UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E DE ESTATÍSTICA
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

ARQUITETURA DE SEGURANÇA PARA REDES APLICADA A SISTEMAS DE GERÊNCIA

por

José Eduardo De Lucca

Dissertação submetida como requisito parcial
para a obtenção do grau de
Mestre em Ciências da Computação

Carlos Becker Westphall
Orientador

Florianópolis, junho de 1995

ARQUITETURA DE SEGURANÇA PARA REDES
APLICADA A SISTEMAS DE GERÊNCIA

JOSÉ EDUARDO DE LUCCA

ESTA DISSERTAÇÃO FOI JULGADA ADEQUADA PARA A OBTENÇÃO DO TÍTULO DE

MESTRE EM CIÊNCIAS DA COMPUTAÇÃO

ESPECIALIDADE SISTEMAS DE COMPUTAÇÃO E APROVADA EM SUA FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO DA UNIVERSIDADE FEDERAL DE SANTA CATARINA.

ORIENTADOR: Carlos Becker Westphall, Dr.

COORDENADOR DO CURSO: Rogério Cid Bastos, Dr.

BANCA EXAMINADORA

PRESIDENTE: Carlos Becker Westphall, Dr.

Elizabeth Sueli Specialski, Msc.

Liane Margarida Rockenbach Tarouco, Dra.

Bernardo Gonçalves Riso, Dr.

SUMÁRIO

LISTA DE ABREVIATURAS

ACSE	*Association Control Service Element*
APDU	*Application Protocol Data Unit*
API	*Application Process Interface*
ASCII	*American Standard Code for Information Interchange*
CCITT	*Comitè Consultatif International de Télégraphique et Téléphonique*
CMIP	*Commom Management Information Protocol*
CMISE	*Commom Management Information Service Element*
CPU	*Central Processing Unit*
CRC	*Cyclic Redundancy Check*
DECNet	*Digital Equipments Company Network*
DES	*Data Encryption Standard*
DIS	*Draft International Standard*
DP	*Draft Proposal*
FTAM	*File Transfer, Access and Management*
IMP	*Interface Message Processor*
ISO	*International Organization for Standardization*
IPC	*InterProcess Communication*
ITU	*International Telecommunications Union*
MHS	*Message Handling Service*
MD5	*Message Digest-5*
MIB	*Management Information Base*
OSI	*Open Systems Interconnection*
RM-OSI	*Reference Model - Open System Interconnection*
ROSE	*Remote Operations Service Element*
RFC	*Request For Comments*
RPC	*Remote Procedure Call*
RSA	*Rivest, Shamir & Adleman Algorithm*
SNA	*System Network Architecture*
SNM	*SunNet Manager*
TCP/IP	*Transmission Control Protocol/Internetwork Protocol*

LISTA DE FIGURAS

RESUMO

A gerência em redes de computadores tornou-se uma necessidade primária, em função da crescente complexidade que ambientes em rede vêm alcançando. A necessidade de possuir uma rede confiável com comunicações seguras e secretas é um dos motivos que conduzem ao estudo sobre segurança em redes de computadores.

Um Sistema de Gerência de Redes e as informações por ele geradas são extremamente sensíveis e, se expostas a ameaças, apresentam vulnerabilidades que podem comprometer seriamente a segurança da rede e seus sistemas como um todo, afetando a estabilidade, a confiabilidade e o uso das redes.

A proposta deste trabalho é apresentar soluções para estes problemas de modo a garantir que a Gerência da Rede não seja um dos pontos expostos a agressores. Para tanto é definido um Modelo Genérico de Arquitetura de Segurança para ser aplicado sobre uma Plataforma Genérica de Gerenciamento de Redes.

ABSTRACT

Computer network management has became a primary necessity, as networking environment complexity is arising. The necessity of a trusted network with secure and secret communication is motivation to studies in computer network security.

A Network Management System and the information generated by it are highly sensitive and, if they are exposed to threats, they show vulnerabilities which can seriously endanger network and systems security, affecting network stability, thrust and usefulness.

The purpose of this work is to present solutions to those problems, to avoid that Network Management be a exposition point to attacks. To do this, a Generic Security Architecture Model is defined and applied on a Network Management platform.

1 Introdução

Segurança em informática pode ser considerada como a garantia ou confiança que os usuários têm em determinado sistema [PFL 89] [DOD 83]. O problema de segurança em computadores (em geral) e em redes de computadores (mais especificamente) é um dos mais sérios que precisam ser resolvidos em ambientes informatizados, uma vez que computadores e redes representam um papel fundamental para estes ambientes. A confiabilidade destas redes é, então, fator preponderante para garantir a produtividade dos sistemas que compõem estas instalações.

Dentro do escopo das Redes de Computadores pode-se definir que garantir a segurança destes sistemas é garantir que os mesmos não serão comprometidos por ataques sobre vulnerabilidades criadas por elementos introduzidos nas redes de computadores, tais como meios de comunicação (cabos, ondas eletromagnéticas, ...), equipamentos de comunicação (modems, roteadores, placas, ...) e software de comunicação/rede (protocolos e aplicações, por exemplo).

A confiabilidade necessária deve estar alicerçada em diversos pontos. Dois dos pontos mais importantes a serem considerados são o gerenciamento dos recursos e a proteção dos elementos que compõem as redes. O termo proteção abrange vários tópicos que visam impedir basicamente o acesso a dados/informações por parte de pessoas não autorizadas. Este acesso pode ser apenas para leitura ou para modificações e até destruição de informações.

Com a evolução da informática e da importância estratégica das atividades relacionadas com informática no desempenho de empresas e instituições em geral, os aspectos ligados à segurança passaram a representar papel importante no cotidiano das mesmas. A migração de ambientes centralizados (*time-sharing*) e de micro-computadores independentes para instalações interligadas por redes de computadores resultaram em novos e sérios problemas de segurança.

A segurança tradicional em ambientes centralizados era basicamente física, com restrições de acesso às instalações e equipamentos. Dados fluindo pelas redes ou residindo temporariamente em nós intermediários são vulneráveis à interceptação, alteração ou destruição. Existem mais formas de executar uma tentativa de acesso a computadores conectados a uma rede do que a computadores isolados. Os problemas de segurança de tais ambientes se complicam com a tendência à grande distribuição física dos recursos e dos usuários. Outro complicador refere-se aos diferentes níveis de segurança física encontrados em cada um destes locais.

Além das ameaças representadas por adversários e ameaças naturais, a falta de políticas claras de segurança, o empirismo sendo adotado como forma de definição dos mecanismos a implantar e a displicência no tratamento da segurança são também fatores que comprometem os sistemas.

A exposição dos sistemas às ameaças se torna cada vez maior à medida que eles integram uma comunidade mais heterogênea através do crescimento das redes e do acesso a sistemas por linha discada. Em conseqüência de todo o quadro apresentado, se torna premente a fixação de políticas, o embasamento teórico e a criação e adoção de mecanismos de segurança para proteger as instalações de possíveis ataques.

Protocolos de Gerência de Redes e os canais de comunicação que transportam informações de gerência são potencialmente vulneráveis a atentados contra a segurança. Cuidados particulares devem, portanto, ser tomados para assegurar que tais protocolos e informações estejam protegidos. A definição de vulnerabilidades e dos riscos de segurança dos Sistemas de Gerência e a criação de ferramentas para tratar estes problemas fazem parte do conjunto de ações fundamentais para o funcionamento confiável das redes. A especificação de ferramentas, seu comportamento e seus inter-relacionamentos compõem uma Arquitetura de Segurança, que é a proposta deste trabalho.

O trabalho está dividido em capítulos onde são apresentados os problemas

relacionados com Segurança e Gerência de Redes, iniciando por Segurança em Redes de Computadores (capítulo 2), criando uma base comum para as discussões que são apresentadas nos capítulos seguintes. A Gerência de Redes é apresentada no capítulo 3, abordando suas vulnerabilidades, as ameaças às quais está sujeita e requisitos de segurança para a Gerência. A proposta da Arquitetura Genérica de Segurança para Gerência de Redes vem a seguir (capítulo 4), onde são descritos o modelo e os mecanismos idealizados. No capítulo 5 é realizada uma discussão sobre os aspectos da implementação realizada, descrevendo as opções feitas e as restrições consideradas, bem como os algoritmos finais utilizados e avaliações realizadas. Acrescentou-se também um anexo com uma descrição mais detalhada de algoritmos de criptografia.

2 Segurança em Redes de Computadores

Neste capítulo, é apresentada uma base comum para discussão que envolve definições e considerações sobre Segurança nas Comunicações e nas Redes de Computadores. Algumas abordagens possíveis para análise de problemas de segurança são discutidas, e apresentados os serviços de segurança conforme a visão tradicional. Os aspectos de segurança do Modelo de Referência OSI encerram este capítulo.

2.1. Motivação

Com a crescente dependência que qualquer atividade tem frente aos sistemas informatizados, possuir segurança em relação aos dados processados em computadores pode ser o diferencial entre manter os sistemas em funcionamento ou paralisá-los em função de tentativas de quebra da segurança e perda da confiabilidade dos sistemas.

O acesso às facilidades de comunicação de dados tais como: transferência de arquivos, *login* remoto, acesso a bases de dados remotas, correio eletrônico e compartilhamento de recursos escassos via redes é um direito e uma necessidade cada vez maior do usuário. Mas também é uma preocupação a mais para os gerentes de rede e especialistas em segurança, uma vez que estes serviços trouxeram novos e sérios problemas de segurança aos sistemas. No princípio da criação e difusão das redes de computadores havia a necessidade de se desenvolver uma tecnologia de baixo custo, que atendesse as necessidades prementes de comunicação, e que fosse simples o suficiente para que se difundisse rapidamente. Isto de fato ocorreu, mas a simplicidade e o baixo custo excluíram os investimentos em tecnologia de segurança para as redes. O que gerou toda esta carga de novos problemas foi a falta de investimento em segurança compatível e nos mesmos moldes em que se investiu em tecnologia básica de redes.

A disponibilidade de recursos de comunicação citados anteriormente, os diferentes níveis de segurança em sistemas interligados, a generalização do acesso e a

disseminação dos micro-computadores dotados de capacidade de comunicação ampliou em muito o espectro de possibilidades de ataques a sistemas via comunicação.

Os dispositivos de segurança em redes de computadores objetivam basicamente:

- garantir ao usuário autorizado o acesso aos recursos desejados de modo confiável, seguro e confidencial;
- garantir a integridade dos dados que fluem na rede;
- garantir a identificação dos interlocutores;
- impedir que usuários não autorizados se beneficiem de recursos/informações;
- impedir a alteração ou destruição indevidas de informações e
- garantir a disponibilidade de recursos para usuários legítimos.

2.2. Definições e Premissas

A principal definição que se faz necessária diz respeito aos elementos sobre os quais essas ameaças podem incidir em um ambiente informatizado e, em especial, em um ambiente em redes de computadores. Como foi dito, a Segurança aplicada ao domínio das Redes de Computadores deve garantir que o sistema não se torne comprometido por ameaças cuja origem esteja localizada em entidades introduzidas pelas redes como: cabos, interfaces, equipamentos específicos, protocolos e aplicações de rede.

As potenciais consequências de um comprometimento de um sistema informatizado poderiam ser, no limite, a indisponibilidade total do sistema, por algum motivo como equipamento avariado, código removido, CPU a 100% de carga, etc. Outras formas não tão facilmente detectáveis de comprometimento de um sistema (mas não menos preocupantes) são o fornecimento de resultados incorretos em um processamento ou a execução de outras atividades que não somente aquelas esperadas, bem como a apropriação de informações por pessoas não autorizadas.

Existem muitas formas de comprometer sistemas porque normalmente existem muitos pontos expostos. Estes pontos de exposição podem ser classificados conforme as seis categorias abaixo:

- Hardware: CPUs, placas, teclados, terminais, estações de trabalho, computadores pessoais, impressoras, discos, linhas de comunicação, roteadores, modems, etc;

- Software: fontes, códigos-objeto, utilitários, sistemas operacionais, programas de comunicação, etc;

- Informação: armazenadas (*on-line* e *off-line*), em memória durante execução, bases de dados, *backups*, *logs*, dados em trânsito em linhas de comunicação, etc;

- Pessoal (usuários e pessoal operacional);

- Documentação: sobre software, sobre hardware, procedimentos administrativos, etc;

- Suprimentos: papel, mídia magnética e ótica, etc.

No jargão de segurança estes itens compõem o que se chama ***ativos***. Os ativos de uma instalação devem ser protegidos de ameaças, porque o uso apropriado desses ativos é que vai permitir o bom funcionamento dos sistemas. Uma alteração, destruição, mal-funcionamento, erro ou indisponibilidade de algum destes ativos pode gerar um comprometimento do sistema.

As **vulnerabilidades** dos ativos devem, portanto, ser estudadas, desde que o **risco** oferecido por tal vulnerabilidade seja considerado (potencialmente) prejudicial e medidas de proteção devem ser adotadas para compensá-las.

De acordo com o modelo de referência OSI/ISO, a Segurança está relacionada com a "minimização das vulnerabilidades dos ativos e dos recursos". A ISO define

vulnerabilidade como fraquezas "que podem ser exploradas para a violação de sistemas ou informações lá contidas". Uma ameaça pode ser "uma potencial violação de segurança" ou ainda "uma condição ou ação que compromete a segurança de uma rede". [ISO 89]

Os pontos vulneráveis de um sistema podem se apresentar como problemas de projeto, erros de configuração ou procedimentos, e a eles estão associados os riscos à segurança. São estes pontos vulneráveis que devem ser examinados e sobre eles tomadas medidas de salvaguarda de modo a minimizar (se não eliminar) os riscos de uma quebra de segurança. Exemplos de medidas de segurança são a realização de backups, controle de acessos, criptografia etc.

Não há sistema invulnerável ou perfeitamente seguro - todos são passíveis de problemas de quebra de segurança. Quando uma quebra da segurança ocorre, deve-se aprender com ela, para que medidas corretivas sejam tomadas evitando que o mesmo problema venha a ocorrer novamente. Além disso, um processo de recuperação (ação pós-fato) deve ser imediatamente iniciado, objetivando reduzir os prejuízos causados pelo problema. Exemplos de processos de recuperação são a reinstalação de backups, a auditoria, a instalação de processos jurídicos etc.

Em resumo, uma abordagem possível para tratar os problemas de segurança em geral e aplicável à informática é a seguinte:

1. identificar os itens a proteger (ativos);

2. identificar as ameaças a estes itens;

3. examinar as vulnerabilidades e riscos;

4. estabelecer medidas protetoras;

5. estabelecer respostas (recuperação).

Por outro lado, os custos, uma eventual sobrecarga ao sistema que as medidas

de segurança causarão e a inconveniência ao usuário legítimo são alguns dos contrapontos da necessidade de segurança; de onde decorre a necessidade de um bom balanceamento entre os benefícios e as dificuldades.

Já está sedimentado em todos os meios que, além dos recursos financeiros, os principais ativos do mundo de hoje são a informação, as instalações e o pessoal. Apenas as perdas relacionadas com informações serão consideradas, pois os demais fogem do escopo deste trabalho. De um modo geral, a forma como estas perdas podem se dar encontram-se relacionadas abaixo:

- pela destruição das informações,
- pela modificação de informações e
- pela apropriação, por parte de pessoas não autorizadas, de informações importantes e/ou sensíveis.

Analisando os ativos apresentados a partir do escopo de segurança em redes de computadores, apenas Hardware, Software e Informação são passíveis de serem tratados neste ambiente, uma vez que o equacionamento das três categorias restantes (Pessoal, Documentação e Suprimentos) só é possível a partir de abordagens que extrapolam o objeto deste estudo. Isto porque a categoria Suprimentos diz respeito a atividades administrativas e segurança física; o mesmo se aplica à categoria Documentação (que é de responsabilidade do setor operacional e cuja integridade deve ser mantida por medidas de segurança física).

Já os ativos da categoria Pessoal devem ser encarados como aqueles responsáveis pela instalação (pessoal operacional - que deve manter o sistema em funcionamento: equipamentos, comunicação, sistema operacional) e os usuários legítimos (ou seja, aqueles com direito de acesso e interessados em ver seus sistemas executando e responsáveis pela entrada de dados, por exemplo).

