chave de polarização produzindo uma rodada de uma subchave. A novidade é que os 8 bytes selecionados para adição na chave de polarização são selecionados dos nove bytes expandidos da chave selecionada pelo usuário em uma base de rotação, isto é, os primeiros bytes 1,2,3,4,5,6,7,8 são usados, então os bytes

2,3,4,5,6,7,8,9 são usados, então os bytes 3,4,5,6,7,8,9,1 são usados, etc. Isto remedia a fraqueza do programa de chave para o SAFER K-64 e SAFER K-128 no qual mudando um byte da chave secreta há a mudança de apenas um byte nesta mesma posição em todas as 2r+1 rodadas das chaves.

Anexados a estas publicações estão os programas em TURBO PASCAL para ambos o SAFER SK-64 e SAFER SK-128, juntamente com exemplos de cifragem para pessoas interessadas em checar as próprias implementações destes cifradores. Estes programas em TURBO PASCAL constituem a descrição oficial dos cifradores SAFER SK-64 e SAFER SK-128.

TURBO PASCAL PROGRAM FOR SAFER SK-64

```
PROGRAM Full_r_Rounds_max_10_of_SAFER_SK64_cipher;


VAR a1, a2, a3, a4, a5, a6, a7, a8, b1, b2, b3, b4, b5, b6,
    b7, b8, r: byte;
    k: ARRAY[1..21,1..8] OF byte; k1: ARRAY[1..9] OF byte;
    logtab, exptab: ARRAY[0..255] OF integer; i, j, n, flag: integer;

PROCEDURE mat1(VAR a1, a2, b1, b2: byte);
BEGIN
  b2:= a1 + a2;
  b1:= b2 + a1;
END;

PROCEDURE invmat1(VAR a1, a2, b1, b2: byte);
BEGIN
  b1:= a1 - a2;
  b2:= -b1 + a2;
END;


BEGIN
  {This portion of the program computes the powers of the primitive
   element 45 of the finite field GF(257) and stores these numbers
   in the table "exptab".  The corresponding logarithms to the base
   45 are stored in the table "logtab".}
  logtab[1]:= 0;
```

```
exptab[0]:= 1;
FOR i:= 1 TO 255 DO
BEGIN
  exptab[i]:= (45 * exptab[i - 1]) mod 257;
  IF exptab[i] < 256 THEN logtab[exptab[i]]:= i;
END;
exptab[128]:= 0; logtab[0]:= 128;

flag:= 0;
writeln;
writeln('Enter number of rounds r (max. 10) desired then hit CR');
readln(r);

REPEAT
  BEGIN
    writeln;
    writeln('Enter plaintext in 8 bytes (a byte is an integer');
    writeln('between 0 and 255 incl. with spaces between bytes)');
    writeln('then hit CR.');
    readln(a1, a2, a3, a4, a5, a6, a7, a8);
    writeln('Enter a key in 8 bytes then hit CR.');
    readln(k[1,1],k[1,2],k[1,3],k[1,4],k[1,5],k[1,6],k[1,7],k[1,8]);
    k1[1]:= k[1,1]; k1[2]:= k[1,2]; k1[3]:= k[1,3]; k1[4]:= k[1,4];
    k1[5]:= k[1,5]; k1[6]:= k[1,6]; k1[7]:= k[1,7]; k1[8]:= k[1,8];
    writeln;
    writeln('PLAINTEXT is ', a1:8,a2:4,a3:4,a4:4,
                        a5:4,a6:4,a7:4,a8:4);
    writeln('The KEY is   ', k[1,1]:8,k[1,2]:4,k[1,3]:4,k[1,4]:4,
                        k[1,5]:4,k[1,6]:4,k[1,7]:4,k[1,8]:4);

    {The next instruction appends a "parity byte" to the key K1.}
    k1[9] := k1[1] xor k1[2] xor k1[3] xor k1[4] xor
           k1[5] xor k1[6] xor k1[7] xor k1[8];
    {The next instructions implement the key schedule
     needed to derive keys K2, K3, ... K2r+1 from the input key K1.}
    FOR n:= 2 TO 2*r+1 DO
    BEGIN
      {Each byte of the  key K1 is further left rotated by 3.}
      FOR j:= 1 TO 9 DO k1[j]:= (k1[j] shl 3) + (k1[j] shr 5);
      FOR j:= 1 TO 8 DO
      BEGIN
        {The key bias is added here to the further right rotated K1.}
        k[n,j]:= k1[((j+n-2) mod 9) + 1] + exptab[exptab[9*n+j]];
      END;
    END;

    {The r rounds of encryption begin here.}
    FOR i:= 1 TO r DO
    BEGIN
```

```
  {Key 2i-1 is mixed bit and byte added  to the round input.}
  a1:= a1 xor k[2*i-1,1]; a2:= a2  +  k[2*i-1,2];
  a3:= a3  +  k[2*i-1,3]; a4:= a4 xor k[2*i-1,4];
  a5:= a5 xor k[2*i-1,5]; a6:= a6  +  k[2*i-1,6];
  a7:= a7  +  k[2*i-1,7]; a8:= a8 xor k[2*i-1,8];

  {The result now passes through the nonlinear layer.}
  b1:= exptab[a1]; b2:= logtab[a2];
  b3:= logtab[a3]; b4:= exptab[a4];
  b5:= exptab[a5]; b6:= logtab[a6];
  b7:= logtab[a7]; b8:= exptab[a8];

  {Key 2i is now mixed byte and bit added to the result.}
  b1:= b1  +  k[2*i,1]; b2:= b2 xor k[2*i,2];
  b3:= b3 xor k[2*i,3]; b4:= b4  +  k[2*i,4];
  b5:= b5  +  k[2*i,5]; b6:= b6 xor k[2*i,6];
  b7:= b7 xor k[2*i,7]; b8:= b8  +  k[2*i,8];

  {The result now enters the linear layer.}
  mat1(b1, b2, a1, a2);
  mat1(b3, b4, a3, a4);
  mat1(b5, b6, a5, a6);
  mat1(b7, b8, a7, a8);

  mat1(a1, a3, b1, b2);
  mat1(a5, a7, b3, b4);
  mat1(a2, a4, b5, b6);
  mat1(a6, a8, b7, b8);

  mat1(b1, b3, a1, a2);
  mat1(b5, b7, a3, a4);
  mat1(b2, b4, a5, a6);
  mat1(b6, b8, a7, a8);

  {The round is now completed!}
  writeln('after round',i:2,a1:8,a2:4,a3:4,a4:4,
                        a5:4,a6:4,a7:4,a8:4);
END;

{Key 2r+1 is now mixed bit and byte added to produce the
 final cryptogram.}
a1:= a1 xor k[2*r+1,1]; a2:= a2  +  k[2*r+1,2];
a3:= a3  +  k[2*r+1,3]; a4:= a4 xor k[2*r+1,4];
a5:= a5 xor k[2*r+1,5]; a6:= a6  +  k[2*r+1,6];
a7:= a7  +  k[2*r+1,7]; a8:= a8 xor k[2*r+1,8];
writeln('CRYPTOGRAM is', a1:8,a2:4,a3:4,a4:4,
                         a5:4,a6:4,a7:4,a8:4);
writeln;
```

```
    writeln('Type 0 and CR to continue or -1 and CR to stop run.');
    read(flag);
  END
 UNTIL flag < 0;
END.
```

EXEMPLOS DE CIFRAGEM COM O SAFER SK-64

Exemplos de cifragem com o SAFER SK-64 (i.e., com a chave fortalecida programada de 64 bits.)

PLAINTEXT is	1	2	3	4	5	6	7	8
The KEY is	0	0	0	0	0	0	0	1
after round 1	131	177	53	27	130	249	141	121
after round 2	68	73	32	102	134	54	206	57
after round 3	248	213	217	11	23	68	0	243
after round 4	194	62	109	79	24	18	13	84
after round 5	153	156	246	172	40	72	173	39
after round 6	154	242	34	6	61	35	216	28
CRYPTOGRAM is	21	27	255	2	173	17	191	45

PLAINTEXT is	1	2	3	4	5	6	7	8
The KEY is	1	2	3	4	5	6	7	8
after round 1	223	98	177	100	46	234	13	210
after round 2	182	246	230	93	158	14	48	89
after round 3	45	234	128	149	40	101	10	134
after round 4	30	17	249	236	158	120	69	100
after round 5	1	200	182	241	0	127	152	162
after round 6	144	85	94	214	5	38	65	150
CRYPTOGRAM is	95	206	155	162	5	132	56	199

TURBO PASCAL PROGRAM FOR SAFER SK-128

```
PROGRAM Full_r_Rounds_max_12_of_SAFER_SK128_cipher;
{Choosing both halves of the key to coincide gives the same result as for
encrypting with this same 64 bit sequence as the key for SAFER SK-64.}

VAR a1, a2, a3, a4, a5, a6, a7, a8, b1, b2, b3, b4, b5, b6, b7, b8, r: byte;
```

```
   k: ARRAY[1..25,1..8] OF byte; ka: ARRAY[1..9] OF byte;
   kb: ARRAY[1..9] OF byte;
   logtab, exptab: ARRAY[0..255] OF integer; i, j, flag: integer;

PROCEDURE mat1(VAR a1, a2, b1, b2: byte);
BEGIN b2:= a1 + a2; b1:= b2 + a1; END;

PROCEDURE invmat1(VAR a1, a2, b1, b2: byte);
BEGIN b1:= a1 - a2; b2:= -b1 + a2; END;

BEGIN
{The program here computes the powers of the primitive element 45 of the
 finite field GF(257) and stores these in the table "exptab".  Corresponding
 logarithms to the base 45 are stored in the table "logtab".}
  logtab[1]:= 0; exptab[0]:= 1;
  FOR i:= 1 TO 255 DO
  BEGIN
    exptab[i]:= (45 * exptab[i - 1]) mod 257;
    IF exptab[i] < 256 THEN logtab[exptab[i]]:= i;
  END;
  exptab[128]:= 0; logtab[0]:= 128;

  flag:= 0; writeln;
  WRITELN('Enter number of rounds r (max. 12) then hit CR.');
  READLN(r);

  REPEAT
    BEGIN
      WRITELN;
      WRITELN('Enter plaintext in 8 bytes with spaces');
      WRITELN('  between bytes, then hit CR.');
      WRITELN('(A byte is an integer between 0 and 255 inclusive.)');
      READLN(a1, a2, a3, a4, a5, a6, a7, a8);
      WRITELN('Enter left half of key (Ka) in 8 bytes then hit CR.');
      READLN(ka[1],ka[2],ka[3],ka[4],ka[5],ka[6],ka[7],ka[8]);
      WRITELN('Enter right half of key (Kb) in 8 bytes then hit CR.');
      READLN(kb[1],kb[2],kb[3],kb[4],kb[5],kb[6],kb[7],kb[8]);
      WRITELN;
      WRITELN('Key Ka  is', ka[1]:4,ka[2]:4,ka[3]:4,ka[4]:4,
                      ka[5]:4,ka[6]:4,ka[7]:4,ka[8]:4);
      WRITELN('Key Kb  is', kb[1]:4,kb[2]:4,kb[3]:4,kb[4]:4,
                      kb[5]:4,kb[6]:4,kb[7]:4,kb[8]:4);
      WRITELN('PLAINTEXT is ', a1:8,a2:4,a3:4,a4:4,a5:4,a6:4,a7:4,a8:4);

    {The next instruction appends a "parity byte" to keys Ka and Kb.}
    ka[9] := ka[1] xor ka[2] xor ka[3] xor ka[4] xor
           ka[5] xor ka[6] xor ka[7] xor ka[8];
    kb[9] := kb[1] xor kb[2] xor kb[3] xor kb[4] xor
           kb[5] xor kb[6] xor kb[7] xor kb[8];
```

```
{The next instructions implement the key schedule
 needed to derive keys K1, K2, ... K2r+1 from the
 128 bit input key (Ka, Kb).}
{K1 is set equal to Kb.}
FOR j:= 1 TO 8 DO k[1,j]:= kb[j];
{Each byte of the  key Ka is right rotated by 3.}
FOR j:= 1 TO 9 DO ka[j]:= (ka[j] shr 3) + (ka[j] shl 5);
FOR i:= 1 TO r DO
BEGIN
  {Each byte of keys Ka and Kb is further left rotated by 6.}
  FOR j:= 1 TO 9 DO
  BEGIN
    ka[j]:= (ka[j] shl 6) + (ka[j] shr 2);
    kb[j]:= (kb[j] shl 6) + (kb[j] shr 2);
  END;
  {The key biases are added to give the keys K2i-1 and K2i.}
  FOR j:= 1 TO 8 DO
  BEGIN
    k[2*i,j]:= ka[((j+2*i-2) mod 9) + 1] + exptab[exptab[18*i+j]];
    k[2*i+1,j]:= kb[((j+2*i-1) mod 9) + 1] + exptab[exptab[18*i+9+j]];
  END;
END;
  {The r rounds of encryption begin here.}
  FOR i:= 1 TO r DO
  BEGIN
    {Key 2i-1 is mixed bit and byte added  to the round input.}
    a1:= a1 xor k[2*i-1,1]; a2:= a2  +  k[2*i-1,2];
    a3:= a3  +  k[2*i-1,3]; a4:= a4 xor k[2*i-1,4];
    a5:= a5 xor k[2*i-1,5]; a6:= a6  +  k[2*i-1,6];
    a7:= a7  +  k[2*i-1,7]; a8:= a8 xor k[2*i-1,8];

    {The result now passes through the nonlinear layer.}
    b1:= exptab[a1]; b2:= logtab[a2]; b3:= logtab[a3]; b4:= exptab[a4];
    b5:= exptab[a5]; b6:= logtab[a6]; b7:= logtab[a7]; b8:= exptab[a8];
    {Key 2i is now mixed byte and bit added to the result.}
    b1:= b1  +  k[2*i,1]; b2:= b2 xor k[2*i,2];
    b3:= b3 xor k[2*i,3]; b4:= b4  +  k[2*i,4];
    b5:= b5  +  k[2*i,5]; b6:= b6 xor k[2*i,6];
    b7:= b7 xor k[2*i,7]; b8:= b8  +  k[2*i,8];

    {The result now enters the first level of the linear layer.}
    mat1(b1, b2, a1, a2); mat1(b3, b4, a3, a4);
    mat1(b5, b6, a5, a6); mat1(b7, b8, a7, a8);
    {The result now enters the second level of the linear layer.}
    mat1(a1, a3, b1, b2); mat1(a5, a7, b3, b4);
    mat1(a2, a4, b5, b6); mat1(a6, a8, b7, b8);
    {The result now enters the third level of the linear layer.}
    mat1(b1, b3, a1, a2); mat1(b5, b7, a3, a4);
```

```
      mat1(b2, b4, a5, a6); mat1(b6, b8, a7, a8);

      {The round is now completed!}
      WRITELN('after round',i:2,a1:8,a2:4,a3:4,a4:4,a5:4,a6:4,a7:4,a8:4);
    END;

    {Key 2r+1 is now mixed bit and byte added to produce the cryptogram.}
    a1:= a1 xor k[2*r+1,1]; a2:= a2  +  k[2*r+1,2];
    a3:= a3  +  k[2*r+1,3]; a4:= a4 xor k[2*r+1,4];
    a5:= a5 xor k[2*r+1,5]; a6:= a6  +  k[2*r+1,6];
    a7:= a7  +  k[2*r+1,7]; a8:= a8 xor k[2*r+1,8];
    WRITELN('CRYPTOGRAM is',a1:8,a2:4,a3:4,a4:4,a5:4,a6:4,a7:4,a8:4); writeln;
    WRITELN('Type 0 and CR to continue or -1 and CR to stop run.');
    READLN(flag);
  END
 UNTIL flag < 0;
END.
```

EXEMPLOS DE CIFRAGEM COM O SAFER SK-128

Exemplos de cifragem com o SAFER SK-128 (i.e., com a chave fortalecida dO programa de 128 bits.)

```
PLAINTEXT is        1   2   3   4   5   6   7   8
KEY Ka is           0   0   0   0   0   0   0   1
KEY Kb is           0   0   0   0   0   0   0   1
after round 1     131 177  53  27 130 249 141 121
after round 2      68  73  32 102 134  54 206  57
after round 3     248 213 217  11  23  68   0 243
after round 4     194  62 109  79  24  18  13  84
after round 5     153 156 246 172  40  72 173  39
after round 6     154 242  34   6  61  35 216  28
after round 7     100  31 172  67  44  75 133 219
after round 8      78 226 239 135 210  83  93  72
after round 9      72  64  46 195 163 159 243 114
after round10       3 133  76 190 191  52 220 123
CRYPTOGRAM is      65  76  84  90 182 153  74 247

PLAINTEXT is        1   2   3   4   5   6   7   8
KEY Ka is           1   2   3   4   5   6   7   8
KEY Kb is           0   0   0   0   0   0   0   0
after round 1      64 214  74 216 103 222  26  54
after round 2      61  14  68  15  46 111 124  80
```

after round 3	197	124	96	59	255	24	2	30
after round 4	63	59	214	103	236	166	153	24
after round 5	66	254	26	45	152	223	5	122
after round 6	89	47	58	105	161	38	135	45
after round 7	19	202	174	44	57	206	52	25
after round 8	78	179	113	208	169	26	121	22
after round 9	53	17	81	215	120	37	206	246
after round10	189	177	9	0	186	82	208	253
CRYPTOGRAM is	255	120	17	228	179	167	46	113

PLAINTEXT is	1	2	3	4	5	6	7	8
KEY Ka is	0	0	0	0	0	0	0	0
KEY Kb is	1	2	3	4	5	6	7	8
after round 1	95	186	209	220	166	66	213	10
after round 2	200	65	189	120	96	135	42	166
after round 3	64	169	43	166	132	171	31	40
after round 4	199	167	76	189	145	158	241	19
after round 5	71	55	184	212	108	198	77	108
after round 6	173	197	139	11	17	48	97	59
after round 7	17	51	142	4	170	7	207	124
after round 8	62	205	253	225	167	179	228	202
after round 9	133	168	127	138	193	243	34	226
after round10	59	194	69	220	220	231	123	148
CRYPTOGRAM is	73	201	157	152	165	188	89	8

22 de Outubro de 1995

Publicação

De James L. Massey

De um programa de chave de 40 bits

Para a cifra SAFER

(*Secure and Fast Encryption Routine*)

Em resposta a solicitações, nós projetamos uma versão do SAFER (*Secure and Fast Encryption Routine)* para acomodar uma chave de 40 bits selecionada pelo usuário com cifragem particularmente rápida. Como todos os antecedentes, o SAFER SK-40 é uma cifra não proprietária que, para o melhor do nosso conhecimento, é de domínio público sem violação de direitos autorais.

O SAFER SK-40 difere dos seus antecessores apenas no programa de chaves. O usuário escolhe os bytes K1(1), K1(2), K1(3), K1(4) e K1(5) da primeira rodada da chave K1. Esta primeira rodada da chave K1 é expandida para nove bytes pelo cálculo dos seguintes bytes redundantes:

K1(6) = K1(1) xor K1(3) xor 129.

K1(7) = K1(1) xor K1(4) xor K1(5) xor 66.

K1(8) = K1(2) xor K1(3) xor K1(5) xor 36.

K1(9) = K1(2) xor K1(4) xor 24.