Mesmo dentro destas três categorias, muitos problemas não são passíveis de

solução a partir do que se convenciona chamar de **Segurança Lógica**, mas apenas por **Segurança Física**. Por Segurança Física se entende a segurança tradicional de informática, basicamente apoiada no controle de acesso físico às instalações e equipamentos e prevenção de acidentes de trabalho, bem como planos de recuperação de desastres como incêndios e inundações. Com a evolução da informática e da importância estratégica das atividades relacionadas com informática no desempenho de empresas e instituições em geral, a Segurança Física passou a abranger também questões como destruição do lixo gerado (formulários e mídia magnética) e "contra-espionagem industrial" (varredura de escutas, p.ex.). Maiores informações podem ser encontradas em [PFL 89] e [COO 89].

Segurança Lógica é considerada a barreira seguinte à Segurança Física, uma vez que objetiva impedir o acesso aos sistemas (Hardware/Software/Dados) a usuários não autorizados que, de alguma forma, possuem meios de alcançá-los. Para tanto, lança mão de software para garantir quatro princípios básicos: autenticação de usuário, disponibilidade de recursos, integridade das informações e confidencialidade das informações.[1]

2.3. Ameaças à segurança das comunicações

A interligação de recursos por equipamentos de comunicação e redes abre dois flancos que devem ser observados:

- ataques à comunicação; e
- ataques ao sistema.

A primeira forma de ataque tem por objetivo a apropriação ou alteração de informações, ou ainda dificultar ou impedir a comunicação. A segunda forma também visa a apropriação ou alteração das informações, bem como a destruição das mesmas ou tornar recursos não disponíveis. As ameaças relacionadas com estes ataques são as seguintes:

---

[1] *Cabe ressaltar que a segurança física está fora do escopo deste trabalho, mas se não houver esta forma de segurança, a Segurança Lógica pouco poderá fazer para garantir seus princípios básicos e a confiabilidade dos sistemas.*

- Interceptação de informações: por escuta (dito ataque passivo) ou quando um intruso se faz passar por outros usuários (mascaramento), requisitando serviços, acessando informações ou se comunicando com outros usuários. Uma solução para o problema de escuta é a criptografia dos dados sensíveis e para o intruso, um bom sistema de controle de acesso e autenticação de usuários. Um esquema deste tipo de agressão é visto na figura 2.1.

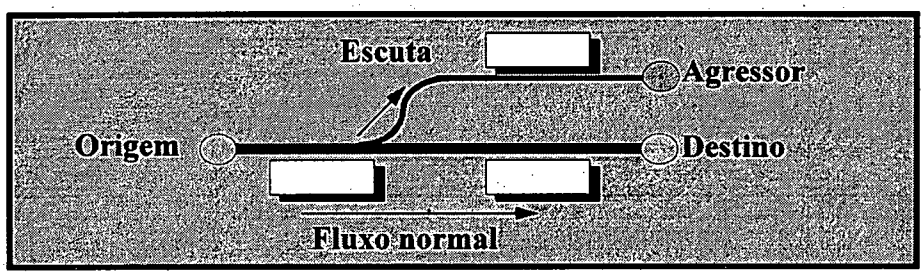

FIGURA 2.1: Interceptação de Mensagens

- Alteração de informações: da mesma forma que um intruso pode acessar informações, ele pode também modificá-las (quer seja para tirar proveito quer seja para comprometer os sistemas). Novamente a solução seria um controle de acesso mais rígido. Duplicação ou re-seqüenciamento de pacotes e alterações no corpo das mensagens são outras formas de ataque (sem necessidade de invasão, uma vez que podem ser realizadas em nós intermediários da rede). Esses métodos de ataque que podem ser evitados usando técnicas de criptografia em campos específicos das mensagens/pacotes. Na figura 2.2 há um esquema que representa esta agressão.

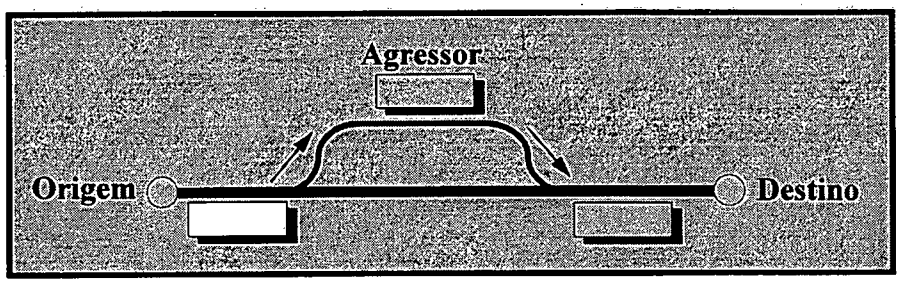

FIGURA 2.2: Alteração de Mensagens

- Destruição de informações: as mesmas técnicas de ataque utilizadas para alteração de

informações são usadas aqui. A figura 2.3 representa este tipo de ataque.

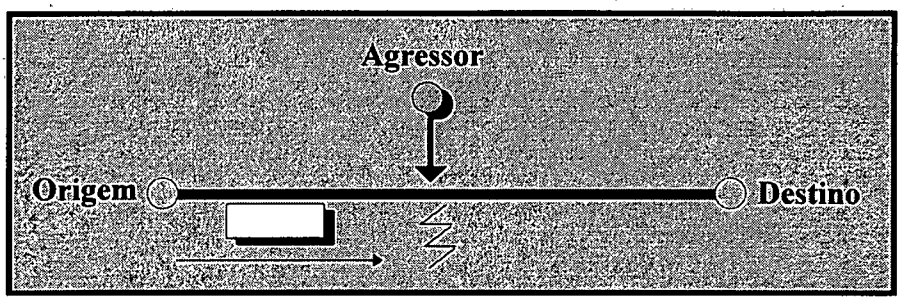

FIGURA 2.3: Destruição de Mensagens

- Mistura de sinais: esta é uma ameaça onde criptografia e controle de acesso não têm aplicabilidade. Objetivando dificultar ou impedir a comunicação, um adversário pode utilizar-se de técnicas de *jamming* (mistura). Para evitá-las são necessários cuidados como seleção do melhor canal ou mudança de freqüência constante. Esta ameaça é especialmente forte para comunicações por satélite e ondas de rádio e pode ser representada conforme a figura 2.4.

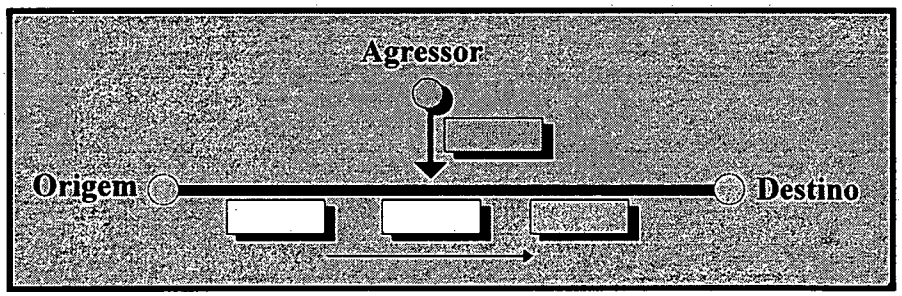

FIGURA 2.4: Mistura de Sinais

- Problemas físicos com equipamentos de comunicação: se um problema deste tipo ocorre (como o corte de linhas de comunicação, p.ex.), pode ocorrer interrupção da comunicação. As formas de se precaver desta forma de ataque são a manutenção de equipamentos de *backup* e rotas alternativas.

Os ataques relacionados com mistura de sinais e problemas físicos não são diretamente tratáveis por um sistema de segurança lógico (software e hardware) e portanto não serão aprofundados aqui. As outras ameaças, relacionadas com invasão de sistemas,

escuta, mascaramento e integridade dos dados são os que podem ser equacionados por serviços de segurança que uma rede pode fornecer diretamente, os quais serão detalhados mais adiante.

2.4. Agressões e Falhas

Outra forma (complementar) de analisar os problemas de segurança é fazer uma classificação das ameaças entre **agressões** e **falhas**, como referenciado em [COO 89]. Apesar das diversas facetas que estas ameaças apresentam, podemos dividi-las em dois grupos:

- **falhas**: são acontecimentos acidentais ou não, onde se enquadram acidentes naturais (como incêndios, inundações, terremotos, ...), ataques de roedores, acidentes ou erros humanos (café, manipulação incorreta de mídia, entrada de informações incorretas, ...) e falhas de hardware e software (*bugs*, erros de comunicação, falhas em periféricos, ...). As falhas são ameaças à segurança das instalações e sistemas porque, como já foi definido, atentam contra a confiabilidade e/ou disponibilidade de um sistema.

- **agressões**: ameaças físicas (bomba, fogo, contra pessoas, roubo com invasão, ...), ameaças lógicas internas (operação inadequada, software propositalmente incorreto,...) e ameaças lógicas externas (vírus, penetras - *hackers* e *crackers*, etc).

Como é possível perceber, agressões são sempre intencionais e, de uma forma ou de outra, hostis. Evidentemente as agressões são problemas que devem ser devidamente equacionados por medidas de segurança eficientes. Os agressores visam a obter benefícios indevidos (financeiros na maioria das vezes) ou apenas prejudicar; e são sempre engendrados por pessoas, seja por descontentamento, por vingança, pelo 'desafio', espionagem/sabotagem, por indiscrição, ou outros motivos da natureza humana. Também aqui se enquadram as ações que, sem intenção delituosa, acabam por comprometer a privacidade de pessoas (agressão moral e legal) e/ou a confiabilidade do sistema, das informações tratadas pelo sistema ou ainda a confidencialidade das informações manipuladas.

Da mesma forma, que ocorreu na abordagem anterior, também aqui não é possível abordar todos os problemas a partir do enfoque de segurança em redes de computadores. Todos os problemas relacionados às falhas somente dizem respeito a aspectos como manutenção e segurança física, p.ex., e não Segurança Lógica. Na lista de agressões também se encontram diversas ameaças que não são tratáveis por medidas de segurança lógica, como ameaça de bomba e roubo.

Então, sob esta ótica, as ameaças que dizem respeito à segurança em redes de computadores são agressões perpetradas por pessoas não autorizadas (as quais serão chamadas invasores ou adversários) que objetivam obter benefícios indevidos ou apenas prejudicar o funcionamento dos sistemas.

2.5. Acesso à Informação e à Capacidade de Processamento

Também é possível analisar as ameaças potenciais às quais os sistemas em rede estão sujeitos a partir das vantagens que se pode obter com a invasão de sistemas. Algumas vantagens[1] seriam:

- o acesso a informações,
- a obtenção/roubo de "ciclos de máquina",
- o acesso a serviços e a outras redes,
- o acesso a software,
- ganho de espaço de armazenamento e
- destruição/modificação de informações (ganhos indiretos, em função de prejuízos a terceiros).

Em suma, o que está em 'disputa' neste contexto de proteção de Sistemas em Rede pode ser resumido nestes dois itens:

---

[1] *Vantagens aqui devem ser entendidas como um ou mais dos seguintes "motivadores": aspecto financeiro, sabotagem/espionagem (industrial, p.ex.), vingança, curiosidade, desafio, diversão.*

1. A **informação** em si: o acesso, a destruição e a modificação de informação e o acesso a serviços; e

2. O acesso à capacidade de **processamento** de informação e ao equipamento: roubo de ciclos de máquina, acesso a serviços, redes e software, e uso da capacidade de armazenamento.

Por informação deseja-se representar muitas coisas: dados para processamento, tecnologia, *know-how*, conhecimento científico, informações econômico-financeiras, estratégias e políticas, projetos, etc. Computadores podem manter informações confidenciais sobre pessoas, sobre objetivos militares e movimentos de tropas, informações vitais para empresas ou governos, saldos bancários, e assim por diante. O valor destas e de outras informações é alto, apesar de ser muito difícil, na maioria dos casos, estabelecer o valor intrínseco de determinada informação. Uma abordagem possível nestes casos é determinar que o valor de determinada informação é igual ao custo de obtenção de determinados dados (como por exemplo o resultado de anos de pesquisas em laboratório sintetizado em artigos) ou ainda da recuperação ou reconstrução das informações perdidas. Outro tipo de informação é a que diz respeito a pessoas (ou mesmo entidades) onde o acesso a estas informações pode configurar um atentado à privacidade.

Então, apesar de as informações possuírem um alto valor, é freqüentemente difícil medi-lo (pela subjetividade da questão). Pode-se estabelecer apenas que o detentor da informação conhece o valor da mesma. A capacidade de acesso à informação, bem como a capacidade de alterá-la ou destrui-la, representa então poder, que é protegido pelos legítimos detentores da mesma e que é buscado (de forma ilegítima) pelos invasores.

O acesso ao hardware e ao software (e o decorrente acesso à capacidade de processamento) também representa poder, já que a utilização dos mesmos permite o processamento de informações. O acesso ilegítimo à capacidade de processamento pode ser apenas roubo de tempo de processamento, mas esse tipo de ato pode levar a conseqüências sérias, como o aumento do custo para usuários legítimos ou, em caso extremo, a negação de serviço para usuários legítimos, uma vez que a CPU e/ou a memória estão ocupadas

realizando tarefas estranhas à instalação. Outras práticas ilegítimas seriam o acesso a serviços não disponíveis na instalação de origem, o acesso não autorizado a software, uso de capacidade de armazenamento, ou ainda para acesso a outras redes, usando o sistema invadido como dissimulação da origem da agressão. Um exemplo típico de roubo de tempo de CPU é a utilização de máquinas para decifrar informações criptografadas (como arquivos de senhas) para ter acesso a novas informações: uma agressão (roubo de ciclos) que alimentará outra agressão (a invasão de outros sistemas).

As agressões referentes à informação e à capacidade de processamento podem ser perpetradas basicamente de três formas: por escuta ou monitoração da rede; por invasão ao sistema; e por mascaramento.

A **escuta/monitoração** do canal é tarefa simples e mesmo com recursos pouco sofisticados é possível alcançar tal feito. Exemplos de formas de se conseguir monitoração de redes vão desde o uso de analisadores de protocolos (recurso caro) até a modificação do software de um computador comum para atuar como escuta. Existem ainda equipamentos próprios para escuta (passiva) que se valem de emanações eletromagnéticas dos cabos e conectores, dispensando inclusive o corte de cabos, bastando que os sensores estejam em contato com a camada externa do cabo ou de conectores. Cabos óticos representam alternativa para proteção contra este tipo de agressão, pois tornam impossível a interceptação sem o rompimento do cabo e instalação de equipamento adequado, o que torna mais fácil a detecção. O uso de micro-computadores com placas de rede em modo "promíscuo" (que recebe todos os pacotes mesmo que não sejam endereçados a ele) e a alteração de software de rede para não descartar os pacotes que são endereçados a outros nós da rede, são tarefas relativamente simples e eficientes para aquisição de informações importantes e sensíveis (como senhas). O comprometimento da segurança de um nó intermediário em redes *store-and-forward* ou de *gateways* e roteadores também pode levar à exposição de informações a terceiros. Então, toda informação que circula por redes pode ser interceptada e, se medidas de segurança não tiverem sido adotadas, esta informação se torna não-confidencial. Pouco se pode fazer a nível de software para impedir este tipo de agressão. O uso de criptografia deve ser considerado pois, apesar de não impedir o ataque, é uma forma de reforçar o sigilo das comunicações.

A **invasão** de sistemas com o objetivo de ganhar acesso a informações e a recursos computacionais é uma das agressões mais comuns e necessita de pouco aparato além de bom conhecimento e muita persistência por parte do perpetrante da ação. As vulnerabilidades relativas à invasão de sistemas podem ser geradas de muitas formas; por exemplo, pela não instalação de senhas por parte de usuários ou por falhas de implementação de softwares. Os riscos associados são os mesmos relacionados para a escuta do canal e mais a possibilidade de negação de serviços para usuários legítimos em função de invasores estarem usufruindo de serviços de forma não autorizada. A negação de serviço pode aparecer de várias formas: CPU com alta carga, espaço de armazenamento não disponível, serviços (como canais de comunicação) sobrecarregados, software eliminado ou afetado por vírus, equipamentos desabilitados, etc. O simples acesso não autorizado poderá caracterizar desde invasão de privacidade a roubo de informações. Mas, grande parte desta vulnerabilidade é de responsabilidade do sistema operacional hospedeiro, que deve oferecer a primeira barreira para evitar acesso não autorizado, tanto ao sistema como um todo quanto aos subsistemas que o compõem (como, por exemplo, o sistema de arquivos). Isto reduz a responsabilidade dos mecanismos de segurança das redes neste tipo de agressão, porque esta forma de agressão normalmente se transforma ou em escuta ou em mascaramento após concretizada a invasão, e em conseqüência devem ser fornecidos mecanismos de defesa contra agressões incorporados às redes.

A terceira forma, o **mascaramento**, consiste na tentativa de personificação de uma terceira entidade em uma comunicação. O objetivo é o mesmo que os anteriores: acesso a informações ou a recursos computacionais. Se é conhecido que uma entidade (usuário, software, sistema, processo, etc) possui acesso a determinada informação, uma forma de consegui-la é se fazer passar (personificar) esta entidade e solicitar a informação contando com a permissão de acesso "roubada". O mascaramento pode ser conseguido por invasão simples (como visto acima) ou por meios muito mais sofisticados como a alteração de pacotes que fluem na rede ou, ainda, forjando pacotes. Esta última forma exige uma combinação de agressões: escuta/monitoração do meio, seleção dos pacotes que interessam (ou seja, aqueles do remetente que serão incorporados e/ou que estejam fazendo acesso às informações desejadas), alteração dos mesmos de modo a não descaracterizar o remetente

mas alterando o nó de origem (para permitir o recebimento das informações solicitadas). Esta técnica é conhecida como *replay*. Também podem ser forjados pacotes identificando o remetente como sendo de um usuário legítimo, criando a idéia que o mesmo deseja acessar as informações solicitadas. Fazendo-se passar por uma entidade comunicante legítima da rede, o software "clandestino" pode ter acesso a informações sensíveis ou a recursos importantes e até provocar eventos anonimamente. Fica claro que este tipo de agressão é complexa, o que pressupõe a necessidade de o autor da agressão ser conhecedor profundo dos protocolos de comunicação utilizados e sugere um alto interesse pelas vantagens advindas da invasão. Disto conclui-se que, quanto mais sofisticada é a agressão, mais sofisticados têm que ser os mecanismos de defesa.

A figura 2.5 apresenta um modelo de mundo onde as três formas de agressão podem ser vislumbradas.

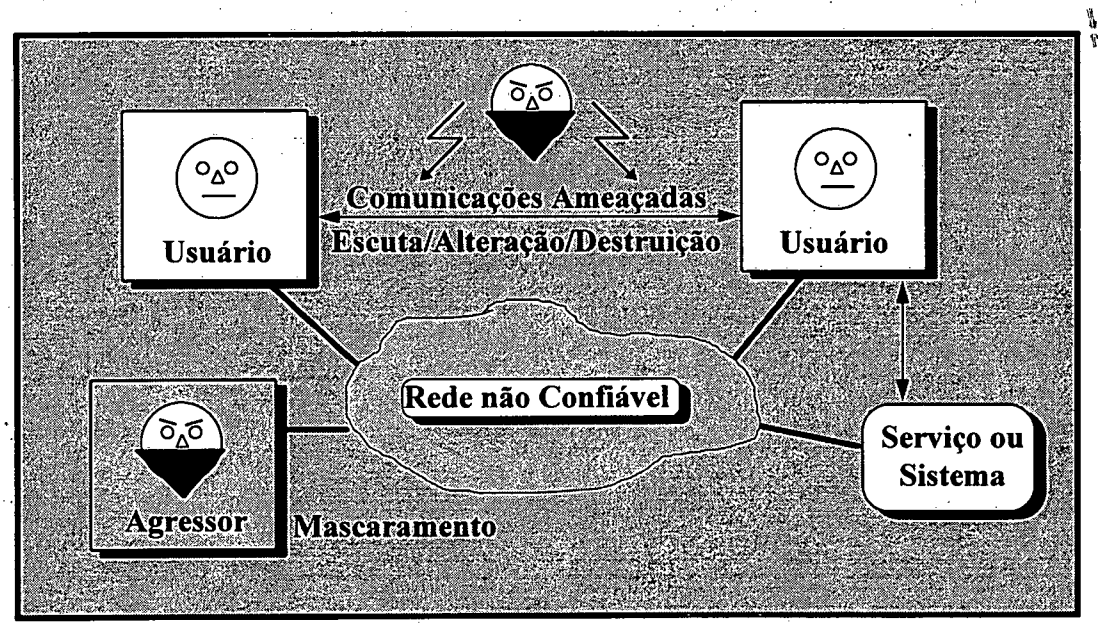

FIGURA 2.5: Ameaças em uma Rede de Computadores [KAR 91]

2.6. Outras Ameaças

Além das apresentadas, existem outras formas de se conseguir auferir benefícios através de sistemas em rede, que são vulnerabilidades dos mesmos. Um problema é o acesso externo por linha discada. Esta ameaça é caracterizada pela impossibilidade (se

não forem tomadas medidas de segurança) de se identificar a origem da ligação. Medidas de proteção são sugeridas como o uso de softwares cadastrados que contenham alguma espécie de identificação (números de série, p.ex.) que "garantiriam" que a origem da chamada é um *host* conhecido; ou então o uso de modems *call-back* que, uma vez ativados, abortariam a conexão e em seguida gerariam uma chamada para o usuário, desde que a identificação do mesmo (fornecida na primeira chamada) fosse correta.

Uma outra abordagem para vencer os mecanismos de proteção sugere que, se não é possível sobrepujá-los usando recursos como mascaramento, escutas, etc; deve-se tentar destrui-los (atentando contra os próprios mecanismos de proteção). A idéia básica é que se os mecanismos estiverem debilitados (ou mesmo forem removidos) torna-se mais fácil ultrapassá-los. Este problema é conhecido como segurança ou confiabilidade da gerência de segurança, que será analisado com mais profundidade adiante (capítulo 3).

2.7. Outras Medidas de Proteção

Como foi apresentado, além das medidas de proteção tradicionais como senhas de acesso e proteção de leitura e escrita de informações sensíveis, os sistemas em rede exigem outras medidas que não apenas procurem impedir o acesso, mas também permitam a autenticação dos usuários, garantir a integridade e a confidencialidade de informações. Além de mecanismos de software/hardware, outra forma de reforçar a segurança que não pode ser desprezada é a educação e conscientização dos usuários da importância que a segurança representa. É importante definir políticas, responsabilidades e atitudes quanto à segurança do sistema.

Outra atividade importante para a segurança é a de análise regular dos acontecimentos, através da auditoria dos registros que refletem as atividades desenvolvidas. A capacidade de monitorar o quão bem está a segurança pode ser conseguida através de análise dos registros de *logins*, das tentativas de *logins*, de problemas relacionados com pacotes (replicação, perdas, taxas de erros, etc), da evolução do uso de recursos e serviços (tráfego da rede, armazenamento, carga de CPU, etc), dentre outros.

A possibilidade de rastrear conexões e identificar usuários suspeitos é muito útil mas deve ser realizada com cuidados quanto a privacidade dos usuários, que não deve ser comprometida em função das necessidades de segurança. Outro fator a observar é o custo e o incômodo para usuários legítimos que medidas de segurança geram para serem efetivadas.

2.8. Serviços de Segurança

Os mecanismos de segurança serão os responsáveis pela execução da política de segurança implantada nos sistemas, e tradicionalmente se apresentam como serviços independentes, que interagem e interoperam para garantir a segurança. Lastreados em todo o embasamento apresentado, pode-se agora apresentar os serviços tradicionais de segurança.

- **Serviço de Confidencialidade**

É o responsável por manter informações secretas, sempre que necessário ou requisitado. Tal serviço se vale de criptografia e sistemas de chaves para atingir seu objetivo. Exemplo de aplicação: transmissão de arquivo com informações confidenciais.

- **Serviço de Integridade**

Permite que se detecte a alteração de informações realizadas sem a devida autorização. Para tanto, se emprega um sistema de *digest* (resumo) criptografado na origem e verificado no destino. O cálculo do *digest* leva em conta toda a informação sendo transportada, ficando, assim, sensível a alterações nos dados. Exemplo de aplicação: mensagens com informações públicas mas com requisito de precisão.

- **Serviço de Autenticação**

Valida o interlocutor, ou seja, permite que se prove que o remetente de uma mensagem ou um usuário/processo é quem diz ser, para evitar, por exemplo, o fornecimento de um serviço para um falso usuário. O uso de assinatura eletrônica, conseguida através de senhas e de criptografia, permite tal validação.

- **Serviço de Controle de Acesso**

Restringe o acesso aos recursos do sistema àqueles usuários autorizados, através de senhas e/ou listas de permissão de acesso. Ex.: leitura, escrita ou execução de um recurso por parte do proprietário do mesmo.

- **Serviço de Não-Repudiação**

É um serviço que visa impedir que um usuário (ou mesmo um processo) negue que enviou ou recebeu um objeto pela rede, apesar de isto ter efetivamente ocorrido. Para tanto, quando um destes eventos ocorre, o sistema providencia uma assinatura criptografada do remetente ou destinatário (conforme o caso) e a entrega a seu par na comunicação. Freqüentemente, este serviço é implementado com o apoio de uma terceira entidade, por vezes conhecida por *juiz*, parte neutra no processo.

- **Serviço de Confiabilidade (ou auto-confiabilidade)**

Cada mecanismo de segurança deve ser confiável, ou seja, deve ser robusto o suficiente para suportar e "sobreviver" a ataques hostis que possam impedi-lo de realizar suas funções. Ex.: o serviço de confidencialidade não pode revelar a senha de um arquivo criptografado transmitido.

- **Serviço de Auditoria**

Mantém registro sobre usuários e processos que acessaram (ou tentaram acessar) os recursos do sistema, para um possível rastreamento e análise posterior das informações coletadas, bem como para demonstrar garantidamente, mais tarde, que uma operação foi realizada. Ex.: tentativas sucessivas de suposição de senha.

Todos os serviços apresentados têm funções específicas objetivando, no conjunto, a instalação de barreiras para restrição seletiva do acesso às informações manipuladas por um sistema. Para que tais barreiras sejam seletivas e não definitivas (oferecendo, para os usuários autorizados, todos os serviços disponíveis na rede) deve-se implementá-las segundo um padrão.

2.9. Segurança no RM-OSI

A ISO definiu, em parceria com o ITU (antigo CCITT), um modelo de referência para interconexão de sistemas abertos, conhecido como RM-OSI. Um esquema básico de gerenciamento destes sistemas foi adicionado ao modelo, e é conhecido como Arquitetura de Gerenciamento de Redes [BRI 93].

2.9.1. O Modelo de Referência OSI

O RM-OSI (Modelo de Referência para Interconexão de Sistemas Abertos) tem como principal objetivo permitir a interconexão de sistemas de computadores heterogêneos a fim de que os processos de aplicação possam se comunicar entre si. O Gerenciamento de Redes faz parte deste modelo.

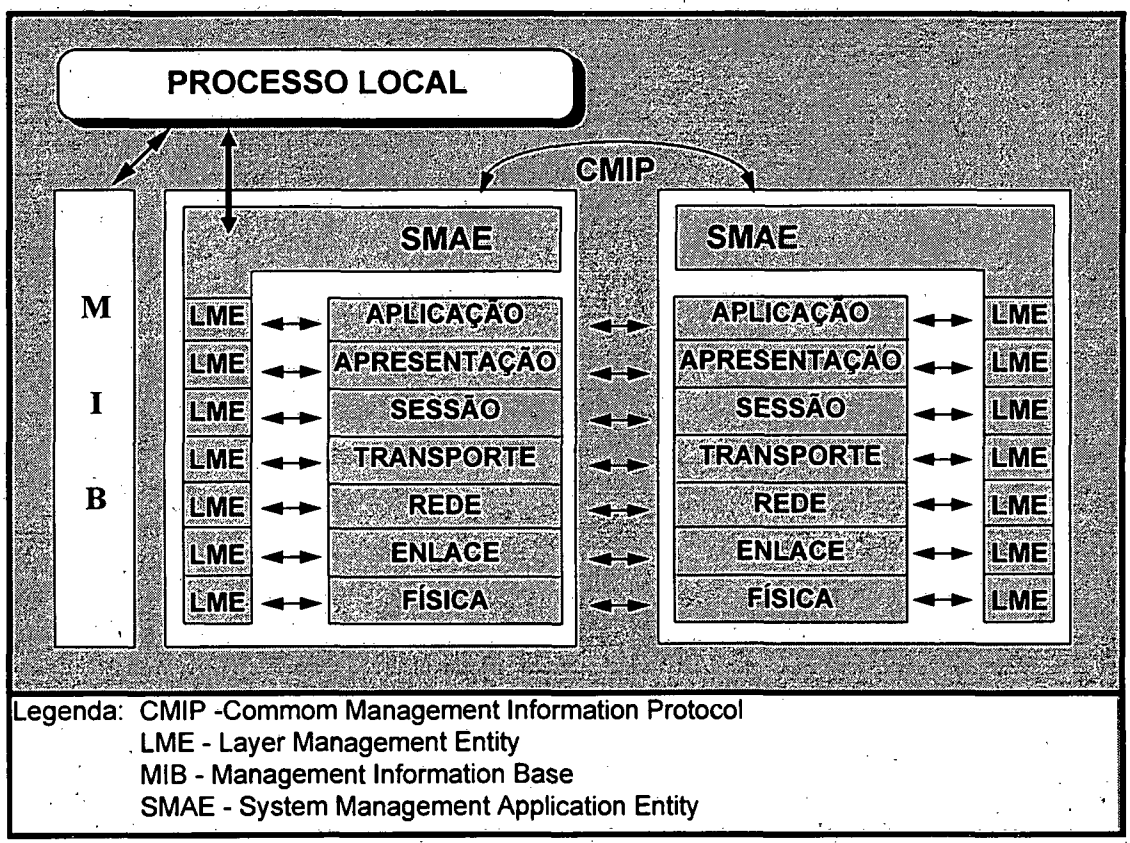

FIGURA 2.6: Modelo de Gerenciamento OSI [BRI 93]

A arquitetura do modelo OSI é composta por camadas, em número de sete, onde cada uma possui um conjunto bem definido de funções, conhecidos como serviços. O

Modelo proporciona interfaces entre as sete camadas, onde uma camada se utiliza dos serviços prestados pela camada imediatamente inferior, para realizar suas tarefas. Além disso, a Arquitetura de Gerenciamento acrescenta outra interface, via uma base de informações de gerenciamento (MIB), o que traz condições para que o gerenciamento seja realizado em todas as camadas [ISO 89] [BRI 93]. A figura 2.6 apresenta um esquema lógico do Modelo de Gerenciamento OSI.

2.9.2. Arquitetura de Gerenciamento OSI

O Gerenciamento OSI é realizado através de um conjunto de processos que podem estar distribuídos entre vários sistemas conectados. Em suma, é o responsável por manter informações fundamentais para o perfeito funcionamento da rede gerenciada. Abrange cinco áreas funcionais:

- Gerência de Configuração
- Gerência de Desempenho
- Gerência de Falhas
- Gerência de Contabilização
- Gerência de Segurança

Mais detalhes sobre Gerência OSI e cada uma das áreas funcionais, são apresentados no capítulo 3. A seguir será destacada a Gerência de Segurança, que se aproxima dos objetivos deste trabalho.

2.9.3. Gerência de Segurança

A área funcional de Segurança deve abordar os aspectos de segurança essenciais para operar uma rede OSI e proteger os objetos gerenciados e se baseia em duas estruturas:

- Arquitetura de Segurança
- Funções de Gerenciamento de Segurança

A Arquitetura especifica os serviços e mecanismos que devem estar presentes e a distribuição deles nas diversas camadas do modelo OSI. O padrão ISO para Segurança em Redes de Computadores abrange os 5 primeiros serviços apresentados na seção 2.8, distribuindo-os entre as camadas OSI. O padrão ISO 7498-2 especifica a Security Architecture [ISO 89], onde estão definidos a terminologia, os serviços e mecanismos de segurança e é feita uma introdução sobre gerência de segurança.

As Funções de Gerenciamento têm o objetivo de fornecer relatórios de eventos e estatísticos, manter registros históricos e controlar os próprios serviços de segurança. Estas funções estão descritas pela ISO em [ISO 90a], [ISO 90b] e [ISO 90c].

Os serviços de segurança definidos em [ISO 89] são:

1. **Autenticação**: deve verificar a identidade das entidades pares da comunicação e a origem das mensagens.
2. **Controle de Acesso**: protege contra a utilização não autorizada de recursos que estejam no sistema.
3. **Confidencialidade**: protege contra a revelação não autorizada de informação em uma comunicação (seja com ou sem conexão, sobre toda a mensagem ou campos selecionados e ainda o fluxo de dados).
4. **Integridade dos dados**: deve proteger contra inserção, modificação ou destruição não-autorizada de mensagens (em transmissões com ou sem conexão, toda a mensagem ou campos selecionados e com ou sem recuperação).
5. **Não-Repudiação**: protege contra tentativa do originador negar o envio ou do destinatário negar o recebimento de mensagem.