Estas regras têm o efeito desejável de causar nas duas chaves de cinco bytes selecionadas pelo usuário, a expansão para chaves de nove bytes que diferem em no mínimo três bytes. Como na chave programada SK-64, existe uma rotação adicional nos bytes da chave de nove bytes antes de ser adicionada a uma “chave de polarização” para produzir a subchave da rodada. Os oito bytes selecionados para adição para a “chave de polarização” são selecionadas da chave expandida de nove bytes em uma base de rotação, isto é, os primeiros bytes 1, 2, 3, 4, 5, 6, 7, 8 são usados, então os bytes 2, 3, 4, 5, 6, 7, 8, 9 são usados, então os bytes 3, 4, 5, 6, 7, 8, 9, 1 são usados, etc.

Criptoanálise diferenciais extensivas do SAFER sugerem que 5 rodadas são suficientes para garantir que o ataque por criptoanálise diferencial requereria tantas cifragens quanto por busca exaustiva da chave em uma chave de 40 bits, isto é, cerca de 2^(40) cifragens seriam necessárias, mas 4 rodadas não garantiriam isso. Portanto nós escolhemos 5 rodadas para o SAFER SK-40.

Anexada a esta publicação estão os programas em TURBO PASCAL para o SAFER SK-40, juntamente com exemplos de cifragem para assistir as pessoas interessadas em checar suas próprias implementações destes cifradores. Estes programas em TURBO PASCAL constituem a descrição oficial do cifrador SAFER SK-40.

TURBO PASCAL PROGRAM FOR SAFER SK-40

```
PROGRAM SAFER_SK40_cipher;

VAR a1, a2, a3, a4, a5, a6, a7, a8, b1, b2, b3, b4, b5, b6,
    b7, b8: byte;
    k: ARRAY[1..11,1..8] OF byte; k1: ARRAY[1..9] OF byte;
    logtab, exptab: ARRAY[0..255] OF integer; i, j, n, flag: integer;

PROCEDURE mat1(VAR a1, a2, b1, b2: byte);
BEGIN
  b2:= a1 + a2;
  b1:= b2 + a1;
END;

PROCEDURE invmat1(VAR a1, a2, b1, b2: byte);
BEGIN
  b1:= a1 - a2;
  b2:= -b1 + a2;
END;

BEGIN
  {This portion of the program computes the powers of the primitive
   element 45 of the finite field GF(257) and stores these numbers
   in the table "exptab".  The corresponding logarithms to the base
   45 are stored in the table "logtab".}
  logtab[1]:= 0;
  exptab[0]:= 1;
  FOR i:= 1 TO 255 DO
  BEGIN
    exptab[i]:= (45 * exptab[i - 1]) mod 257;
    IF exptab[i] < 256 THEN logtab[exptab[i]]:= i;
  END;
  exptab[128]:= 0;
  logtab[0]:= 128;

  flag:= 0;
  REPEAT
    BEGIN
      writeln;
      writeln('Enter plaintext in 8 bytes (a byte is an integer');
      writeln('between 0 and 255 incl.) with spaces between bytes');
      writeln('then hit CR.');
      readln(a1, a2, a3, a4, a5, a6, a7, a8);
      writeln('Enter a key in 5 bytes then hit CR.');
      readln(k[1,1],k[1,2],k[1,3],k[1,4],k[1,5]);
      k1[1]:= k[1,1]; k1[2]:= k[1,2]; k1[3]:= k[1,3];
```

```
k1[4]:= k[1,4];  k1[5]:= k[1,5];

{The four redundant bytes of the key K1 are now computed.}
k[1,6]:= k[1,1] xor k[1,3] xor 129;  k1[6]:= k[1,6];
k[1,7]:= k[1,1] xor k[1,4] xor k[1,5] xor 66;  k1[7]:= k[1,7];
k[1,8]:= k[1,2] xor k[1,3] xor k[1,5] xor 36;  k1[8]:= k[1,8];
k1[9] := k[1,2] xor k[1,4] xor 24;

writeln;
writeln('PLAINTEXT is ', a1:8,a2:4,a3:4,a4:4,
                    a5:4,a6:4,a7:4,a8:4);
writeln('The KEY is   ', k1[1]:8,k1[2]:4,k1[3]:4,k1[4]:4,k1[5]:4);
writeln('Extended K1 is', k1[1]:7,k1[2]:4,k1[3]:4,k1[4]:4,
                     k1[5]:4,k1[6]:4,k1[7]:4,k1[8]:4,k1[9]:4);

{The next instructions implement the key schedule
 needed to derive keys K2, K3, ... K11 from the 5 byte input key.}
FOR n:= 2 TO 11 DO
BEGIN
  {Each byte of the  key K1 is further left rotated by 3.}
  FOR j:= 1 TO 9 DO k1[j]:= (k1[j] shl 3) + (k1[j] shr 5);
  FOR j:= 1 TO 8 DO
  BEGIN
    {The key bias is added here to the further right rotated K1.}
    k[n,j]:= k1[((j+n-2) mod 9) + 1] + exptab[exptab[9*n+j]];
  END;
END;

{The 5 rounds of encryption begin here.}
FOR i:= 1 TO 5 DO
BEGIN
  {Key 2i-1 is mixed bit and byte added  to the round input.}
  a1:= a1 xor k[2*i-1,1]; a2:= a2  +  k[2*i-1,2];
  a3:= a3  +  k[2*i-1,3]; a4:= a4 xor k[2*i-1,4];
  a5:= a5 xor k[2*i-1,5]; a6:= a6  +  k[2*i-1,6];
  a7:= a7  +  k[2*i-1,7]; a8:= a8 xor k[2*i-1,8];

  {The result now passes through the nonlinear layer.}
  b1:= exptab[a1]; b2:= logtab[a2];
  b3:= logtab[a3]; b4:= exptab[a4];
  b5:= exptab[a5]; b6:= logtab[a6];
  b7:= logtab[a7]; b8:= exptab[a8];

  {Key 2i is now mixed byte and bit added to the result.}
  b1:= b1  +  k[2*i,1]; b2:= b2 xor k[2*i,2];
  b3:= b3 xor k[2*i,3]; b4:= b4  +  k[2*i,4];
  b5:= b5  +  k[2*i,5]; b6:= b6 xor k[2*i,6];
  b7:= b7 xor k[2*i,7]; b8:= b8  +  k[2*i,8];
```

```
  {The result now enters the linear layer.}
  mat1(b1, b2, a1, a2);
  mat1(b3, b4, a3, a4);
  mat1(b5, b6, a5, a6);
  mat1(b7, b8, a7, a8);

  mat1(a1, a3, b1, b2);
  mat1(a5, a7, b3, b4);
  mat1(a2, a4, b5, b6);
  mat1(a6, a8, b7, b8);

  mat1(b1, b3, a1, a2);
  mat1(b5, b7, a3, a4);
  mat1(b2, b4, a5, a6);
  mat1(b6, b8, a7, a8);

  {The round is now completed!}
  writeln('after round',i:2,a1:8,a2:4,a3:4,a4:4,
                        a5:4,a6:4,a7:4,a8:4);
END;

{Key 11 is now mixed bit and byte added to produce the
 final cryptogram.}
a1:= a1 xor k[11,1]; a2:= a2  +  k[11,2];
a3:= a3  +  k[11,3]; a4:= a4 xor k[11,4];
a5:= a5 xor k[11,5]; a6:= a6  +  k[11,6];
a7:= a7  +  k[11,7]; a8:= a8 xor k[11,8];
writeln('CRYPTOGRAM is', a1:8,a2:4,a3:4,a4:4,
                      a5:4,a6:4,a7:4,a8:4);
writeln;
writeln('DECRYPTION');
 {Key 11 is now mixed bit and byte subtracted from input
  as specified in the input transformation for decryption.}
a1:= a1 xor k[11,1]; a2:= a2  -  k[11,2];
a3:= a3  -  k[11,3]; a4:= a4 xor k[11,4];
a5:= a5 xor k[11,5]; a6:= a6  -  k[11,6];
a7:= a7  -  k[11,7]; a8:= a8 xor k[11,8];

FOR i:= 1 TO 5 DO
BEGIN
  {The input first passes through the inverse linear layer.}
  invmat1(a1, a2, b1, b3);
  invmat1(a3, a4, b5, b7);
  invmat1(a5, a6, b2, b4);
  invmat1(a7, a8, b6, b8);

  invmat1(b1, b2, a1, a3);
  invmat1(b3, b4, a5, a7);
  invmat1(b5, b6, a2, a4);
```

```
      invmat1(b7, b8, a6, a8);

      invmat1(a1, a2, b1, b2);
      invmat1(a3, a4, b3, b4);
      invmat1(a5, a6, b5, b6);
      invmat1(a7, a8, b7, b8);

      {Key 12-2i is mixed byte and bit subtracted from the result.}
      a1:= b1  -  k[12-2*i,1]; a2:= b2 xor k[12-2*i,2];
      a3:= b3 xor k[12-2*i,3]; a4:= b4  -  k[12-2*i,4];
      a5:= b5  -  k[12-2*i,5]; a6:= b6 xor k[12-2*i,6];
      a7:= b7 xor k[12-2*i,7]; a8:= b8  -  k[12-2*i,8];

      {The result now passes through the inverse nonlinear layer.}
      a1:= logtab[a1]; a2:= exptab[a2];
      a3:= exptab[a3]; a4:= logtab[a4];
      a5:= logtab[a5]; a6:= exptab[a6];
      a7:= exptab[a7]; a8:= logtab[a8];

      {Key 11-2i is mixed bit and byte subtracted from result.}
      a1:= a1 xor k[11-2*i,1]; a2:= a2  -  k[11-2*i,2];
      a3:= a3  -  k[11-2*i,3]; a4:= a4 xor k[11-2*i,4];
      a5:= a5 xor k[11-2*i,5]; a6:= a6  -  k[11-2*i,6];
      a7:= a7  -  k[11-2*i,7]; a8:= a8 xor k[11-2*i,8];

      {The round is now completed!}
      writeln('after round',i:2,a1:8,a2:4,a3:4,a4:4,
                          a5:4,a6:4,a7:4,a8:4);
    END;
    writeln('PLAINTEXT is ', a1:8,a2:4,a3:4,a4:4,
                        a5:4,a6:4,a7:4,a8:4);
    writeln;
    writeln('Type 0 and CR to continue or -1 and CR to stop run.');
    read(flag);
  END
 UNTIL flag < 0;
END.
```

EXEMPLOS DE CIFRAGEM COM O SAFER SK-40

```
PLAINTEXT is        1   2   3   4   5   6   7   8
The KEY is        231  16   0   8   0
Extended K1 is    231  16   0   8   0 102 173  52   0
after round 1      74  34  15 223  42  57 123  63
after round 2     104  59  95  82 106 157 105 195
```

```
after round 3          201   5 202   0 230 176  64  63
after round 4          125  81 110  26  35  89  92  28
after round 5           37  74  23   9 112 210 143  97
CRYPTOGRAM is           66 222 120  35 138 245 208  83

PLAINTEXT is             8   7   6   5   4   3   2   1
The KEY is              64   8  32  16 135
Extended K1 is          64   8  32  16 135 225 149 139   0
after round 1           56 204   6   4 192 156  85 189
after round 2           82 160 167 111  11  11 216 214
after round 3           40 106  50 173 193 191 136 117
after round 4           10  39 213 193  98 168 148 184
after round 5          118 110  14 145 173  81 155 189
CRYPTOGRAM is           19  10 113 154 116 110 177 143

PLAINTEXT is             1   2   3   4   4   3   2   1
The KEY is             135  40  32  48  64
Extended K1 is         135  40  32  48  64  38 181 108   0
after round 1          176 218 107 107 148   6 192 197
after round 2           83  64  79 115  72 171   6 103
after round 3          136  43 194 163 100   5  17 136
after round 4          111 208 179  80  60 102  22  10
after round 5          230 193 159 140 137 238 105 219
CRYPTOGRAM is          139  93  10 182  99  19 184 233
```

SAFER, 21 de dezembro de 1995

NOME

safer - encryption and decryption using SAFER

SINOPSE

safer

[-e | -d] [-ecb | -cbc | -cfb | -ofb]

(-k keyString | -kx keyHexString)

[-r nofRounds] [-s] [-v]

[inputFile [outputFile]]

safer

[-h | -hc] [-tan | -abr]

[-k keyString | -kx keyHexString]

[-r nofRounds] [-s] [-v]
[inputFile [[outputFile] hashvalFile]

DESCRIÇÃO

O SAFER lê o arquivo de entrada e escreve os dados de cifragem ou decifragem no arquivo de saída ou escreve o valor de *hash* no arquivo de valor de *hash*. Se o nome do arquivo não é dado em uma linha de comando, o SAFER usa o padrão de entrada e saída. Se o nome do arquivo de entrada é dado com "-" , o SAFER lê do padrão de entrada.

SAFER [36] (*Secure And Fast Encryption Routine*) é um cifrador de bloco desenvolvido pelo professor J.L. Massey *at the Swiss Federal Institute of Technology*. Existem quatro versões deste algoritmo, denominadas: SAFER K-64 [36], SAFER K-128 [37], SAFER SK-64 [39] and SAFER SK-128 [39]. Os numerais 64 e 128 significam o comprimento da chave selecionada pelo usuário, 'K' significa a versão original do programa de chaves e 'SK' significa a versão fortalecida do programa de chaves (na qual a fraqueza da chave original foi removida).

OPÇÕES

-e Cifragem (default).

-d Decifragem.

-k A chave é especificada com um *keyString*. Se o comprimento do *keyString* é menor que 10 caracteres, o *keyString* é interpretado como uma chave de 64 bits, senão como uma chave de 128 bits.

-kx A chave é especificada com *keyHexString*. Se o comprimento do *keyHexString* é menor do que 17 dígitos hexadecimais, a *keyHexString* é interpretada como uma chave de 64 bits, senão como uma chave de 128 bits. Para os modos -cbc, -cfb e -ofb, é possível especificar

um valor inicial denotado por y[0]. Neste caso a chave e o valor inicial são separados por uma coluna, isto é, '1234:9A'.

-r *nofRounds* dá o número de rodadas no processo de cifragem (e respectivamente no de decifragem) . O valor de *Default* são 6 rodadas para o SAFER K-64, 8 rodadas para o SAFER SK-64 e 10 rodadas para o SAFER K-128 e para o SAFER SK-128.

-s O programa de chaves fortalecido contido no SAFER SK-64 ou no SAFER SK-128 é usada, ao invés da chave programada original contida no SAFER K-64 ou no SAFER K-128.

-h Calcula o valor de *hash* de 128 bits dos dados de entrada. O valor de *hash* é escrito no arquivo de *hash* (ou na saída padronizada).

-hc Calcula o valor de *hash* de 128 bits dos dados de entrada. A entrada é copiada para o arquivo de saída (ou saída padronizada) e o valor de *hash* é escrito no arquivo de *hash* (ou erro padronizado).

-v *Verbose mode*. Os parâmetros selecionados são escritos para os padrões de erro.

Notações:

z = chave de 64-bit ou 128-bit
x[i] = i-th 64-bit do bloco de texto claro (i = 1..L+1)
y[i] = i-th 64-bit do bloco de text cifrado (i = 1..L+1)
x[1]..x[L] = texto claro original (o último bloco é preenchido com zeros)
x[L+1] = comprimento do texto claro original em bits
x[1]..x[L+1] = texto claro
y[1]..y[L+1] = texto cifrado
<a, b> = bloco de 128-bit composto de dois blocos de 64-bit
E(z, .) = função de cifragem sob a chave z
D(z, .) = função de decifragem, x = D(z, E(z, x))

^ = bit-by-bit exclusive-OR

~ = bit-by-bit complement

Modos de Cifragem/Decifragem:

-ecb electronic code book mode

y[i] = E(z, x[i])

x[i] = D(z, y[i])

-cbc cipher block chaining mode (default)

y[i] = E(z, x[i] ^ y[i-1])

x[i] = D(z, y[i]) ^ y[i-1]

-cfb ciphertext feedback mode

y[i] = x[i] ^ E(z, y[i-1])

x[i] = y[i] ^ E(z, y[i-1])

-ofb output feedback mode

h[i] = E(z, h[i-1])

y[i] = x[i] ^ h[i]

x[i] = y[i] ^ h[i]

Hash Functions:

If no key is given, safer uses the all zero key.

<h[0], g[0]> = z

hash value = <h[L+1], g[L+1]>

-tan tandem Davies-Meyer scheme (default)

w[i] = E(<g[i-1], x[i]>, h[i-1])

h[i] = h[i-1] ^ w[i]

g[i] = g[i-1] ^ E(<x[i], w[i]>, g[i-1])

-abr abreast Davies-Meyer scheme

h[i] = h[i-1] ^ E(<g[i-1], x[i]>, h[i-1])

g[i] = g[i-1] ^ E(<x[i], h[i-1]>, ~g[i-1])

Key Formats:

keyHexString = z:y[0] = { '0'..'9' | 'a'..'f' | 'A'..'F' | ':' }

keyString = z = { ' '..'~' }

EXEMPLOS

Para cifrar e decifrar um arquivo no modo de realimentação de cifra (*ciphertext feedback mode*):

safer -e -cfb -kx 123456:cDd7 data data.cr

safer -d -cfb -kx 123456:cDd7 data.cr data.ori

data e data.ori são idênticos

Para computar o valor *hash*:

safer -h data

Para computar o valor *hash* e cifrar os dados do primeiro passo:

safer -hck "k e y" data | safer -kx 12E3 - data.cr

PATENTE

"Apesar do projeto do SAFER K-64 ter sido solicitado pela *Cylink Corporation (Sunnyvale, CA, USA)*, a *Cylink* cedeu todos os direitos autorais deste algoritmo. Esta atitude da *Cylink* justifica-se pelo fato de que esta Companhia ganharia mais deste novo comércio do que perderia da competição. O SAFER K-64 não foi patenteado e é de graça para uso por qualquer um sem violação dos direitos autorais." [37]

AUTOR

Richard De Moliner (demoliner@isi.ee.ethz.ch)
Signal and Information Processing Laboratory
Swiss Federal Institute of Technology
CH-8092 Zurich, Switzerland

20 de novembro de 1995

From: massey@isi.ee.ethz.ch (Prof. James L. Massey)
Subject: SAFER and the NSA

A todos que tem tido comunicação comigo com interesse no SAFER

Eu fui informado por um membro do *Special Projects Team of the Ministry of Home Affairs, Singapore*, que na p. 341 da Segunda edição do livro, *Applied Cryptography*, Bruce Schneier escreveu: “O SAFER foi projetado pela Cylink, e a Cylink está manchada pela NSA [34]. Eu recomendo anos de intensa criptoanálise antes de usar o SAFER de qualquer forma”. Eu ainda não tenho a cópia deste livro, mas eu me sinto compelido a levar a conhecimento de vocês os fatos relativos a este problema.

O SAFER na verdade foi projetado por uma solicitação da CYLINK, conforme eu tornei público claramente no início dos artigos que eu publiquei sobre este cifrador
[J. L. Massey, "SAFER K-64: A Byte-Oriented Block-Ciphering Algorithm," in Fast Software Encryption (Ed. R. Anderson), Lecture Notes in Computer Science No. 809. New York: Springer, 1994, pp. 1-17 and J. L. Massey, "SAFER K-64: One Year Later," in Fast Software Encryption (Ed. B. Preneel), Lecture Notes in Computer Science No. 1008. New York: Springer, 1995, pp. 212-241.]
A razão que levou a CYLINK patrocinar este projeto está também relatada neste artigo.