Os diversos níveis do RM-OSI possuem atribuições de segurança distintos, apresentados resumidamente a seguir:

- Camadas Física e de Enlace: a segurança consiste na confidencialidade e está baseada no que é chamado *link encryption*, uma forma de modulação especial do sinal, combinado *a priori* entre os interlocutores. Outra possibilidade é o tráfego de desnorteamento, também chamado de *full period*, onde tráfego cifrado e sem sentido é incluído permanentemente no canal, quando não há tráfego real, com o objetivo de dissimular os momentos em que há mensagens verdadeiras na linha.

- Camada de Rede: os principais serviços são os de confidencialidade e de integridade. Há um problema com a confidencialidade que é a impossibilidade de criptografar os campos de controle (cabeçalhos, ...), uma vez que a mensagem irá percorrer um caminho desconhecido e passará por vários IMPs (*Interface Message Processors)* antes de chegar ao destino, sendo então necessário para os IMPs conhecer o destino para fazer o roteamento (camada 3). Se os cabeçalhos estiverem cifrados, todos os IMPs devem conhecer a chave, deixando de ser confidencial a mensagem, desta forma, não se pode criptografar os cabeçalhos do nível de rede. Para a integridade, utiliza-se a técnica do *digest* criptografado e de outros campos sensíveis como o de seqüência dos pacotes.

- Camada de Transporte: deve haver negociação, quando da conexão, sobre os mecanismos de segurança. Os mecanismos pretendem, neste nível, garantir principalmente a integridade e autenticação fim-a-fim, utilizando-se criptografia dos campos de seqüenciação, *checksum* e outros.

- Camada de Sessão: não há serviço de segurança ISO definido para este nível, apesar de reconhecidamente o controle de sessões ser básico para o controle de acesso à rede. (DECnet oferece vários mecanismos neste nível, bem como SNA).

- Camada de Apresentação: pela definição primeira deste nível por parte da ISO, este seria o ponto para inclusão de serviços de tradução de caracteres, compressão e criptografia de dados. Ou seja, casa natural de serviços de segurança; mas apenas estão previstos serviços

de confidencialidade, integridade, autenticação e não-repudiação, praticamente todos baseados em criptografia e variações.

- Camada de Aplicação: neste nível estão 5 serviços OSI: Autenticação, Controle de Acesso, Integridade, Confidencialidade e Não-Repudiação. Pela colocação das aplicações do usuário neste nível, a possibilidade de identificação do usuário e a interação com este, além de ser a melhor posição para determinar acuradamente o que precisa ser protegido e contra quais ameaças, é natural que aqui estejam presentes os principais mecanismos de segurança. Abaixo os serviços com mais detalhes:

    - Autenticação: pode ser utilizada pelos protocolos FTAM (*File Transfer, Access and Management)*, VT (*Virtual Terminal*), RPC (*Remote Procedure Call*) e MHS (*Message Handling System*), p. ex.; para certificarem-se de quem é o interlocutor. Pode ser utilizado quando da associação e/ou a cada APDU (*Application Protocol Data Unit*) trocada.

    - Controle de Acesso: vale-se de senhas e listas de permissão para uso de recursos ou dados sensíveis (com FTAM, VT e RPC, p. ex.).

    - Confidencialidade: permite a seleção dos campos a criptografar para garantir o sigilo não apenas do conteúdo da mensagem mas até do destinatário ou remetente.

    - Integridade: criptografia de campos selecionados e/ou acrescentados (*checksum*, p.ex.) para permitir posterior verificação de integridade da mensagem.

    - Não-Repudiação: este mecanismo é obtido pelo uso de assinaturas digitais (criptografadas) agregadas às APDUs e notarização.

2.9.4. Ferramentas de Apoio à Segurança

Abaixo são descritas, de modo simplificado, algumas ferramentas disponíveis para implementação dos serviços de segurança citados acima.

Pode-se perceber claramente que um dos focos principais dos mecanismos de segurança, em todos os níveis, é a criptografia. A criptografia é utilizada desde os mais remotos tempos como salvaguarda ao elo mais fraco na corrente do sigilo: a integridade das pessoas - transportadoras ou depositárias das mensagens. Agora ela é usada também para evitar problemas que podem vir a ser causados por pessoas com interesses escusos. Técnicas de criptografia são usadas para salvaguardar informações enquanto elas estão armazenadas em um nó da rede ou enquanto elas transitam entre nós, via rede. O uso contra ataques aos nós da rede é complementar a outros, como o controle de acesso lógico e físico. Já no caso de prevenção contra ataques sobre o meio de comunicação, a criptografia é de suma importância uma vez que as oportunidades de interceptação são muitas.

O método mais tradicional na criptografia é o da Chave Secreta ou Chave Única, também chamada criptografia simétrica. Esta única chave serve tanto para cifrar quanto para decifrar uma mensagem (ou um arquivo, ou o que for o objeto da cifragem). Se o uso deste tipo de chave não estiver relacionado com transmissão de dados então é uma boa escolha. Mas esta abordagem leva ao problema de que ambos os interlocutores (no caso de uma comunicação confidencial) devem conhecer a chave. Para que haja o acerto sobre a chave, é necessária uma negociação *off-line* ou por outro caminho que não a rede por onde se dará a transmissão da mensagem, pois se a chave for enviada pela mesma rede, a segurança estará quebrada, uma vez que a linha poderá estar sob escuta. O algoritmo chamado DES (*Data Encryption Standard*) implementa a criptografia simétrica e é hoje muito utilizado [PFL 89] [GAR 92].

Outra abordagem é a da Chave Pública, também conhecida como Chave Dupla. Esta chave na verdade é um par de chaves onde uma é pública e a outra privada. A chave pública deve ser amplamente divulgada e poderá/deverá inclusive estar à disposição em um servidor de chaves. A chave privada é de propriedade do usuário e deve ser mantida em segredo. O truque está no algoritmo de criptografia que permite a cifragem com a chave pública e somente a chave privada poderá decifrá-la, garantindo assim que uma mensagem endereçada a um usuário e criptografada com a chave pública deste usuário somente será compreendida por ele. O inverso também é verdadeiro, ou seja, uma mensagem cifrada com

a chave privada somente será decifrada com a chave pública correspondente. Tal característica pode ser utilizada para diversos mecanismos como autenticação da origem e integridade da mensagem. Vale lembrar que a chave privada não pode facilmente ser deduzida a partir da pública e vice-versa. O algoritmo RSA (*Rivest, Shamir e Adleman*) é um exemplo de algoritmo assimétrico [NEC 90].

Não apenas a salvaguarda de informações pode ser conseguida com a criptografia, mas a verificação da autenticidade (ou seja, a autenticação e integridade) de uma mensagem também pode ser conseguida utilizando métodos de criptografia. Uma forma convencional de implementar este conceito é o uso de um campo de *checksum* (qualquer forma de verificação, como CRC - *Cyclic Redundancy Check*, por exemplo) que freqüentemente está presente nas mensagens que trafegam nas redes ou pode ser incluído para este fim específico. Se uma mensagem for alterada, o campo de *checksum* deve ser recalculado para representar fielmente a nova mensagem. Se este campo for criptografado, não há como uma alteração indevida da mensagem passar despercebida por parte do receptor da mesma. Isto porque seria necessário também alterar o *checksum* e o interceptador não conhece a chave utilizada e, mesmo que recalcule corretamente o *checksum*, o mesmo não tem como cifrar o campo corretamente, permitindo que o receptor real da mensagem perceba que a mensagem não está íntegra. Da mesma forma, também estará garantida a origem da mensagem, pois, se o agressor alterar o campo que indica o remetente da mensagem, o destinatário real considerará a mensagem não-autêntica e/ou não-íntegra por discordância no *checksum*. Técnicas mais robustas se valem de algoritmos de *hash* que praticamente impedem que mais de uma mensagem gere o mesmo resultado e que são matematicamente não-inversíveis (técnicas chamadas de *digest algorithms*), evitando assim a possibilidade (ainda que remota) de composição de uma mensagem que gerasse o mesmo *checksum*.

O uso da criptografia permite ainda a autenticação das entidades pares. Isto pode ser conseguido pela cifragem, por parte dos interlocutores, de um campo definido da mensagem com conteúdo previamente determinado, com a própria chave privada. O destinatário pode então ter certeza do remetente, pelo uso da chave pública do remetente para decifrar o campo e verificar se o conteúdo é o esperado (esta técnica é conhecida como

digital fingerprint, e é uma variante da técnica de assinatura digital). Uma apresentação mais técnica dos métodos de criptografia DES e RSA e de um método de *digest* (MD5) é feita no Anexo A.

Outras ferramentas exigem a "presença" de uma terceira parte, neutra e confiável pelas entidades comunicantes. É o caso de um mecanismo de notarização, que servirá de guardião de informações e fará papel de juiz para resolver pendências relacionadas com repudiação, por exemplo.

Em relação ao tráfego na rede, podem ser necessárias técnicas de controle de roteamento e de tráfego de desnorteamento que devem confundir o adversário que tenta deduzir informações a partir da análise do tráfego ou mesmo evitar *links* reconhecidamente inseguros.

2.10. Conclusão

Os problemas relacionados à Segurança em Redes de Computadores são muitos e sérios. A diversidade com que se apresentam são o primeiro desafio para o estabelecimento de políticas e a criação de mecanismos de defesa.

As vulnerabilidades, as ameaças e os riscos devem ser minuciosamente levantados e medidas de prevenção adotadas de acordo com as necessidades dos usuários e das instalações. A plataforma tradicional de segurança identifica os serviços básicos de segurança, que devem interoperar de modo harmônico, para conseguir uma plena estabilidade e confiabilidade da rede para os usuários.

3 Segurança da Gerência de Redes

Um Sistema de Gerência de Redes e as informações por ele geradas são de grande importância porque estão relacionadas com todas as funções e operações sobre os recursos disponíveis através da rede. Problemas de segurança associados ao Gerenciamento podem, então, ser muito prejudiciais à rede e aos usuários. Neste capítulo são apresentadas as vulnerabilidades e ameaças às quais estão expostos os Sistemas de Gerência e os requisitos para garantir a segurança em tais sistemas.

3.1. Gerência de Redes

Os sistemas de gerência existentes hoje em dia aderem com mais ou menos força ao padrão definido pelo RM-OSI, em [ISO 89]. Em função disto, será apresentada a seguir uma visão de Gerenciamento OSI, o que fornecerá uma visão ampla dos Sistemas de Gerência.

Gerência de Redes é uma aplicação distribuída onde processos de gerência (agentes e gerentes) trocam informações com o objetivo de monitorar e controlar a rede. Os processos de gerência se relacionam com objetos gerenciáveis, conforme suas atribuições:

- Gerente: obtém informações a partir dos agentes sobre os objetos gerenciados e os controla, emitindo operações de gerenciamento[1] para os agentes.
- Agente: executa operações de gerenciamento sobre objetos gerenciados e transmite as notificações[2] emitidas pelos objetos gerenciados ao gerente.
- Objeto Gerenciável (ou gerenciado): representação do recurso do sistema que está sujeito ao gerenciamento. As informações referentes aos recursos estão em uma base de informação de gerenciamento (MIB), acessível somente pelo agente.

---

[1] *Primitivas enviadas aos objetos gerenciados ou agentes para o monitoramento e controle da rede.*

[2] *Informações de gerenciamento emitidas pelos objetos gerenciados ou agentes em resposta à ocorrência de algum evento.*

A dinâmica dos processos de gerência pode ser assim descrita: o processo gerente envia solicitações ao processo agente que por sua vez responde às solicitações e também transmite notificações referentes aos objetos gerenciados que residem na MIB [WES 92] [BRI 93]. Esta comunicação se dá através de Elementos de Serviço que compõem a Camada de Aplicação:

- CMISE: Elemento de Serviço de Informação de Gerenciamento Comum, que, utilizando o protocolo CMIP, provê um meio básico de troca de informações para as operações de gerenciamento.

- ROSE: Elemento de Serviço de Operação Remota, que é elemento básico da Camada de Aplicação, permite interações entre aplicações remotas com resposta a cada operação.

- ACSE: Elemento de Serviço de Controle de Associação, permite o estabelecimento de associações entre gerentes e agentes.

Como citado em 2.9.2, a Gerência de Redes OSI atua em cinco áreas funcionais de gerenciamento, apresentadas na figura 3.1:

- Gerência de Configuração: exerce controles sobre a configuração física e lógica da rede.

- Gerência de Desempenho: analisa e controla o desempenho e as taxas de erros da rede, incluindo informações históricas sobre o funcionamento da rede através de *logs*. Visa garantir o bom desempenho do sistema através da análise de sua utilização, fornecendo, para a administração da rede, informações sobre gargalos e desperdício dos recursos disponíveis.

- Gerência de Falhas: responsável pela detecção, isolamento e controle de procedimentos anormais da rede. Quando possível, deve permitir à administração se antecipar às falhas para corrigir os problemas antes de as falhas ocorrerem.

- Gerência de Contabilização: faz a coleta e o processamento dos dados referentes ao consumo de recursos na rede, permitindo um controle mais adequando quanto aos custos da rede.

- Gerência de Segurança: controla e monitora os mecanismos de segurança dos processos da rede, abrangendo controle de acesso, integridade, autenticação de mensagens e de entidades, confidencialidade, etc.

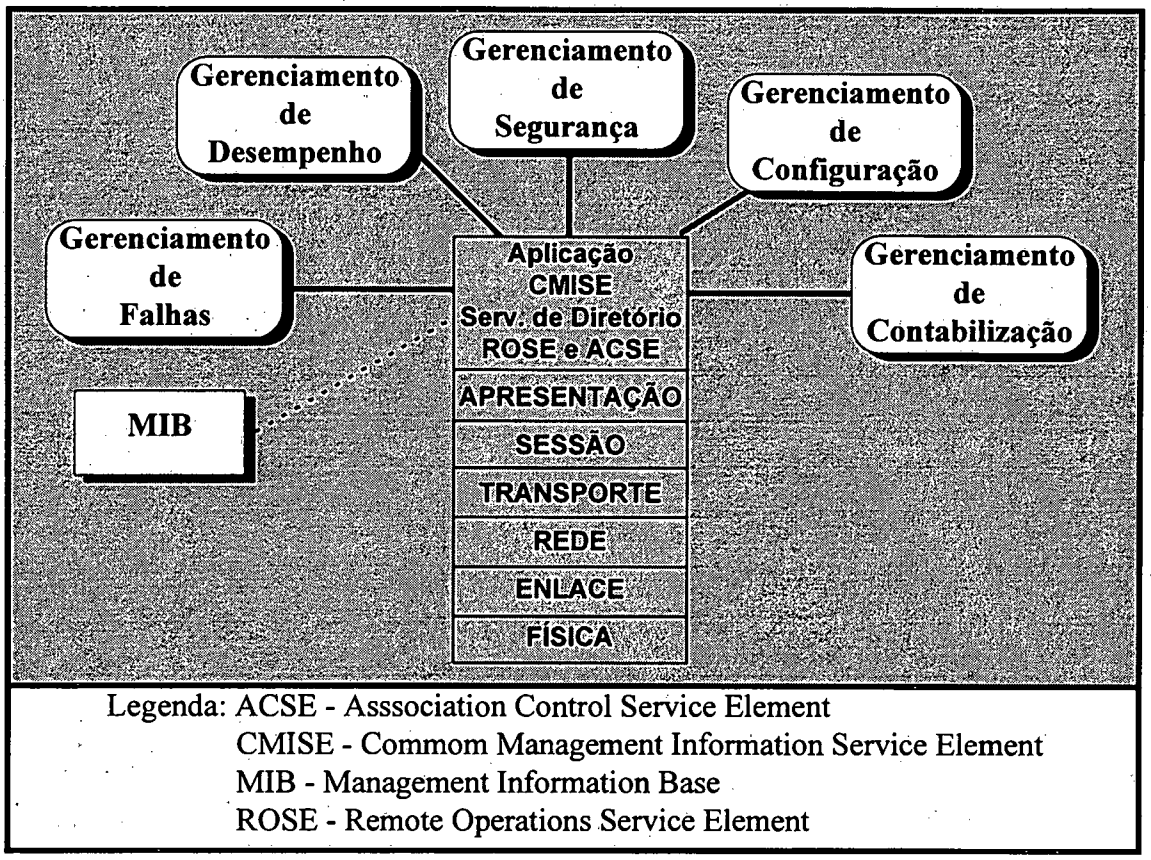

FIGURA 3.1: Áreas Funcionais de Gerência [BRI 93]

3.2. Vulnerabilidades dos Sistemas de Gerência de Redes

Toda e qualquer informação relacionada ao Sistema de Gerência, em um determinado instante, ou está em uma MIB, sendo tratada pelos processo de gerência, ou está trafegando pela rede (em uma comunicação típica entre um agente e um gerente ou entre dois gerentes) ou ainda poderá ser deduzida (reproduzida) com informações parciais oriundas destas duas fontes. Toda informação associada ao Sistema de Gerência é útil para a manutenção da rede em operação com confiabilidade. Sem dúvida, os Sistemas de Gerência facilitam a administração das redes, seja pela automatização de algumas atividades, por permitir maior controle sobre os recursos da rede, ou ainda, por fornecer informações

(estatísticas, p.ex.) que permitirão ajustes, correções ou adaptações às necessidades dos usuários.