O único critério de projeto dado a mim pela CYLINK foi que a cifra deve ser orientada a byte para facilitar a implementação desta em cartões inteligentes e que ela deveria ser rápida. Dentro deste direcionamento, o projeto do SAFER foi feito inteiramente por mim. Eu pessoalmente desenvolvi a cifra usando o meu PC em minha casa e eu fiz esta cifra tão forte quanto pude fazê-la. Enquanto estive projetando o SAFER eu nem consultei, nem fui consultado por qualquer pessoa da National Security Agency (NSA). A única assistência que eu tive foram retornos dados pelo grupo comissionado da CYLINK para executar as criptoanálises diferenciais dos meus projetos, o qual eu relatei claramente na seção 7 do artigo ["*SAFER K-64: A Byte-Oriented Block-Ciphering Algorithm*"] e adicionado conhecimento na seção 8 do último artigo *["SAFER K-64: One Year Later*"]. Mais especificamente, enquanto eu estava projetando o SAFER, eu nem consultei, nem fui contactado por qualquer pessoa de qualquer companhia ou qualquer agência.

Eu chamei o algoritmo do SAFER original de "SAFER K-64" para enfatizar que eu usei uma chave selecionada de 64 bits. Depois eu incorporei o projeto de uma chave programada de 128 bits que me foi proposto pelo *Special Projects Team of the Ministry of Home Affairs*, Singapura para obtenção do SAFER K-128" que foi publicado na seção 2 do último artigo ["*SAFER K-64: One Year Later*"]. A intenção do *Special Projects Team of the Ministry of Home Affairs*, Singapura, foi fortificar o cifrador, uma intenção que eu fiquei feliz em acomodar.

No meu e-mail publicado em 5 de setembro de 1995 (revisado em 15 de setembro de 1995 por um *glitch* de programação), eu reportei duas fraquezas no SAFER que foi encontrada por Lars Knudsen e por Sean Murphy. Eu mencionei o ataque de Knudsen em *hashing* com o SAFER na seção 7 do último artigo *["SAFER K-64: One Year Later"*]. Ambas aquelas fraquezas adviram do fato de que no programa das chaves do SAFER K-64 e do SAFER K-128, um byte da chave selecionada pelo usuário afeta somente o byte na posição correspondente em todas as chaves rodadas. Knudsen propôs um novo programa de geração de chaves que não só evita a estacionaridade do byte, mas possui outras propriedades também. No meu e-mail publicado em 5 de setembro de 1995 (revisada em 15 de setembro de 1995), adotei a chave programada de Knudsen em uma nova versão do SAFER que chamei "SAFER SK-64" (onde SK significa *Strengthened Key*). Eu também modifiquei a chave programada para o SAFER K-128 correspondente e publicado como

"SAFER SK-128". Em uma mensagem de e-mail datada de 22 de outubro de 1995, publiquei "SAFER SK-40", uma versão do SAFER com uma chave selecionada pelo usuário com 40 bits. Desenvolvi este novo programa de chave, como resultado de uma visita a Cylink um mês antes, durante a qual algumas pessoas expressaram interesse em uma chave com este comprimento por motivo das regras de exportação. Fiz o SAFER SK-40 tão seguro quanto foi possível fazer para chave com este comprimento e eu não conheço nenhum ataque que seja mais rápido do que o que seria por busca exaustiva.

O descrito acima é verdade e um relato completo do desenvolvimento do SAFER. Eu estou preparado em qualquer momento para levar este juramento a veracidade. Infelizmente, isto provavelmente não é bastante para desfazer o dano causado à reputação do SAFER pela insinuação de Bruce Schneier que o SAFER pode estar manchado de alguma forma por qualquer conexão com o NSA. Bruce Schineier não me informou que estava fazendo esta declaração nesta nova edição (apesar de eu ter contribuído com todas as informações que eu tinha acerca do SAFER e todos os questionamentos dele quando o mesmo estava preparando esta nova edição), para me dar o mínimo de oportunidade para negar a insinuação. Em minha opinião, a publicação desta insinuação é irresponsável, pois não existe como provar que a insinuação é falsa. A forma responsável para criticar uma proposta de uma cifra é demonstrar a sua fraqueza, não sugerir sem nenhuma evidência que alguma pessoa escusa pode ter construído dificuldade para encontrar alguma fraqueza nela. Até onde sabemos, o SAFER SK-64, SAFER SK-128 e SAFER SK-40 são cifradores bem fortes para as chaves com estes comprimentos e os mais velhos SAFER K-64 e SAFER K-128 são também fortes, se não tão bons quanto as novas versões.

Sinceramente, James L. Massey

APÊNDICE D

CÓDIGO FONTE DO SISTEMA DE GERENCIAMENTO DE CHAVES (VISUAL BASIC)

Form de Cadastro de Usuário

```
Option Explicit

' Variável para verificar se o login é novo ou está sendo alterado
Public bNovo As Boolean

' Variável para conexão com o banco de dados
Dim bd As ADODB.Connection

' Variável para guardar a data_hora da alteração
Dim sData_Altr As Date

Private Sub cmdConfirmar_Click()

'Variável para consulta ao banco de dados
Dim mRecordset As ADODB.Recordset

If mskLogin.ClipText = "" Then
   MsgBox "O campo Login deve ser preenchido.", vbOKOnly + vbCritical, "Erro"
   mskLogin.SetFocus
   Exit Sub
End If

If Len(Trim(mskLogin.ClipText)) <> 7 Then
   MsgBox "O campo Login deve ser preenchido com 6 digitos.", vbOKOnly + vbCritical, "Erro"
   mskLogin.SetFocus
   Exit Sub
End If

cmdConfirmar.Enabled = False
```

```
Set mRecordset = New ADODB.Recordset
mRecordset.ActiveConnection = bd
mRecordset.Source = "SELECT * FROM login WHERE login = '" & Trim(mskLogin.ClipText) & "'"
mRecordset.Open

If mRecordset.EOF Then
   MsgBox "Não existe nenhum usuário cadastrado com este login." + vbNewLine + "Preencha todos os campos e aperte em Salvar para inserir um novo usuário", vbOKOnly + vbInformation, "Cadastro de Usuários"

   bNovo = True

   txtNome.Text = ""
   txtSenha.Text = ""
   txtConfirmacaoSenha.Text = ""
   cmbNivel.ListIndex = -1

   cmbNivel.SetFocus
   cmdSalvar.Enabled = True
   cmdExcluir.Enabled = False
   cmdPesquisar.Enabled = False
Else
   bNovo = False

   txtNome.Text = mRecordset.Fields("nome")
   txtSenha.Text = mRecordset.Fields("senha")
   txtConfirmacaoSenha.Text = txtSenha.Text
   cmbNivel.Text = mRecordset.Fields("nivel")

   cmbNivel.SetFocus

   cmdSalvar.Enabled = True
   cmdExcluir.Enabled = True
   cmdPesquisar.Enabled = False
End If

mskLogin.Enabled = False
mskLogin.BackColor = &H8000000F

mRecordset.Close
Set mRecordset = Nothing

End Sub

Private Sub cmdExcluir_Click()

'Variável para armazenamento da instrução SQL
Dim sSql As String
Dim matricula As String
'Tratamento de erro
On Error GoTo Erro

   If bNovo = False Then
      'matricula = len
      sSql = "DELETE * FROM login WHERE login = '" & Format(mskLogin.ClipText, "0000000") & "'"

      If MsgBox("Deseja realmente excluir o usuário de login " & mskLogin.Text & " ?", vbYesNo + vbInformation) = vbYes Then
         bd.Execute sSql
```

```
        Else
          Exit Sub
        End If

          MsgBox "Exclusão efetuada com sucesso.", vbOKOnly + vbInformation
        cmdLimpar_Click
    End If

    Exit Sub

Erro:
    MsgBox Err.Description, vbOKOnly + vbCritical, "Erro"

End Sub

Private Sub cmdLimpar_Click()

mskLogin.Enabled = True
mskLogin.BackColor = &H80000005

mskLogin.Mask = ""
mskLogin.Text = ""
mskLogin.Mask = "###.###-#"

txtNome.Text = ""
txtSenha.Text = ""
txtConfirmacaoSenha.Text = ""
cmbNivel.ListIndex = -1

cmdSalvar.Enabled = False
cmdExcluir.Enabled = False
cmdConfirmar.Enabled = False
cmdPesquisar.Enabled = True

mskLogin.SetFocus

End Sub

Private Sub cmdpesquisar_Click()

frmPesquisarUsuarios.Show vbModal

If frmPesquisarUsuarios.bOK Then

    bNovo = False
    cmdSalvar.Enabled = True
    cmdExcluir.Enabled = True
    cmdPesquisar.Enabled = False
    cmdConfirmar.Enabled = False

    mskLogin.Mask = ""
    mskLogin.Text = frmPesquisarUsuarios.adoUsuarios.Recordset.Fields("login")
    mskLogin.Format = "000.000-0"
    cmbNivel.Text = frmPesquisarUsuarios.adoUsuarios.Recordset.Fields("nivel")
    txtSenha.Text = frmPesquisarUsuarios.adoUsuarios.Recordset.Fields("senha")
    txtNome.Text = frmPesquisarUsuarios.adoUsuarios.Recordset.Fields("nome")
    txtConfirmacaoSenha.Text = frmCadastroUsuarios.txtSenha.Text
```

```
    Unload frmPesquisarUsuarios

    mskLogin.Enabled = False
    mskLogin.BackColor = &H8000000F

    cmdConfirmar.Enabled = False

    cmbNivel.SetFocus
Else
    bNovo = True
    cmdSalvar.Enabled = False
    cmdExcluir.Enabled = False

    mskLogin.SetFocus
End If

End Sub

Private Sub cmdSair_Click()
    Unload Me
End Sub

Private Sub cmdSalvar_Click()

Dim sSql As String

On Error GoTo Erro

    If Trim(mskLogin.ClipText) = "" Or Trim(cmbNivel.Text) = "" Or Trim(txtSenha.Text) = "" Then
        MsgBox "Todos os campos devem ser preenchidos.", vbOKOnly + vbCritical, "Erro"
        Exit Sub
    End If

    If Len(Trim(txtSenha.Text)) <> 5 Then
        MsgBox "A senha deve conter 5 digitos numéricos.", vbOKOnly + vbCritical, "Erro"
        txtSenha.SetFocus
        Exit Sub
    End If

    If Trim(txtSenha.Text) <> Trim(txtConfirmacaoSenha.Text) Then
        MsgBox "A confirmação da senha não confere.", vbOKOnly + vbCritical, "Erro"
        txtSenha.SetFocus
        Exit Sub
    End If

    If bNovo Then
        sSql = "INSERT INTO login VALUES('" & mskLogin.ClipText & "','" & txtNome.Text & "','" &
txtSenha.Text & "','" & cmbNivel.Text & "','" & sLogin & "','" & Now & "')"
        bd.Execute sSql

        MsgBox "Inserção efetuada com sucesso.", vbOKOnly + vbInformation
        cmdLimpar_Click
    Else
        sSql = "UPDATE login SET nome = '" & txtNome.Text & "', senha = '" & txtSenha.Text & "', nivel = '"
& cmbNivel.Text & "', usuario_altr = '" & sLogin & "', data_altr = '" & Now & "' WHERE login = '" &
mskLogin.ClipText & "'" ' AND data_altr = '" & sData_Altr & "'"
        bd.Execute sSql
```

```
        MsgBox "Atualização efetuada com sucesso.", vbOKOnly + vbInformation
        cmdLimpar_Click
    End If

    Exit Sub

Erro:
    MsgBox Err.Description, vbOKOnly + vbCritical, "Erro"

End Sub

Private Sub Form_Load()

MousePointer = vbHourglass

Set bd = New ADODB.Connection
bd.ConnectionString = "DSN=Chave"
bd.Open

Me.Height = 2760
Me.Width = 7440
Me.Top = (MDIPrincipal.ScaleHeight - Me.Height) / 2
Me.Left = (MDIPrincipal.ScaleWidth - Me.Width) / 2

MousePointer = vbDefault

End Sub

Private Sub mskLogin_Change()

If Len(mskLogin.ClipText) = 7 Then
    cmdConfirmar.Enabled = True
Else
    cmdConfirmar.Enabled = False
End If

End Sub

Private Sub txtConfirmacaoSenha_Change()

    If Not IsNumeric(txtConfirmacaoSenha.Text) Then
        SendKeys "{Backspace}"
    End If

End Sub

Private Sub txtConfirmacaoSenha_GotFocus()

    txtConfirmacaoSenha.SelStart = 0
    txtConfirmacaoSenha.SelLength = Len(txtConfirmacaoSenha)

End Sub

Private Sub txtSenha_Change()

    If Not IsNumeric(txtSenha.Text) Then
        SendKeys "{Backspace}"
    End If
```

```
    If Len(Trim(txtSenha.Text)) = 5 Then
       txtConfirmacaoSenha.SetFocus
    End If

End Sub

Private Sub txtSenha_GotFocus()
    txtSenha.SelStart = 0
    txtSenha.SelLength = Len(txtSenha.Text)
End Sub
```

Form de Alteração de Constantes

```
Option Explicit
'Variável para conexão com o banco
Dim bd As ADODB.Connection
'Variável para resgistrar a data de alterações
Dim sData_Altr As Date
Public primo As Double

Private Sub cmdAlterar_Click()

 lblContar.Visible = True
 txtQtdDigitos.Visible = True
 txtNumPrimo.Enabled = True
 txtRaiz.Enabled = True
 cmdSalvar.Enabled = True


End Sub

Private Sub cmdLimpar_Click()

 lblContar.Visible = False
 txtQtdDigitos.Visible = False
 txtNumPrimo.Enabled = True
 txtRaiz.Enabled = True
 cmdSalvar.Enabled = False
 cmdAlterar.Enabled = False
 txtNumPrimo.Text = ""
 txtRaiz.Text = ""


End Sub

Private Sub cmdpesquisar_Click()
'Variável de conexão com o banco para pesquisa
Dim mRecordset As ADODB.Recordset

cmdLimpar.Visible = True
cmdAlterar.Enabled = True
primo = Empty

Set mRecordset = New ADODB.Recordset
mRecordset.ActiveConnection = bd
```

```
mRecordset.Source = "SELECT * FROM Constantes"
mRecordset.Open

txtNumPrimo.Text = mRecordset.Fields("Primo_N")
txtRaiz.Text = mRecordset.Fields("Raiz_Primitiva")
primo = mRecordset.Fields("Primo_N")
txtNumPrimo.Enabled = False
txtRaiz.Enabled = False

mRecordset.Close
Set mRecordset = Nothing

End Sub

Private Sub cmdSair_Click()
  Unload Me
End Sub

Private Sub cmdSalvar_Click()

'Variável para executar a alteração do número primo e a raiz
Dim sSql As String

    If txtNumPrimo.Text = "" Or txtRaiz.Text = "" Then
       MsgBox " Você deve preencher todos os campos.", vbOKOnly + vbCritical, "Erro"
       Exit Sub
    Else
       sSql = "UPDATE Constantes SET  Raiz_Primitiva = '" & txtRaiz.Text & "', usuario_altr = '" & sLogin &
"', data_altr = '" & Now & "', Primo_N = " & txtNumPrimo.Text & " WHERE Primo_N = " & primo
       bd.Execute sSql
       MsgBox "Atualização efetuada com sucesso.", vbOKOnly + vbInformation
       txtNumPrimo.Text = ""
       txtRaiz.Text = ""
       lblContar.Visible = False
       txtQtdDigitos.Visible = False
       txtNumPrimo.Enabled = False
       txtRaiz.Enabled = False
       cmdSalvar.Enabled = False
    End If
 cmdAlterar.Visible = False

End Sub

Private Sub Form_Load()

Me.Height = 1845
Me.Width = 4380
Me.Top = 2000
Me.Left = 3000

txtNumPrimo.Enabled = False
txtRaiz.Enabled = False
cmdSalvar.Enabled = False
cmdAlterar.Enabled = False
cmdLimpar.Visible = False
'Conexão com o Banco
Set bd = New ADODB.Connection
bd.ConnectionString = "DSN=Chave"
bd.Open
```

```
End Sub

Private Sub txtNumPrimo_Change()

   txtQtdDigitos.Text = Len(txtNumPrimo.Text)

End Sub
```

Form de Apresentação

```
Private Sub frmApres_Click()
   Unload Me
   frmLogin.Show 1
End Sub

Private Sub imgLogo_Click()
   Unload Me
   MDIPrincipal.Show
End Sub

Private Sub lblPlatform_Click()
   Unload Me
   frmLogin.Show 1
End Sub


Private Sub lblProductName_Click()
   Unload Me
   frmLogin.Show 1
End Sub


Private Sub Timer1_Timer()

Dim i As Integer

   For i = 1 To Timer1.Interval * 5
      ProgressBar1.Value = i
   Next i

   Unload Me
   frmLogin.Show 0

End Sub
```

Form Chave Secreta

```
Option Explicit
Dim bd As ADODB.Connection
'Variável para cálculo da Chave Secreta
Public Y As Double
```

```
Public W As Double
Dim K As Double
Public N As Double

Private Sub cmdCalcular_Click()

Dim resultado As Double
 On Error GoTo Erro

  W = Y ^ CDbl(txtChavPri.Text)
  W = Int(W + 0.5)
  N = Int(N + 0.5)
  K = Fix(W / N)
  resultado = W - (N * Fix(W / N))
  txtChavSec.Text = resultado

Exit Sub

Erro:
   MsgBox "Este cálculo não pode ser processado, o resultado é um número muito grande.", vbOKOnly + vbCritical, "Erro de Overflow"
End Sub

Private Sub cmdLimpar_Click()

  Y = Empty
  txtChavPri.Text = ""
  txtChavSec.Text = ""
  datcobPub.Text = ""

End Sub

Private Sub cmdSair_Click()
  Unload Me
End Sub

Private Sub datcobPub_Change()
Dim mRecordset As ADODB.Recordset
Dim mRecordset1 As ADODB.Recordset

  If datcobPub.Text <> "" Then
    Set mRecordset = New ADODB.Recordset
    mRecordset.ActiveConnection = bd
    mRecordset.Source = "SELECT * FROM Constantes"
    mRecordset.Open

    Set mRecordset1 = New ADODB.Recordset
    mRecordset1.ActiveConnection = bd
    mRecordset1.Source = "SELECT * FROM CPC WHERE Usuario = '" & datcobPub.Text & "'"
    mRecordset1.Open

    Y = mRecordset1.Fields("Chave_Pub")
    N = mRecordset.Fields("Primo_N")
  End If

End Sub

Private Sub Form_Load()
```

```
  Set bd = New ADODB.Connection
  bd.ConnectionString = "DSN=Chave"
  bd.Open

  Me.Height = 2415
  Me.Width = 4680
  Me.Top = 2000
  Me.Left = 3000
End Sub
```