Entretanto, neste ponto também é possível observar que o próprio Sistema de Gerência e as informações por ele geradas são de extrema valia para indicar pontos vulneráveis a ataques, ter acesso e controlar indevidamente recursos da rede, manipular informações, em suma, realizar atividades prejudiciais à rede, aos sistemas e/ou aos usuários.

Sob certa ótica, é possível até afirmar que uma rede com um Sistema de Gerência implantado é menos segura do que a mesma rede sem o Sistema de Gerência. Isto porque, quando há gerência, há mais vulnerabilidades e há mais pontos onde é possível desferir um ataque. Assim sendo, existem mais possibilidades de comprometimento da rede e de seus sistemas. Aqui estão alguns exemplos:

- Se um agente emite um alarme acerca de falha em um mecanismo e este alarme é interceptado por um invasor; então está-se fornecendo uma informação valiosa para que um intruso possa realizar outras agressões.

- Uma notificação de alarme forjada por um intruso pode levar a alguma ação (por parte do gerente "iludido") que libere informações ou serviços a usuários que não teriam autorização em situações normais.

- Uma entidade infiltrada que se faz passar por gerente pode ter acesso a informações sensíveis mantidas na MIB, inclusive com poder de alteração (como desativação de serviços de segurança ou alteração de registros de contabilização).

- Um agente mascarado pode fornecer acesso a recursos da rede para usuários não autorizados e/ou impedir o acesso a tais recursos para usuários legítimos; ou ainda forjar informações com o intuito de forçar o gerente a alocar mais ou melhores recursos.

Estas e muitas outras vulnerabilidades não existiriam se não existisse um Sistema de Gerência. Ressalve-se que também são possíveis ataques que resultariam nos

mesmos "benefícios", mas a Gerência acrescenta pontos passíveis destes ataques aos já existentes; e por isso pode-se concluir que, por um lado, um Sistema de Gerência de Redes torna a rede mais insegura, ao mesmo tempo que cria mecanismos de controle que serão úteis também na manutenção da segurança da rede.

3.3. Ameaças sobre Sistemas de Gerência

As ameaças que serão abordadas dizem respeito às agressões que podem ser perpetradas por intrusos na rede ou por usuários que tentam obter mais recursos ou informações do que são autorizados. Alguns exemplos destas agressões foram apresentadas anteriormente e que, de acordo com definições também já apresentadas, podem ser classificadas em:

- Mascaramento;
- Monitoração ou Escuta Passiva;
- Escuta Ativa.

Estas ameaças às quais estão expostos os Sistemas de Gerência de Redes são apresentadas com mais detalhes a seguir.

3.3.1. Mascaramento

Mascaramento é a pretensão de uma entidade de se fazer passar por outra de modo a ter acesso a informações, ganhar novos privilégios e afetar os sistemas. São evidentes as vantagens que uma entidade pode obter ao se fazer passar por outra. A primeira delas é o anonimato, tornando difícil a descoberta da origem da agressão. Outras vezes, a intenção é "incriminar" outro usuário por atos ilícitos. A figura 3.2 mostra duas entidades mascaradas interagindo com um gerente e um agente de modo a conseguir privilégios, acessar informações e outras vantagens.

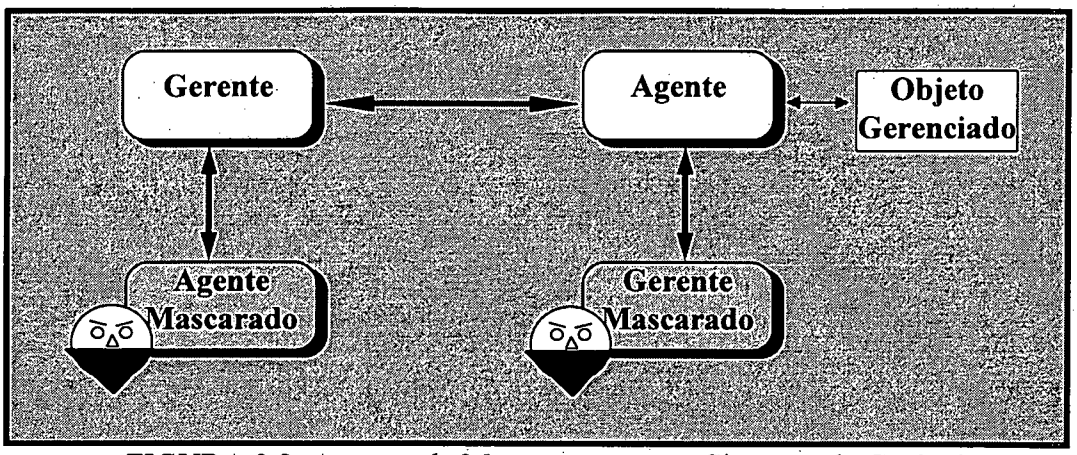

FIGURA 3.2: Ameaça de Mascaramento em Sistemas de Gerência

Para criar uma entidade mascarada, o agressor deve ter acesso à rede, mas pode ser um acesso autorizado (lícito) ou não. Então, uma primeira barreira contra este tipo de agressão é um Sistema de Controle de Acesso à rede o mais confiável possível. Controle de acesso envolve identificação, autenticação e autorização, além de uma política de segurança consistente e usuários conscientes da necessidade e importância da segurança para a rede e seus sistemas. Por **identificação** entende-se uma estrutura de nomes que garanta a identificação única para cada entidade da rede. Mas não basta identificação porque as entidades podem não ser confiáveis ao se identificar, sendo necessária então a confirmação da identidade, ou seja, a **autenticação**, que é a validação de que uma entidade é quem ou aquilo que diz ser. A **autorização** permite indicar se determinada entidade (identificada e autenticada) possui acesso legítimo (autorizado) a determinado recurso ou operação e deve evitar o acesso caso contrário.

Mas não se pode pensar em Controle de Acesso somente em um primeiro momento, como no caso do primeiro acesso em uma sessão (*login*) mas também em outras atividades durante a interação com os recursos do sistema, de forma continuada, sob pena de abordar o problema de maneira muito simplista.

No que diz respeito a entidades de Gerência, o controle de acesso é um ponto crucial, justamente pelos problemas que podem ser causados se um agressor conseguir forjar uma entidade e esta seja "aceita" como legítima pelas entidades pares comunicantes. Os

protocolos de gerência, por sua natureza, não são complexos e, portanto, a criação de entidades que simulam agentes ou gerentes não é tarefa rigorosamente difícil.

É então importante para a segurança em um Sistema de Gerência que cada entidade componente do mesmo esteja devidamente identificada e autenticada e tenha os direitos de acesso definidos e controlados. Para tanto, faz-se necessária a especificação e implantação de Serviços de Identificação/Autenticação específicos para entidades comunicantes, e de Confidencialidade de Acesso aos recursos (no caso, o acesso à MIB).

3.3.2. Escuta Passiva

Neste caso, há apenas coleta de informações que transitam na rede. Apesar de, em um primeiro momento, os riscos representados pela escuta passiva parecerem pequenos, é possível recolher muitas informações úteis para o comprometimento de uma rede. Um exemplo são as informações que dizem respeito à segurança da rede ou sobre falhas, que fluem entre agentes e gerentes da rede. Estas informações podem ser senhas de usuários, informações trocadas entre entidades para autenticação, informações sobre configuração e informações sobre falha de algum mecanismo de segurança.

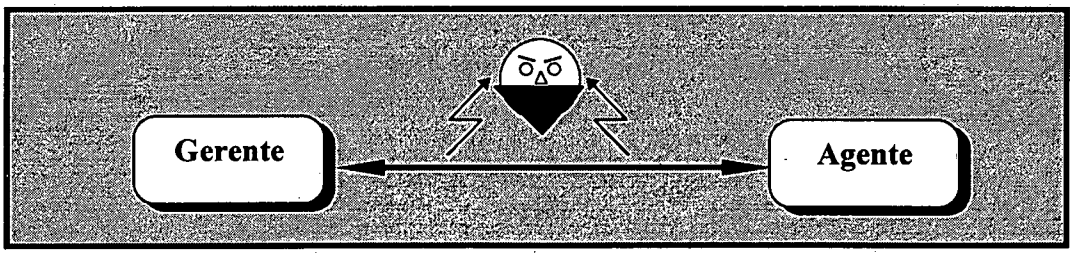

FIGURA 3.3: Ameaça de Escuta Passiva em Sistemas de Gerência

Quando informações sensíveis como as citadas devem transitar pela rede, é fundamental que sejam adotadas medidas para evitar que tais informações sejam acessadas indevidamente. Para tanto é necessário definir um Serviço de Confidencialidade de Comunicação, para evitar que agressores coletem informações nas comunicações entre entidades, como apresentado na figura 3.3.

É fácil perceber que grande parte das informações que são trocadas entre entidades de um Sistema de Gerência são sensíveis e portanto devem ser protegidas.

3.3.3. Escuta Ativa

A escuta ativa difere da escuta passiva por não apenas coletar informações que fluem pela rede, mas também por alterá-las de alguma forma, seja no conteúdo, na seqüência, no tempo ou pela destruição ou criação de mensagens; de forma a realizar ou induzir ações não autorizadas ou criar condições para ações não autorizadas ou ainda encobrir atos ilícitos praticados. A interferência é realizada sobre as comunicações, como mostra a figura 3.4.

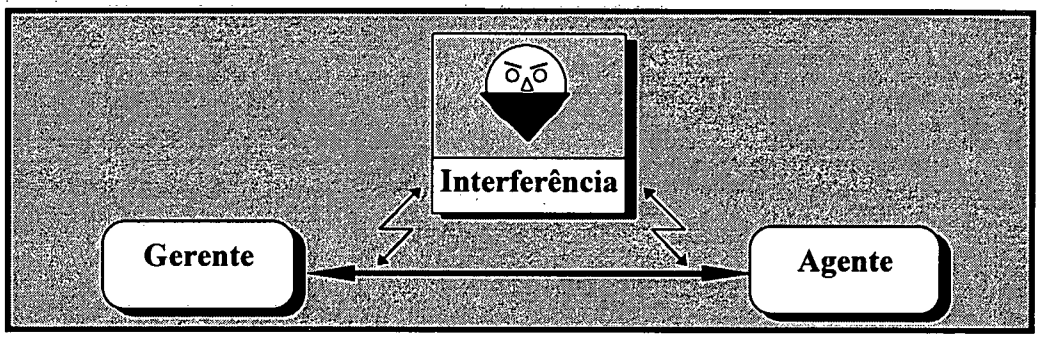

FIGURA 3.4: Ameaça de Escuta Ativa em Sistemas de Gerência

A barreira criada pelo controle de acesso citado acima não é efetiva contra esta ameaça uma vez que a atuação pode se dar sobre mensagens legítimas de entidades autenticadas e autorizadas. A autenticação das entidades comunicantes, como citado, visa garantir que não há entidades mascaradas mas não impede que sejam criadas mensagens espúrias em nome de uma entidade legítima. Outros casos (não abordados pelos serviços de autenticação e autorização) dizem respeito à destruição de mensagens, atrasos forçados em mensagens, reordenação de mensagens e repetição de mensagens em outro momento.

Duas medidas de proteção se tornam então necessárias: autenticação da origem das mensagens e garantia da integridade das mensagens. Sem estes dois serviços a rede continuará aberta a ataques.

Deve-se então acrescentar ao Serviço de Identificação/Autenticação de

entidades a tarefa de autenticar a origem, a forma e o momento do envio das mensagens. Além disso, um Serviço de Integridade de mensagens deve ser estabelecido e será o responsável pela garantia de que uma mensagem não sofreu alterações em seu caminho desde a origem ao destinatário, envolvendo as tarefas de evitar alteração de informações, re-seqüenciamento e a simples destruição.

A autenticação, por sua vez, também depende da integridade das mensagens para algumas tarefas. Por exemplo, se uma mensagem foi alterada indevidamente, de nada adianta validar sua a origem, pois a mensagem será descartada por falta de integridade. Então há uma interdependência entre os serviços e eles devem estar em atividade para que se possa dar confiabilidade à rede.

3.4. Conclusão

Neste capítulo foram apresentados os principais problemas de segurança relacionados aos sistemas de gerência de redes, e também os requisitos para garantir a segurança destes sistemas. As ameaças sobre um sistema de gerência são ainda mais prejudiciais do que sobre um ambiente de rede sem gerência pois as informações e ações de gerência são, por natureza, muito importantes e representam o *status* atual ou futuro da rede. Também as ações de gerência são fundamentais para o bom comportamento da instalação, e o comprometimento da gerência pode abrir todos os sistemas para a ação de invasores.

Os Sistemas de Gerência de Redes estão sujeitos a todas estas ameaças porque se valem de um paradigma de separação das funções de gerência, o que gera tráfego de informações entre as entidades de gerência. Torna-se fundamental a existência de um conjunto de serviços que atenda a demanda por segurança destes sistemas. No próximo capítulo, é descrito um Modelo de Arquitetura de Segurança que, aplicada à Gerência de Redes, permitirá um controle maior sobre os pontos vulneráveis e a implantação de obstáculos à ação de invasores.

4 Arquitetura de Segurança para Gerência de Redes

4.1. Definições Preliminares

Como apresentado no capítulo 3, a Gerência de Redes está baseada na comunicação entre entidades agentes e gerentes e em uma base de informações de gerência (MIB). Toda informação produzida e manipulada pelo Sistema de Gerência é útil para a manutenção da rede em operação com confiabilidade e com bom desempenho. Mas, da mesma forma que traz benefícios, abre vulnerabilidades muito perigosas para todo o sistema gerenciado, pois se o simples acesso às informações de gerência já pode significar grande vantagem para um agressor, maior é a possibilidade de, por meios escusos, gerar solicitações de alteração na configuração, manipulação de informações de contabilização e desempenho ou mesmo desativar serviços de segurança.

Ambientes de Gerência de Redes necessitam então de serviços de segurança para garantir a confiabilidade, uma vez que ter domínio sobre o gerenciamento da rede é o mesmo que deter o domínio sobre **toda** a rede, visto que estes ambientes possibilitam o controle de configuração, falhas, desempenho, contabilização e da própria segurança da rede.

Para este contexto, a seguir é apresentada a especificação de uma Arquitetura de Segurança para um Ambiente de Gerência de Redes genérico. Neste ambiente genérico interagem as seguintes entidades:

- **Agentes**: atuam diretamente sobre objetos que representam recursos da rede (objetos gerenciáveis), alterando seu estado, detectando falhas, acumulando informações ao longo do tempo e emitindo relatórios. Os agentes também interagem com os gerentes (ver definição abaixo) atendendo solicitações ou emitindo avisos. Mantém atualizada e consistente uma base de informações acerca dos objetos que gerencia (chamada MIB - *Management Information Base*) e o acesso a esta base só pode ser realizado pelo agente proprietário da mesma.

- **O gerente**: centraliza informações oriundas dos agentes e é responsável pela gerência de um conjunto de objetos gerenciáveis (via agentes), valendo-se de serviços de gerenciamento que são submetidos aos agentes. Eles interagem também com outros gerentes.

- **O objeto gerenciável**: representa um recurso da rede que é monitorado e controlado por um agente.

4.2. Requisitos de Proteção

No contexto de um Sistema de Gerência, as ameaças que cabem ser analisadas, dentre todas as ameaças à segurança em uma rede de computadores, são as seguintes:

a) acesso não autorizado à informação de gerência que flui pela rede;
b) acesso não autorizado à informação de gerência mantida na MIB;
c) alteração e re-seqüenciamento de mensagem de gerenciamento; e
d) geração de mensagens de gerenciamento por terceiros (entidades que não fazem parte do Sistema de Gerência).

Os Serviços de Segurança que precisam estar disponíveis para contrapor as ameaças, vistas acima, são:

1. Serviço de Confidencialidade (contra (a) e (b));
2. Serviço de Integridade (contra (c)); e
3. Serviço de Autenticação (contra (d)).