Form do Catálogo Público Custodiado

```
Option Explicit

Dim bd As ADODB.Connection
Public ChvPub As Variant
Dim nome As String

Private Sub cmdConfirmar_Click()
   datgridCPC_DblClick
End Sub

Private Sub cmdExcluir_Click()
Dim sSql As String

On Error GoTo Erro

nome = adoCPC.Recordset.Fields("Usuario")

   sSql = "DELETE * FROM CPC WHERE Usuario = '" & nome & "'"

  If MsgBox("Deseja realmente excluir o usuário " & nome & " ?", vbYesNo + vbInformation) = vbYes Then
     bd.Execute sSql
  Else
    Exit Sub
  End If

    MsgBox "Exclusão efetuada com sucesso.", vbOKOnly + vbInformation
    cmdLimpar_Click
    adoCPC.Refresh
    datgridCPC.Refresh
  Exit Sub

Erro:
    MsgBox Err.Description, vbOKOnly + vbCritical, "Erro"

End Sub

Private Sub cmdLimpar_Click()
    txtNome.Text = ""
End Sub

Private Sub cmdSair_Click()
  Unload Me
End Sub
```

```
Private Sub datgridCPC_DblClick()
  frmChaveSecreta.datcobPub.Text = adoCPC.Recordset.Fields("usuario")
  frmChaveSecreta.Show 0
  Hide
End Sub

Private Sub Form_Load()

If sNivel = "1" Then
  cmdExcluir.Visible = True
End If

Me.Top = 2000
Me.Left = 3000
Me.Height = 2925
Me.Width = 6225
Set bd = New ADODB.Connection
bd.ConnectionString = "DSN=Chave"
bd.Open

End Sub

Private Sub txtNome_Change()

adoCPC.CommandType = adCmdText
adoCPC.RecordSource = "SELECT * FROM CPC WHERE Usuario LIKE '" & txtNome.Text & "%'"
adoCPC.Refresh

End Sub
```

Form do Login

```
Option Explicit

'Variável p/ conexão com o banco
Dim bd As ADODB.Connection
Public newlogin As Boolean


Private Sub cmdCancelar_Click()

If MDIPrincipal.newlogin Then
   Unload Me
   MDIPrincipal.Show 0
Else
  Me.Hide
  End

End If

End Sub
```

```
Private Sub cmdOk_Click()

Dim mRecordset As ADODB.Recordset

Set mRecordset = New ADODB.Recordset
mRecordset.ActiveConnection = bd
mRecordset.Source = "SELECT nome, nivel FROM login WHERE login = '" & Trim(MskLogin.ClipText)
& " ' AND senha = '" & Trim(txtSenha.Text) & "'"
mRecordset.Open

If mRecordset.EOF Then
   MsgBox "O login ou a senha estão inválidos.", vbOKOnly + vbCritical, "Verificação de login e senha"
   MskLogin.SetFocus
   SendKeys "{Home}+{End}"

   mRecordset.Close
   Set mRecordset = Nothing

   Exit Sub
Else
   sNivel = mRecordset.Fields("nivel")
   sLogin = MskLogin.ClipText
   MDIPrincipal.StatusBar.Panels(1) = "Usuário:   " & mRecordset.Fields("Nome")
   mRecordset.Close
   Set mRecordset = Nothing
End If

Me.Hide
Unload Me
MDIPrincipal.Show 0
MDIPrincipal.mnuConstante.Visible = False
MDIPrincipal.mnuUsuário.Visible = False

If sNivel = 1 Then
  MDIPrincipal.mnuConstante.Visible = True
  MDIPrincipal.mnuUsuário.Visible = True
End If

End Sub

Private Sub Form_Load()

Me.Height = 2220
Me.Width = 3750
Me.Top = (MDIPrincipal.ScaleHeight - Me.Height)
Me.Left = (MDIPrincipal.ScaleWidth - Me.Width)

'Conexão com o banco
Set bd = New ADODB.Connection
bd.ConnectionString = "DSN=Chave"
bd.Open
End Sub

Private Sub Form_Unload(Cancel As Integer)

bd.Close
Set bd = Nothing

End Sub
```

```
Private Sub mskLogin_Change()

If Len(Trim(MskLogin.ClipText)) = 7 Then
   txtSenha.SetFocus
   SendKeys "{Home}+{End}"
End If

End Sub

Private Sub txtSenha_KeyDown(KeyCode As Integer, Shift As Integer)
 If KeyCode = 13 Then
   cmdOk_Click
 End If
End Sub
```

Form da Pesquisa de Usuários

```
Option Explicit

Public bOK As Boolean
Private Sub cmdCancelar_Click()

bOK = False

Unload Me

End Sub

Private Sub cmdOk_Click()

bOK = True

Me.Hide

End Sub

Private Sub Form_Load()

MousePointer = vbHourglass

Me.Height = 2685
Me.Width = 7650
Me.Top = (MDIPrincipal.ScaleHeight - Me.Height) / 2
Me.Left = (MDIPrincipal.ScaleWidth - Me.Width) / 2

MousePointer = vbDefault

End Sub
```

Form do Cadastro de Chaves

```
Option Explicit

Dim bd As ADODB.Connection
```

```
'Variáveis para cálculo do módulo
Dim Z As Double
Public N As Double
Public raiz As Double
Dim resultado As Double
Dim calcular As Double

Private Sub cmdCadastrar_Click()

Dim mRecordset As ADODB.Recordset
Dim sSql As String

On Error GoTo Erro

    Set mRecordset = New ADODB.Recordset
    mRecordset.ActiveConnection = bd
    mRecordset.Source = "SELECT * FROM CPC WHERE Usuario = '" & txtUsuario.Text & " ' "
    mRecordset.Open

    If mRecordset.EOF Then

        sSql = "INSERT INTO CPC VALUES('" & txtUsuario.Text & "','" & txtChvPub.Text & "','" & Now &
"')"
        bd.Execute sSql

        MsgBox "Inserção efetuada com sucesso.", vbOKOnly + vbInformation
        cmdLimpar_Click
    Else
        sSql = "UPDATE CPC SET Chave_Pub = " & FormatNumber(txtChvPub.Text, 0, vbFalse, vbFalse,
vbFalse) & " , Data_Alt = '" & Now & "' WHERE Usuario = '" & txtUsuario.Text & "'"
        bd.Execute sSql

        MsgBox "Atualização efetuada com sucesso.", vbOKOnly + vbInformation
        cmdLimpar_Click
    End If

   Exit Sub
Erro:
    MsgBox Err.Description, vbOKOnly + vbCritical, "Erro"


End Sub

Private Sub cmdCalcule_Click()

Dim mRecordset As ADODB.Recordset

On Error GoTo Erro

If txtChvPriv.Text = "" Then

    MsgBox "Você deve preencher o campo destinado a chave privada.", vbCritical, "Cadastro de Chave"
    Exit Sub

  Else
      If txtChvPriv.Text < 2 Then
          MsgBox "Você deve preencher o campo destinado a chave privada com," + vbNewLine + "no
mínimo, 4 dígitos .", vbCritical, "Cadastro de Chave"
          txtChvPriv.SetFocus
```

```
            Exit Sub
        Else
            Set mRecordset = New ADODB.Recordset
            mRecordset.ActiveConnection = bd
            mRecordset.Source = "SELECT * FROM Constantes WHERE Primo_N"
            mRecordset.Open

            raiz = mRecordset.Fields("Raiz_Primitiva")
            Z = raiz ^ CDbl(txtChvPriv.Text)
            N = mRecordset.Fields("Primo_N")
            Z = Int(Z + 0.5)
            N = Int(N + 0.5)
            calcular = Fix(Z / N)
            resultado = Z - (N * calcular)
            txtChvPub.Text = resultado
            lblChvPublica.Visible = True
            txtChvPub.Visible = True
        End If
    End If

Exit Sub

Erro:
    MsgBox "Este cálculo não pode ser processado, o resultado é um número muito grande.", vbOKOnly + vbCritical, "Erro de Overflow"


End Sub

Private Sub cmdLimpar_Click()
 txtChvPriv.Text = ""
 txtChvPub.Visible = False
 txtChvPriv.SetFocus
End Sub

Private Sub cmdSair_Click()
  txtUsuario.Enabled = True
  Unload Me

End Sub

Private Sub Form_Load()
Me.Height = 2415
Me.Width = 4500
Me.Top = 2000
Me.Left = 3000

txtUsuario.Text = Mid(MDIPrincipal.StatusBar.Panels(1)(), 12, 10)
txtUsuario.Enabled = False

cmdCadastrar.Enabled = False
Set bd = New ADODB.Connection
bd.ConnectionString = "DSN=Chave"
bd.Open


End Sub

Private Sub txtChvPriv_Change()
```

```
  If txtChvPriv.Text = "" Then
    cmdCadastrar.Enabled = False
  Else
    cmdCadastrar.Enabled = True
  End If

End Sub
```

Form do MDIPrincipal

```
Private Sub mnuConstante_Click()
  frmAltConstante.Show 0
End Sub

Private Sub mnuLogin_Click()
  newlogin = True
  frmLogin.Show 0
  If newlogin <> True Then Unload Me
End Sub

Private Sub mnuSair_Click()
  End
End Sub

Private Sub mnuUsuário_Click()
  frmCadastroUsuarios.Show 0
End Sub

Private Sub ToolBarChave_ButtonClick(ByVal Button As MSComctlLib.Button)

On Error GoTo Erro

   Select Case Button.Key
      Case "FunPub": frmSisGerChave.Show 0
      Case "FunPriv": frmCPC.Show 0
      Case "Secreta": frmChaveSecreta.Show 0
      Case "SAFER":  Shell "C:\COMMAND.COM", vbMaximizedFocus
      Case "Maple": Shell "C:\MAPLEV4\BIN.WIN\WMAPLE32.EXE", vbNormalFocus
      Case "Sair": mnuSair_Click
   End Select

Exit Sub

Erro:
   If Button.Key = "SAFER" Then
        MsgBox "Não foi possível iniciar o SAFER, verifique o caminho do prompt do DOS.", vbOKOnly +
vbCritical, "Erro de Patch"
   Else
     If Button.Key = "Maple" Then
        MsgBox "Não foi possível iniciar o Maple, verifique se o programa está instalado no seu micro.",
vbOKOnly + vbCritical, "Erro de Patch"
     End If
```

```
    End If

End Sub
```