O **Serviço de Confidencialidade** (ou privacidade) é necessário para evitar o acesso não autorizado às informações de gerência, tanto as que estão armazenadas na MIB quanto as que fluem na rede entre agentes e gerentes. Para tal é necessário o emprego de criptografia, uma vez que é impossível evitar que uma mensagem que esteja trafegando na rede seja interceptada ou que as informações que estão na MIB (presume-se um arquivo) sejam acessadas, então deve-se impedir que as informações tenham significado se não se estiver de posse de uma chave para decifrá-las.

O **Serviço de Integridade** deve estar atento para evitar a alteração de mensagens ou o re-seqüenciamento (re-submissão de mensagem antiga). Mensagens originais alteradas ou re-submetidas representam uma séria ameaça de danos ao sistema, já que podem vir a ser consideradas mensagens válidas, pois foram montadas e expedidas originalmente por entidades autorizadas. Por isso, é necessária a inclusão de dois mecanismos de proteção: um que garanta que as mensagens não foram alteradas durante o trânsito pela rede (Integridade de conteúdo) e outro que verifique a seqüencialização das mensagens, evitando a inclusão de mensagens antigas (Integridade no tempo).

Por sua vez, o **Serviço de Autenticação** deve garantir que a origem das mensagens de gerenciamento são entidades legítimas, para evitar a execução de ações indevidas ou o acesso a informações por terceiros não autorizados. Este serviço deve providenciar meios para que somente as entidades que façam parte do Sistema de Gerência possam comunicar-se entre si.

4.3. Modelo Lógico da Arquitetura de Segurança

Uma Arquitetura de Segurança para aplicação em Sistemas de Gerência de Redes especifica os serviços e mecanismos de segurança que devem ser adotados e define como estes devem atuar sobre e/ou em conjunto com o Sistema de Gerência.

A Arquitetura de Segurança é composta por **Interfaces de Segurança** que devem acompanhar todas as entidades que compõem o Sistema de Gerência (gerentes e agentes), oferecendo serviços a estas entidades. Estas Interfaces comunicam-se diretamente com gerentes e agentes e utilizam-se dos serviços de suporte à comunicação oferecidos pela rede, para estabelecer os canais de comunicação entre as partes.

Cada **Interface de Segurança** deve garantir que as mensagens recebidas pelos agentes e gerentes são realmente internas ao Sistema (autênticas) e que não foram alteradas (íntegras). Além disso, pode ser desejável a confidencialidade da comunicação entre as entidades do Sistema e esta característica também deve ser garantida pela Interface de Segurança. Esta interface atuará como uma *clearing house* entre cada par comunicante,

impedindo que informações de gerência sejam acessadas, alteradas ou forjadas por entidades não autorizadas ou que estas informações falsas cheguem ao Sistema de Gerência.

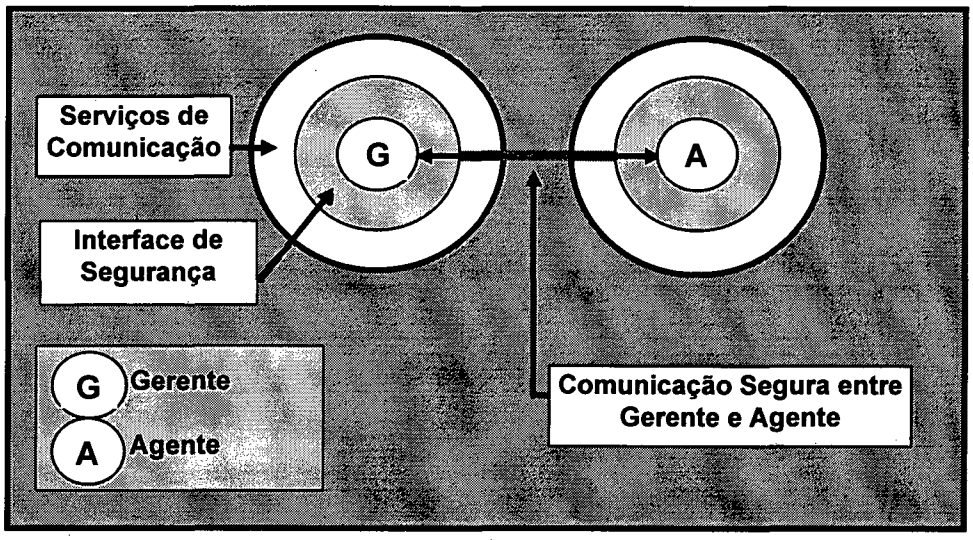

FIGURA 4.1a: Esquema Lógico da Arquitetura de Segurança

A Interface de Segurança é, portanto, uma redoma que encapsula totalmente cada agente e cada gerente do Sistema de Gerência, de forma que toda comunicação entre estas entidades se dê somente através da Interface, como visto nas figuras 4.1a e 4.1b. A figura 4.1a representa a Interface de Segurança em relação a entidades agentes e gerentes. A figura 4.1b apresenta como são os inter-relacionamentos entre uma entidade de um Sistema de Gerência e os serviços de comunicação que os suportam como disponíveis no ambiente de rede em questão, quando incluída a Interface de Segurança proposta.

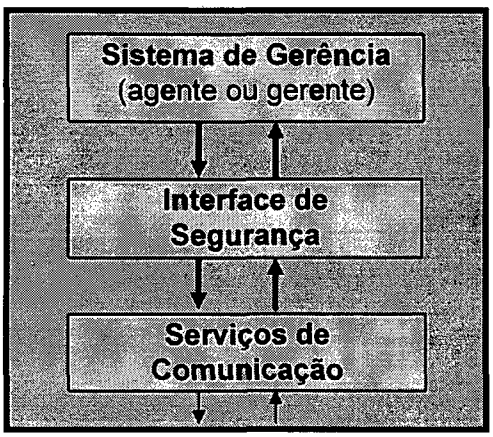

FIGURA 4.1b: Inter-relacionamentos da Interface de Segurança

Os serviços de rede devem suportar a comunicação fim-a-fim das Interfaces de Segurança, desta forma, o implementador deve, baseado no ambiente de rede subjascente, encontrar o ponto correto de inclusão da interface de segurança. Os serviços diretamente implementados pela Interface de Segurança são:

- **Serviço de Autenticação**, garantindo a origem autêntica das mensagens;
- **Serviço de Integridade**, que impede o processamento de mensagens adulteradas ou forjadas;
- **Serviço de Confidencialidade de Comunicação**, tornando as mensagens inescrutáveis por terceiros, enquanto úteis;
- **Serviço de Confidencialidade de Acesso**, que garante a proteção às informações de gerência mantidas na MIB.

Todos os Serviços acima devem estar presentes nas Interfaces de Segurança dos agentes e dos gerentes componentes do Sistema de Gerência, à exceção do último, cuja presença somente é necessária nas entidades agente, pois diz respeito apenas a atividades destes. A diferença entre os dois tipos de Interface de Segurança está representada nas figuras 4.2a e 4.2b.

FIGURA 4.2a: Interface de Gerentes

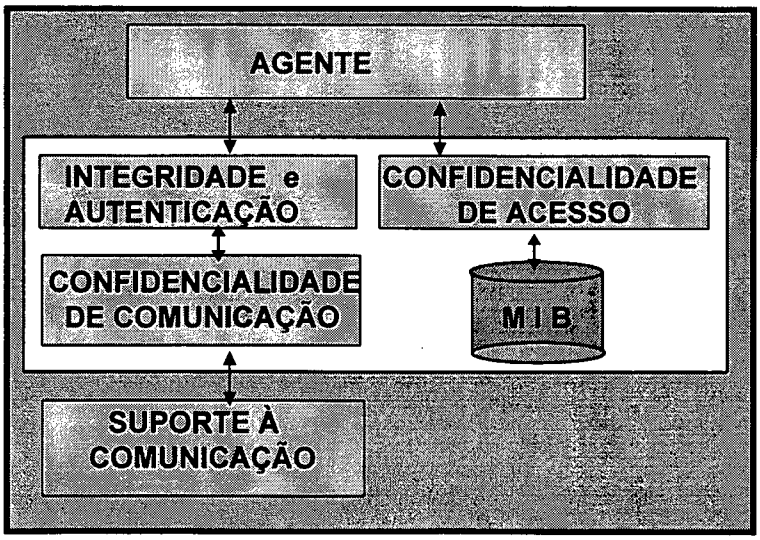

FIGURA 4.2.b: Interface de Agentes

Além dos serviços citados, é de grande importância que as Interfaces de Segurança mantenham registros das ocorrências em *logs* para que seja possível a realização de auditorias, como atividade de gerenciamento de segurança. Nos casos de tentativa de burla da segurança (detecção de mensagens alteradas ou re-apresentação de pacotes fora da seqüência, p.ex.) as Interfaces devem levantar alarmes de segurança para o Sistema de Gerência.

Somente trafegarão na rede (no escopo de Sistemas de Gerência) pacotes de comunicação entre Interfaces de Segurança que encapsulam pacotes de agentes e gerentes, com todos os mecanismos de segurança para evitar possíveis agressões. A interface de segurança emissora é responsável pela incorporação dos mecanismos ao pacote original e a interface do lado receptor é responsável pelas verificações e liberação ou não de pacotes.

O acesso às informações de gerência guardadas na MIB também deve ser restrito. Isto se deve à necessidade de manter as informações sensíveis longe do acesso alheio, uma vez que a liberação indevida destas informações podem gerar ataques à segurança. As informações mantidas em uma MIB dizem respeito ao estado de entidades componentes do sistema de gerência. Este tipo de informação pode ser utilizada para se desferir um ataque, com possibilidade de graves repercussões.

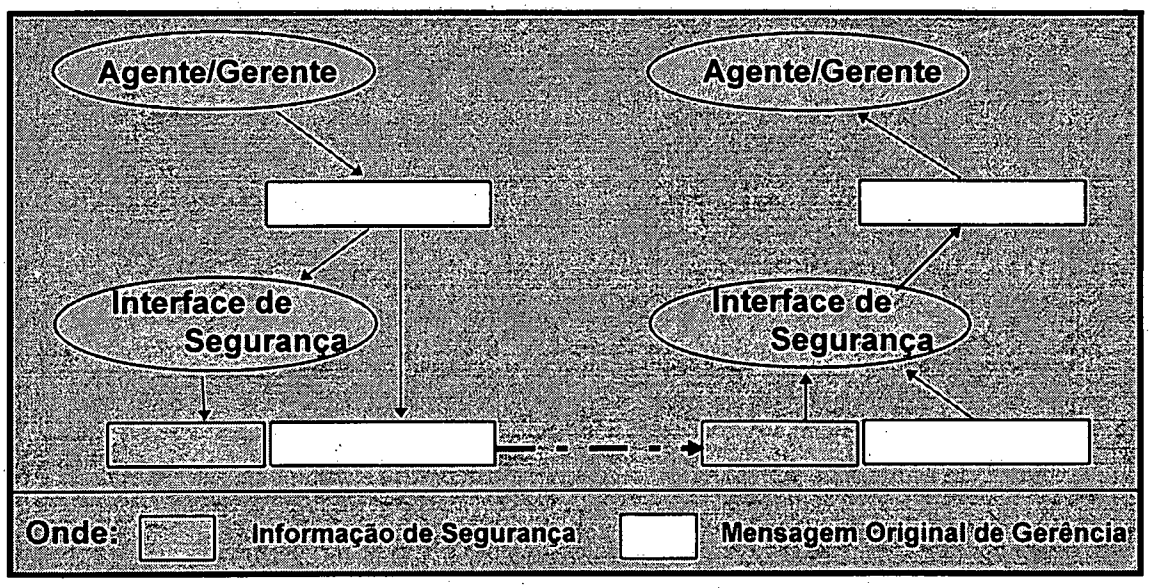

FIGURA 4.3: Formação de uma Mensagem Segura

4.4. Princípios

A implementação da Arquitetura de Segurança para Gerência de Redes deve seguir dois princípios fundamentais:

- Genericidade: ou seja, deve se adequar às diferentes plataformas de gerência disponíveis, sem impedir possíveis integrações ou compatibilizações que já existam entre as mesmas;

- Transparência: a implantação dos serviços de segurança não deve interferir no desempenho e nas características do Sistema de Gerência hospedeiro.

Estes conceitos são flexíveis o suficiente para permitir que implementações sejam feitas sem necessidade de alterações de agentes e gerentes do Sistema de Gerência, ou que estes sejam construídos já visando a Arquitetura de Segurança.

As mensagens de gerência são tratadas como pacotes fechados pela Interface de Segurança, que apenas se encarrega de envolvê-las em uma "cápsula" de segurança e inserir os mecanismos necessários. Estas medidas garantem o caráter genérico de aplicação da Arquitetura de Segurança frente às plataformas, permitindo, então, a sua implantação em diferentes Sistemas de Gerência.

A transparência na implantação dos serviços de segurança junto ao Sistema de Gerência é uma tarefa mais complexa. O esquema idealizado para a comunicação entre a Interface de Segurança e os agentes, gerentes, e mesmo entre Interfaces de Segurança comunicantes deve se basear no mecanismo de comunicação inter-processos (IPC - *Inter Process Communication*) utilizado pelo Sistema de Gerência objeto da implantação da Arquitetura de Segurança.

Outras características, fundamentais para que a Arquitetura de Segurança desempenhe suas funções, são:

a) em toda máquina onde houver entidades de gerência sendo executadas, devem haver Interfaces de Segurança associadas a estas entidades.

b) a comunicação entre agente e Interface (bem como entre gerente e Interface) deverá ser restrita ao domínio da máquina onde estes se encontram para evitar liberação de informações através da rede.

c) a Interface de Segurança deve residir na mesma máquina em que residem a entidade de gerência à qual serve.

A transparência da Interface de Segurança significa que os agentes e gerentes deverão comportar-se da mesma forma caso a Interface esteja ou não implantada. A comunicação entre agentes e gerentes se dará do mesmo modo mas, em vez de estas entidades entregarem mensagens para serem transportadas diretamente aos serviços de suporte à comunicação (ou seja, ao software de rede propriamente dito), elas as entregarão à Interface de Segurança, que tratará as mensagens de modo a incluir atributos (informações de controle) para garantir confidencialidade, autenticação e integridade. Da mesma forma, a recepção de mensagens não mais será direta mas também passará antes pela Interface de Segurança que fará as verificações pertinentes antes de liberar ou rejeitar uma mensagem por motivos de segurança.

4.5. Mecanismos Utilizados

Diversos mecanismos de segurança são necessários para a implementação do Modelo Lógico apresentado acima. A seguir, são apresentados estes mecanismos, relacionados ao(s) serviço(s) que atendem e são discutidos os motivos que levaram à escolha ou definição dos mesmos.

4.5.1. Serviço de Confidencialidade

Para suportar o ***Serviço de Confidencialidade*** (ou privacidade) é necessário o uso de criptografia. A criptografia é a única forma de garantir que uma mensagem que esteja trafegando pela rede e seja interceptada não forneça informações valiosas para o agressor. A

mensagem poderá ser interceptada, mas dificilmente será decodificada sem que a chave esteja disponível. Como já discutido, há dois tipos básicos de criptografia: por chave secreta e por chave pública [PFL 89]. A adoção de uma ou outra forma é função das características desejadas e/ou impostas pelo serviço que se deseja implantar e é um fator importante a ser observado.

O esquema de chave secreta é baseado em uma única chave que tem a função de cifrar e decifrar os textos que são submetidos. Cada par comunicante deve possuir esta chave para que as comunicações entre estas partes sejam estritamente confidenciais. É possível também que um grupo de entidades compartilhe uma mesma chave secreta, se a confidencialidade na comunicação é importante apenas em relação ao exterior deste grupo. No caso de aplicações de tempo crítico, é recomendável, em alguns serviços de segurança, utilizar chave única para evitar a deterioração do desempenho do sistema como um todo.

O emprego de um sistema de chaves públicas[1] é mais aplicado quando o processo de distribuição de chaves entre as partes comunicantes é particularmente sensível. Este esquema evita a necessidade de um transporte e difusão *off-line* das chaves entre as entidades cooperantes (o que, muitas vezes, é impossível em se tratando de software). Mesmo um meio de comunicação considerado não seguro pode ser utilizado para intercâmbio de chaves (quando e se isto for necessário) deste esquema [PFL 89]. Um contratempo é a menor eficiência no processo de cifragem e decifragem de grandes massas de dados em relação ao esquema de chave secreta.

O mecanismo de chave pública pode ser utilizado para garantir a autenticidade, a confidencialidade e a integridade. Com este mecanismo, toda entidade comunicante possui um par de chaves complementares. Cada uma destas chaves decifra o código que a outra chave cria. Uma das chaves é mantida em segredo e a outra pode (e deve) ser "publicada" e tornada largamente disponível através da rede. Conhecer a chave pública não ajuda na dedução da chave correspondente que é mantida secreta, e vice-versa.

---

[1] *Um sistema de chaves públicas prevê a existência de duas chaves: o que uma chave cifra o* *cifra seu par decifra e vice-versa. Maiores detalhes podem ser encontrados em [NEC 90][PFL 89] e no Anexo A.*

Qualquer entidade pode usar a chave pública do destinatário para criptografar uma mensagem a ser enviada para este, e o destinatário deverá usar sua própria chave privada correspondente para descriptografar aquela mensagem. Ninguém mais além do destinatário poderá decifrá-la, porque ninguém mais conhece ou tem acesso à chave privada do destinatário. Nem mesmo o remetente que criptografou a mensagem pode inverter o processo. Isto garante a privacidade, ou seja, a **confidencialidade da informação** que flui pela rede. Um esquema de confidencialidade conseguida com sistema de chave pública é apresentado na figura 4.4, a seguir.

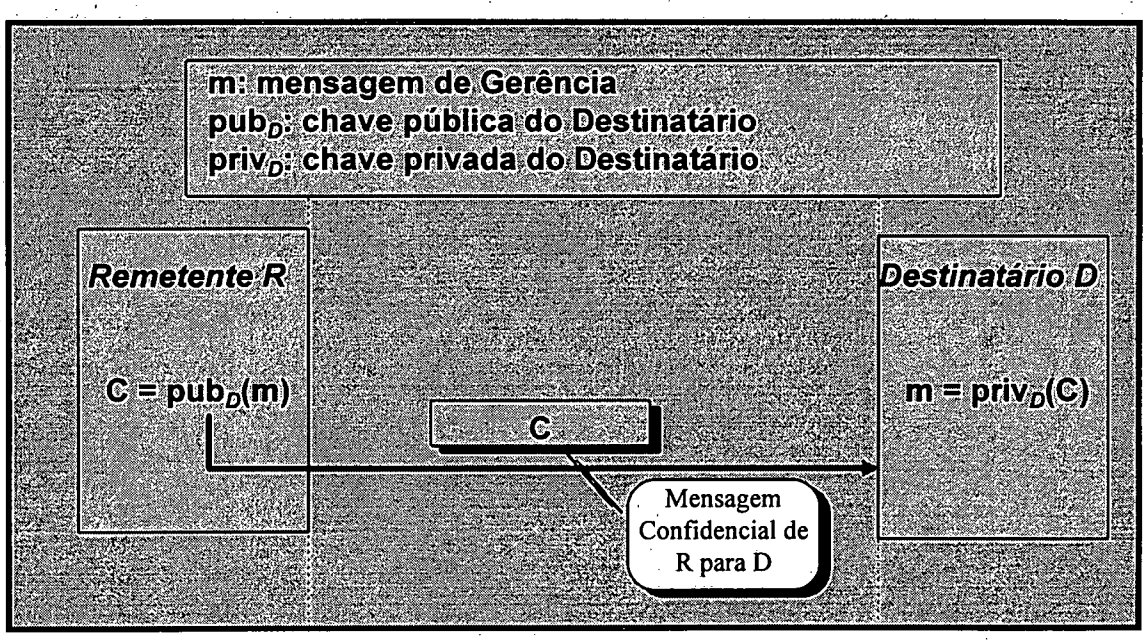

FIGURA 4.4: Confidencialidade usando chave pública

O mecanismo de chave secreta também pode ser utilizado para implantar serviços de segurança. De posse da chave secreta correspondente àquela relação, o remetente da mensagem submete esta à função de criptografia, com a chave secreta como segundo argumento. Como resultado, obtém-se um texto cifrado que deve ser enviado ao destinatário. Este, de posse do texto e conhecendo a chave secreta, pode decifrar e obter a mensagem original. Esta forma também garante a confidencialidade da informação que flui pela rede. A figura 4.5 apresenta o esquema para uso do sistema de chave secreta para obtenção da confidencialidade.

O acesso às informações de gerência guardadas na MIB deve, também, ser restrito. Existem duas formas básicas de prover tal restrição: pela instalação de um **Serviço de Controle de Acesso** próprio ou a adoção de um **Serviço de Confidencialidade**, onde as informações armazenadas na MIB estariam cifradas. Um Serviço de Controle de Acesso deve prover meios de garantir que somente entidades autorizadas (por meio de listas de permissão, por exemplo) tenham acesso à MIB [RAM 95].

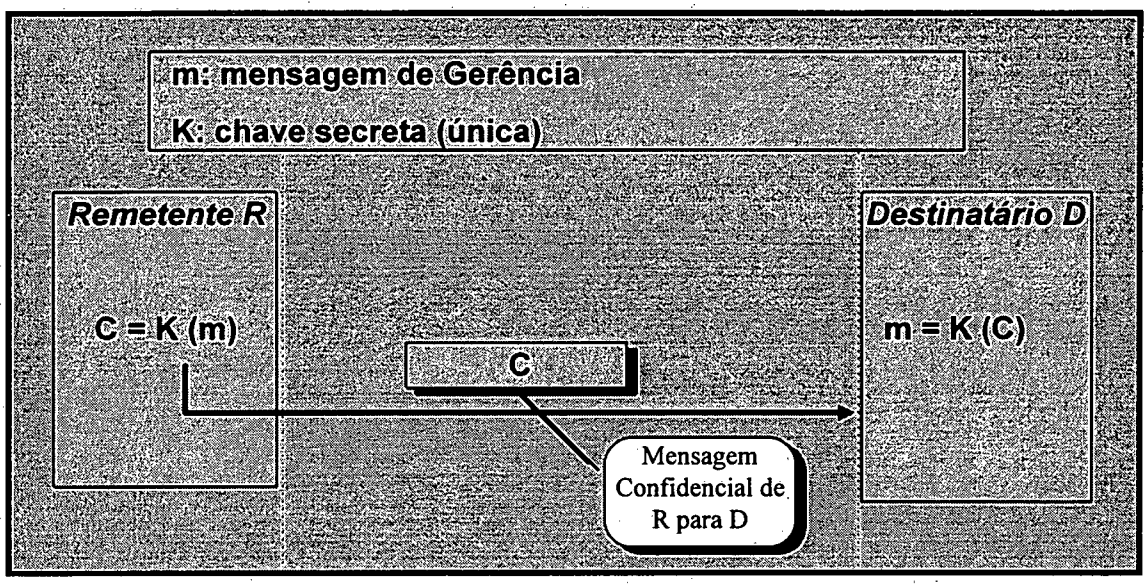

FIGURA 4.5: Confidencialidade usando chave secreta

A estratégia de estabelecer um Serviço de Confidencialidade no acesso à MIB, garantindo que um e somente um agente (o seu criador e mantenedor) terá acesso direto às informações lá contidas, se apresenta como a mais interessante. Três itens motivam esta escolha:

- o mecanismo de confidencialidade já está disponível para outros serviços de segurança (Confidencialidade de Comunicação), eliminando a necessidade de construção de um novo mecanismo de segurança, o controle de acesso;
- elimina a necessidade de criação de uma Interface de Segurança também para a MIB, centralizando nos agentes a implementação dos mecanismos de segurança;
- não possui certas vulnerabilidades presentes nos mecanismos de controle de acesso, como a abertura para o mascaramento.

O acesso à MIB (especificamente ao arquivo que contém as informações) pode ser aberto pois, como as informações lá contidas estarão cifradas, elas não serão úteis para agressores, uma vez que estes não terão tempo hábil para decifrá-las antes que ocorram alterações nas mesmas (invalidando assim todo o trabalho de criptoanálise realizado até ali pelos agressores).

A forma de implantar o Serviço de Confidencialidade no acesso à MIB é o uso de criptografia em todos os acessos à mesma, conforme apresentado na figura 4.6. O agente responsável pela MIB possui uma chave que é utilizada para cifrar todas as informações antes de armazená-las e decifrar as informações quando do acesso. A cifragem das informações contidas na MIB pode ser feita com a mesma chave que o agente utiliza para garantir a privacidade das comunicações ou com uma chave própria para a tarefa, podendo ser inclusive com o uso de uma técnica de chave secreta, mais eficiente em termos de tempo para criptografar e descriptografar. Isto porque o acesso à MIB é completamente independente de todo o processo de comunicação entre entidades, ou seja, as rotinas de acesso à MIB são puramente locais, segundo o modelo adotado; não havendo comunicação entre entidades remotas para efetuar acessos e alterações sobre os dados lá contidos. Tal mecanismo permite inclusive que o acesso em si possa ser realizado sem restrições, mas uma vez que as informações estão criptografadas, não há liberação efetiva das mesmas para aqueles que não possuírem as chaves.

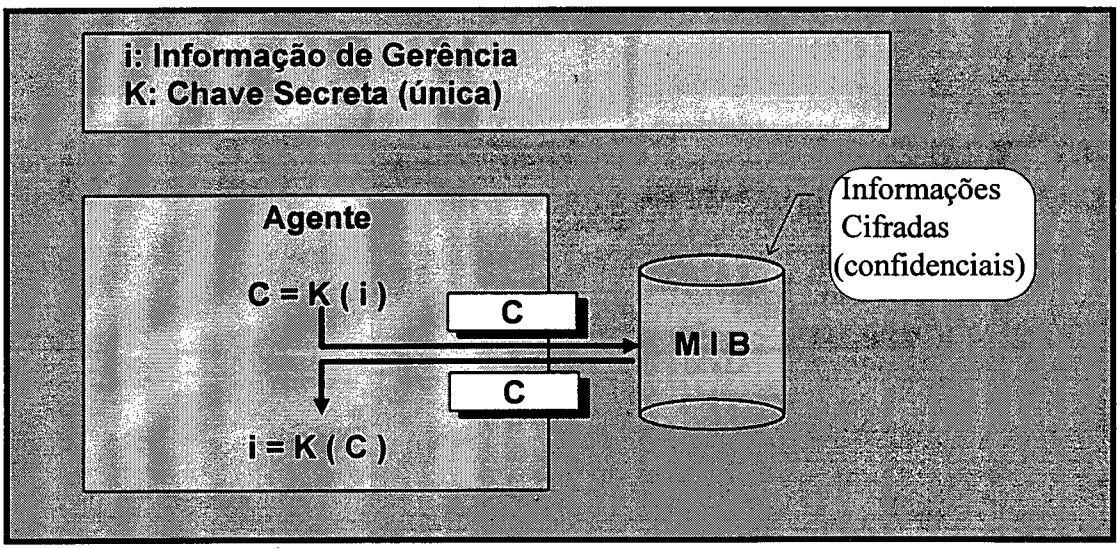

FIGURA 4.6: Confidencialidade de Acesso à MIB

4.5.2. Serviço de Integridade

A alteração de mensagens e o re-seqüenciamento (re-submissão de mensagem antiga) são sérias ameaças pois uma mensagem forjada de um destes modos pode "ser aprovada" no teste de autenticação (uma vez que é uma mensagem originalmente montada e expedida por uma entidade autorizada) e, com isso causar danos ao sistema. Por isso o **Serviço de Integridade** deve estar atento para evitá-las. Duas providências se fazem necessárias:

i) para evitar que uma mensagem alterada (adulterada) seja considerada válida, a ação a ser tomada é a cifragem de um campo que contenha uma forma de *checksum*[2] de toda a mensagem. Com isso se garante a integridade da mensagem, pois um agressor não terá como alterar a mensagem, gerar um novo *checksum* e criptografá-lo pois não possuirá a chave correta. Já o destinatário pode verificar a integridade simplesmente usando a chave devida (que possui) para conferir o *checksum* calculado com o decifrado. Qualquer alteração da mensagem é imediatamente detectada. Pode-se perceber como é constituída uma mensagem com mecanismo de controle de alteração pela fig. 4.7. Este esquema é semelhante ao proposto em [OMU 90] e [GAL 91]. Em um esquema de chave pública (assimétrica), a chave que deve ser utilizada para a cifragem é a privada do remetente da mensagem, e para decifrar, o destinatário deve utilizar a chave pública do remetente.

ii) para evitar o re-seqüenciamento, a técnica aqui proposta é a inclusão de um **campo de sincronização**. Este campo deverá conter dois valores definindo uma seqüência que é determinada a cada interação entre cada par de entidades (ou seja, a cada mensagem, o remetente incluirá o valor atual da seqüência e indicará qual valor deverá ser usado na próxima comunicação entre estas duas entidades). Este **campo de sincronização**

---

[2] *Checksum está sendo usado como nome genérico para uma função calculada sobre toda a* *toda mensagem, e que é matematicamente improvável a criação de outra mensagem que gere o mesmo valor quando submetida à função.*

(que é composto por dois sub-campos, chamados **senha** e **próxima-senha**) também deve ser criptografado. A lei de formação da seqüência não é importante e pode ser implementada diferentemente em cada uma das partes. A figura 4.8 apresenta a concatenação de um campo de sincronização por uma interface de segurança e sua verificação na interface par. A abordagem freqüentemente adotada para este problema (de detecção de re-submissão) é a de sincronização de relógios entre todos os computadores do Sistema de Gerência e o decorrente uso de *timestamps* para garantir o momento do envio da mensagem [GAL 91]. A abordagem aqui proposta é muito mais simples (tanto para implementação quanto para validação) e não menos eficiente, uma vez que exige apenas uma especificação acerca da próxima comunicação a cada interação. Se for adotado um esquema de chave pública, o campo de sincronização deve ser criptografado com a chave pública do destinatário, garantindo a privacidade do campo. O destinatário facilmente decifra o campo utilizando a sua própria chave privada. No caso de esquema de chave síncrono, a simples cifragem deste campo com a chave secreta garante a privacidade.

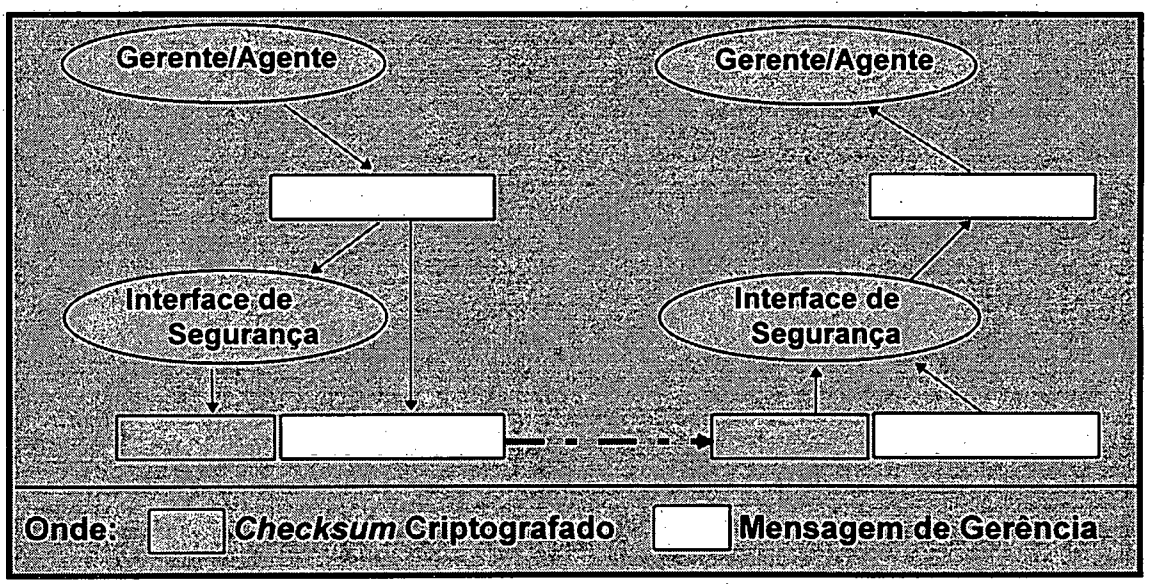

FIGURA 4.7: *Checksum* criptografado para garantir não-alteração

Então, uma parte da integridade das mensagens é garantida da mesma forma que a autenticação (ver item 4.5.3), porque não há como alterar o conteúdo das mensagens

sem que isso se reflita no *checksum*. Tal *checksum*, estando criptografado, não pode ser alterado porque não há como cifrá-lo de modo correto novamente sem que se conheça a chave utilizada. Desta forma, fica garantido que qualquer alteração de uma mensagem por terceiros seja facilmente detectada. Além disso, não é necessário cifrar toda a mensagem, o que representa ganho de eficiência.