Form do Chave Secreta

```
Option Explicit
Dim bd As ADODB.Connection
'Variável para cálculo da Chave Secreta
Public Y As Double
Public W As Double
Dim K As Double
Public N As Double

Private Sub cmdCalcular_Click()

Dim resultado As Double
 On Error GoTo Erro

  W = Y ^ CDbl(txtChavPri.Text)
  W = Int(W + 0.5)
  N = Int(N + 0.5)
  K = Fix(W / N)
  resultado = W - (N * Fix(W / N))
  txtChavSec.Text = resultado

Exit Sub

Erro:
    MsgBox "Este cálculo não pode ser processado, o resultado é um número muito grande.", vbOKOnly +
vbCritical, "Erro de Overflow"
End Sub

Private Sub cmdLimpar_Click()

  Y = Empty
  txtChavPri.Text = ""
  txtChavSec.Text = ""
  datcobPub.Text = ""

End Sub

Private Sub cmdSair_Click()
  Unload Me
End Sub

Private Sub datcobPub_Change()
Dim mRecordset As ADODB.Recordset
Dim mRecordset1 As ADODB.Recordset

  If datcobPub.Text <> "" Then
    Set mRecordset = New ADODB.Recordset
    mRecordset.ActiveConnection = bd
    mRecordset.Source = "SELECT * FROM Constantes"
    mRecordset.Open
```

```
    Set mRecordset1 = New ADODB.Recordset
    mRecordset1.ActiveConnection = bd
    mRecordset1.Source = "SELECT * FROM CPC WHERE Usuario = '" & datcobPub.Text & "'"
    mRecordset1.Open

    Y = mRecordset1.Fields("Chave_Pub")
    N = mRecordset.Fields("Primo_N")
  End If

End Sub

Private Sub Form_Load()


  Set bd = New ADODB.Connection
  bd.ConnectionString = "DSN=Chave"
  bd.Open

  Me.Height = 2415
  Me.Width = 4680
  Me.Top = 2000
  Me.Left = 3000
End Sub
```

Módulo do Programa

```
'Para Saber o Nível do Usuário
Global sNivel As String
'Para Saber o Login do Usuário
Global sLogin As String

' Rotina para evita que o programa abra mais de uma vez
Sub main()

Dim sTitulo As String

    If App.PrevInstance = True Then

        sTitulo = App.Title
        App.Title = "... segunda chamada ao mesmo programa."
        AppActivate sTitulo
        SendKeys "% R", True
        End

    Else

        frmApres.Show 1

    End If

End Sub
```

REFERÊNCIAS BIBLIOGRÁFICAS

[1] MEIRELLES, Hely Lopes. *Direito Administrativo Brasileiro.* Malheiros Editores, 19ª edição, São Paulo, 1994.

[2] SILVA, José Afonso da. *Curso de Direito Constitucional Positivo.* Malheiros Editores, 9ª edição, São Paulo, 1994.

[3] MORAES, Alexandre de. *Direito Constitucional.* Atlas S. A ., 5ª edição, São Paulo, 1999.

[4] CARVALHO FILHO, José dos Santos. *Manual de Direito Constitucional.* Lúmen Juris, 3ª edição, Rio de Janeiro, 1999.

[5] REIS, Maria Helena Junqueira. "Crime Informático" [s.d]

[6] BRASIL, *Constituição da República Federativa do Brasil*. Promulgada em 05 de Outubro de 1988, 1ª edição, São Paulo, Rideel, 1992.

[7] BERNSTEIN, Terry; BHIMANI, Anish B.; SIEGEL, Carol A . e SCHULTZ, Eugene. *Segurança na Internet.* Editora Campus, 1997.

[8] SOARES, Luiz Fernando Gomes; LEMOS, Guido e COLCHER, Sérgio. *Redes de Computadores*. Editora Campus, 2ª Edição, 1995.

[9] INTERNATIONAL Organization for Standardization / International Eletrotechnical Committee. " Information Processing Systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture". International Standard 7498-2, 1989.

[10] KAUFMAN, Charlie; PERLMAN, Radia e SPECINER, Mike. *Network Security.* Prentice Hall, 1995.

[11] TANENBAUM, Andrew S. *Computer Networks.* 3ª edição, Prentice Hall, 1996.

[12] *Segurança Máxima: o guia de um hacker para proteger seu site na Internet e sua rede.* Editora Campus, 2000.

[13] ROCHA JR., Valdemar Cardoso da. "Uma Introdução à Criptografia". XI Simpósio Brasileiro de Telecomunicações, Texto de Minicurso. Natal - RN, Maio, 1993.

[14] LEITE, Maria Marta. *Princípios de redes de Computadores.* Florianópolis, 1996.

[15] SGARRO, Andrea. *Códigos Secretos*. Editora Melhoramentos, 1994.

[16] DENNING, Dorothy Elizabeth Robling. *Cryptography and Data Security.* Addison Wesley, 1982.

[17] MASSEY, J. L. "An Introduction to Contemporary Cryptology". *Proceeding of the IEEE*, vol. 76, n. 5, pp. 533-549, Maio 1988.

[18] KRUH, L. "The Churchyard Ciphers" *Cryptologia* , vol. 1(4), pp. 372-375, Out. 1977.

[19] SAM, E. – "Musical Cryptography" *Cryptologia*, vol. 3(4), pp. 193-201, Out. 1979.

[20] WARD, J. B. *The Beale Papers*. Pamphlet printed by Virginian Book and Job Print, reprinted by The Beale Cypher Assoc., Medfield, Mass, 1885.

[21] ASSOC, The Beale Cypher. "The Beale Ciphers".Medfield, Mass, 1978.

[22] HAMMER, C. "Signature Simulation and Certain Cryptographic Codes" Comm. ACM vol. 14(1) pp 3-14, Jan. 1971.

[23] KAHN, D. *The Codebreakers, Macmillan Co.* New York ,1967.

[24] SHANNON, C. E. "Communication Theory of Secrecy Systems," Bell Syst. Tech. J. Vol. 28 pp. 656-715, Out. 1949.

[25] W. DIFFIE e M. HELLMAN. *"New Directions in Cryptography",* IEEE Transactions on Information Theory, vol. IT-22, n.6, pp. 644-654, NOV. 1976.

[26] NEWMAN, D. B., Jr e R. L. PICKHOLTZ. *"Cryptography in the Private Setor",* IEEE Communications Magazine, vol. 24, n. 8, pp. 7-10, Ago.1986.

[27] S. POHLIG e M. HELLMAN. "An Improved Algorithm for Computing Logarithms over GF (p) and its Cryptographic Significance", IEEE *Transactions on Information Theory,* vol. IT-24, n. 1, pp. 106-110, Jan.1978.

[28] RIVEST, R. L., A . SHAMIR e L. ADLEMAN. "A Method for Obtaining Digital Signatures and Public-Key Criptosystems", *Communications of the ACM*, vol. 21, n.2, pp. 120-126, Fev. 1978.

[29] W. DIFFIE e M. HELLMAN. "Privacy and Authentication: An Introduction to Cryptography", *Proc. IEEE,* vol. 67 (3) pp. 397-427, Mar. 1979.

[30]OMURA, J. K., "Novel Applications of Cryptography in Digital Communications", *IEEE Communications Magazine*, pp. 21-29, Maio 1990.

[31] DAVIES, D. W.; W. L. PRICE, "Security for Computer Networks", John Willey & Sons, Inc., New York, 1989.

[32] H. FEISTEL, "Cryptography and Computer Privacy", Sci. Am., vol. 228, n. 5, pp. 15-23, Maio 1973.

[33] SCHNEIER, Bruce. *Applied Cryptography.* John Wiley & Sons, 2ª edição, 1996.

[34] PETE LOSHIN and PAUL MURPHY. "Electronic Commerce: On-Line Ordering and Digital Money", Charles River Media, 2nd edition, Ago., 1997.

[35] MASSEY, James L. *Cryptography: Fundamentals and Applications.* ATS Seminars, Zurich, 1995.

[36] MASSEY, James L. "SAFER K-64: A Byte-Oriented Block-Ciphering Algorithm". pp.1-17 in Fast Software Encryption (Ed. R. Anderson), Lecture Notes in Computer Science No. 809. New York: Springer, Jan. 1994.

[37] MASSEY, James L. "SAFER K-64: One Year Later". Presented at the K. U. Leuven Workshop on Algorithms, Leuven, Belgium, 14-16 December, 1994, and to appear in Fast Software Encryption II (Ed. B. Preneel), to be published by Springer-Verlag in the Lecture Notes in Computer Science Series, Abr. 1995.

[38] MOLINER, Richard de. (org.) "Organização do Manual do SAFER". [s.d]

[39] MASSEY, James L. "Announcement of a Strengthened Key Schedule for the Cipher SAFER". ETH Zurich, Set., 1995.

[40] ROCHA JR., Valdemar Cardoso da. "SAFER +" , Seminário do Grupo de Pesquisa em Comunicações – CODEC, Depto. De Eletrônica e Sistemas – UFPE, 1999.

[41] LEITE, Maria Marta. "Princípios de Redes de Computadores", Universidade Federal de Santa Catarina, Depto. De Informática e Estatística. Florianópolis, 1996.

[42] CARVALHO, T.C.M.B. et alli. *Arquiteturas de Redes de Computadores: OSI e TCP/IP.* 1ª ed. São Paulo. Makron Books/BRISA, 1994.