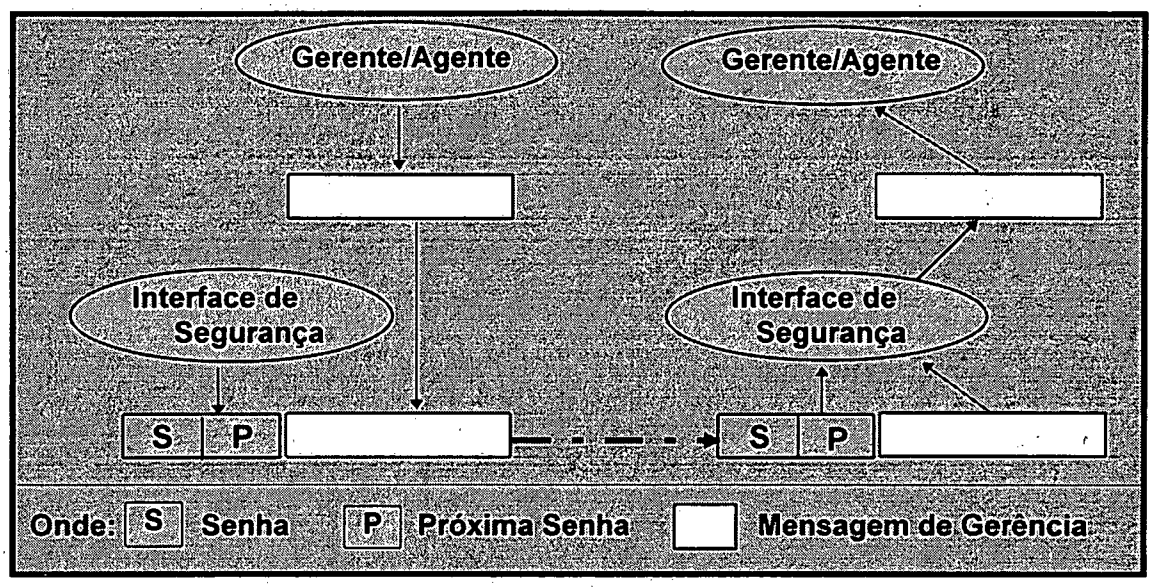

FIGURA 4.8: Campo de Sincronização para garantir não-resubmissão

Evitar que uma mensagem que foi interceptada e armazenada por um intruso possa ser reproduzida em outro momento também é uma tarefa do Serviço de Integridade. Para isto, a proposta apresentada aqui (item *ii* acima) introduz um campo de sincronização que permitirá determinar, a cada passo, se a comunicação está ocorrendo de acordo com o determinado pelas partes. No momento em que a seqüência indicada no sub-campo **senha** não for a correta (a que era esperada, definida na interação anterior), é porque houve tentativa de burla da segurança. Estes sub-campos representam um papel de senha, combinada a cada nova interação entre as Interfaces. O sub-campo **senha** indica o estado no qual a seqüência de comunicação está em determinada interação e o sub-campo **próxima-senha** informa qual deve ser o valor para a próxima interação. Este mecanismo simples, juntamente com o mecanismo de autenticação (seção 4.5.3), garante que sejam desconsidaradaos os re-envios de mensagens guardadas por intrusos, quando ocorrerem. Os valores gerados para os campos podem ser gerados por algoritmos distintos em cada uma das partes.

Um único problema no esquema de senhas proposto no item *ii* se apresenta em uma situação muito pouco provável e que é descrita a seguir: um intruso monitora a linha e armazena as mensagens legítimas (com mecanismos de seqüenciamento, autenticação e de confidencialidade intactos) e descobre que uma determinada mensagem (mensagem k), com sub-campo **senha** contendo o valor 'X', produziu efeitos que interessavam a este intruso (interesses escusos). A partir deste momento, sempre que ele desejar reproduzir os efeitos da mensagem de senha 'X', basta monitorar a linha esperando por uma mensagem que possua o sub-campo **próxima-senha** indicando 'X' (na verdade deve deduzir o conteúdo a ser esperado a partir da mensagem k-1), e neste momento introduzir indevidamente a mensagem previamente interceptada e armazenada que tem como campo **senha** o valor 'X'. Além disso, a possibilidade de re-submissão somente existe em um sentido, ou seja, somente poderá ser re-introduzida a mensagem se uma condição específica se reproduzir com as mesmas duas interfaces enviando mensagens no mesmo sentido e com a mesma senha esperada. A figura 4.9 apresenta um possível caso de escuta e re-submissão.

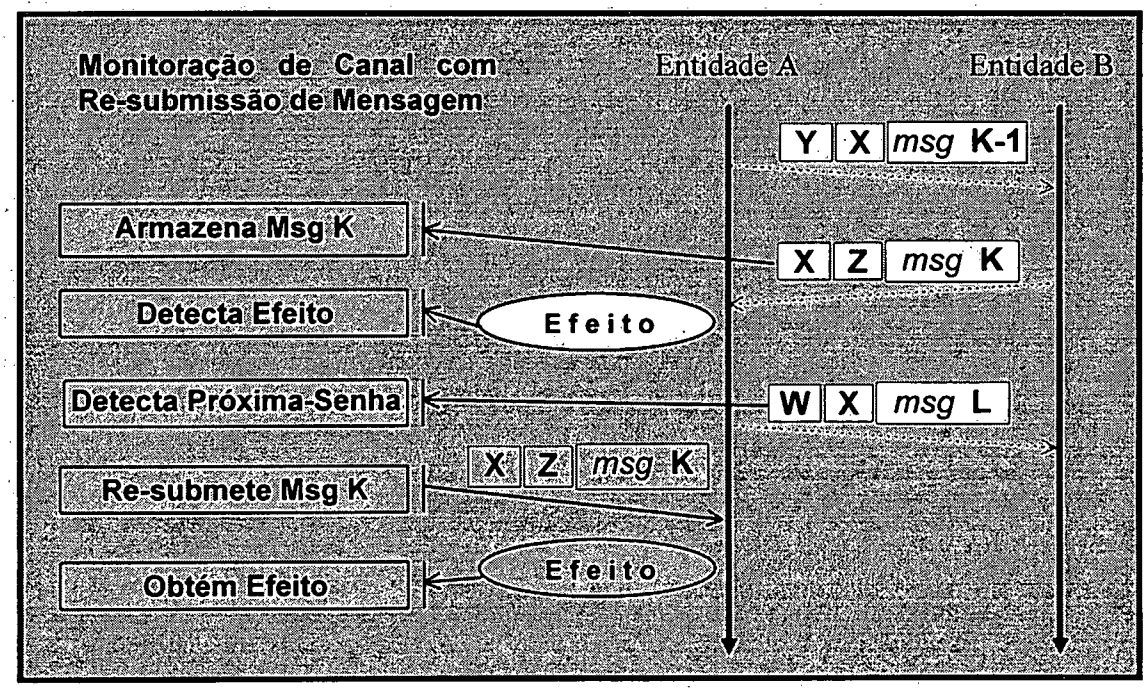

FIGURA 4.9: Exemplo de Re-submissão

À primeira vista, a simples cifragem dos campos que indicam os valores de sincronização garantiriam a privacidade, evitando que se pudesse saber quando uma mensagem de senha 'X' é esperada. Mas isto não basta, uma vez que apesar de restringir em muito a possibilidade de ocorrer o problema de reprodução de mensagem antiga, o intruso não necessita saber o valor original dos campos, podendo conhecer apenas aos valores já criptografados. Em determinado momento, uma mensagem circula autorizando a próxima mensagem com valor de **senha** determinado. A próxima mensagem conterá um valor no campo de senha que estará permanentemente relacionado ao valor anteriormente verificado em **próxima-senha**. Basta, então que se preserve as direções de fluxo das mensagens originais, que o intruso poderá descobrir quando inserir a mensagem armazenada.

Por outro lado, se a lei de formação das seqüências (senhas) for tal que não gere valores repetidos dentro de um intervalo de tempo considerado grande o suficiente, o problema pode ser considerado equacionado. Um exemplo seria a implementação de um algoritmo de geração de números pseudo-aleatórios cujos ciclos são muito grandes e mantendo uma tabela dos valores recentemente emitidos como **próxima-senha**, para evitar o seu reaproveitamento muito cedo. Evidentemente, esta abordagem não elimina o problema, mas a possibilidade de ocorrer tal evento é tão ínfima que podem ser consideradas satisfeitas as condições de segurança. Isto também é verdade porque é impossível garantir que não haja repetição de valores, uma vez que, por maior que seja a área reservada para os campos de sincronização (em *bytes*), esta área sempre será finita, e por isso, sempre será necessário o reaproveitamento de senhas no futuro.

Esquema semelhante é empregado na nomeação de recursos de sistemas operacionais distribuídos, como Amoeba e Chorus, quando da formação de *capabilities* [TAN 92] [ROZ 89], com sucesso, em função da baixa probabilidade de repetição dos valores dentro de um tempo de vida útil para os sistemas. O mesmo pode ser afirmado em relação às senhas utilizadas aqui, uma vez que o tempo de vida das mesmas é infinitamente menor que os recursos citados acima, já que elas serão substituídas a cada nova interação entre entidades de gerência.

Um adendo à implementação acima permite eliminar completamente o problema citado. Este adendo é realizado com a incorporação de um terceiro campo na mensagem que conteria uma chave (no esquema de chave secreta), que pode ser denominada **chave da sessão**, e que seria empregado para criptografar os campos de sincronização. O campo chave da sessão, ao ser inserido na mensagem, seria criptografado pela chave utilizada na comunicação para garantir a confidencialidade. A chave deve ser alterada a cada interação (sendo por isso caracterizada como chave da sessão), podendo ser, por exemplo, uma função da data do sistema (não é *timestamp* porque não há necessidade de sincronização). Os campos de sincronização devem ser cifrados pela chave da sessão. Isto garantiria a não-criação e o não-reaproveitamento de mensagens, bem como a impossibilidade de ocorrência do problema narrado acima, uma vez que a chave utilizada para cifrar os dados é alterada a cada interação (ou, pelo menos, antes de se reaproveitar uma senha nas interações entre as entidades envolvidas), impedindo assim que a monitoração forneça meios para identificar os valores dos campos de sincronização. O esquema de chave de sessão é representado na figura 4.10.

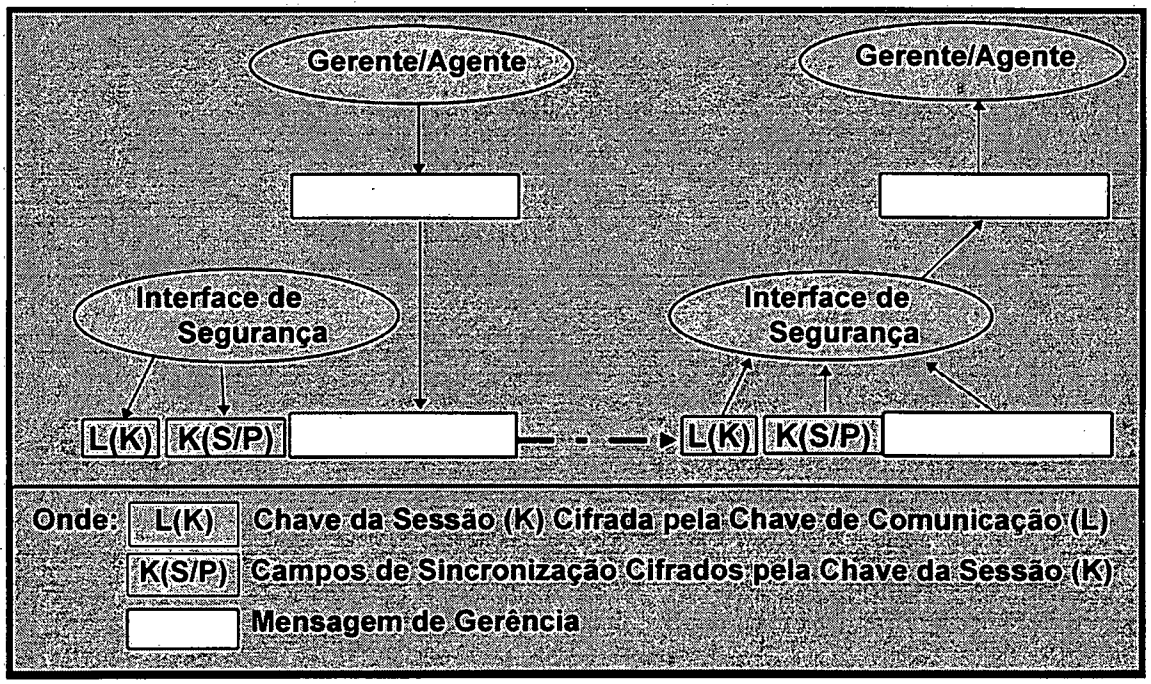

FIGURA 4.10: Exemplo de Uso de Chave da Sessão

4.5.3. Serviço de Autenticação

Considerando o esquema de chave pública, a **autenticação** ou verificação de autenticidade pode ser conseguida com a cifragem prévia da mensagem pelo originador, usando a chave privada do mesmo, que é de conhecimento apenas do autor. O destinatário pode, então, verificar a origem, decifrando a mensagem utilizando a chave pública do remetente, que é de conhecimento geral. Deve haver um campo na mensagem, ao menos, que seja de conhecimento de ambos os comunicantes (ou que possa ser calculado a partir da mensagem) para que se possa realizar a verificação. Se esta tarefa for bem sucedida, isto prova que o remetente é o verdadeiro originador da mensagem, além de certificar que a mensagem não foi posteriormente alterada por terceiros, porque somente o remetente possui a chave privada que cifrou a mensagem, que forma par com a chave pública utilizada para decifrar a mesma.

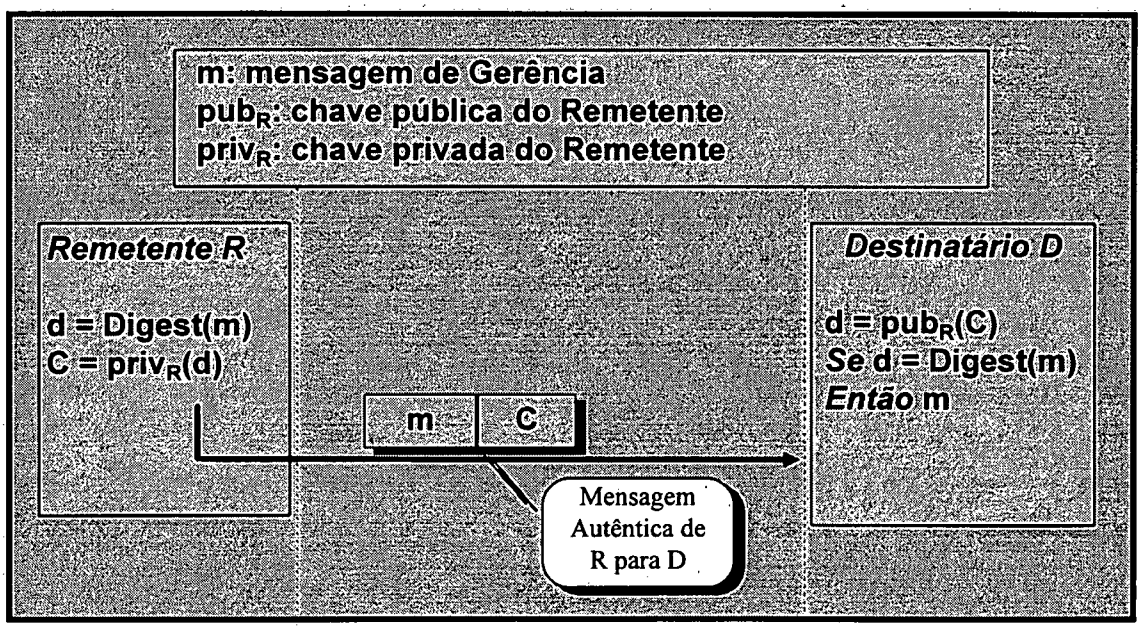

FIGURA 4.11: Autenticação com chave pública

Para provar a origem não é necessário criptografar toda a mensagem, mas apenas algo que a sintetize em um pequeno campo. Esta é a idéia dos campos de *checksum* (ou CRC) que são, na verdade, uma função de toda a mensagem. O princípio é o mesmo de um *checksum* para verificar se uma comunicação não foi afetada, com a ressalva da importância de que não seja computacionalmente viável a criação, por parte de um intruso,

de uma mensagem substituta que, submetida à função que calcula o checksum, produza idêntico resultado. O conceito utilizado é de *digest*, que é uma função *hash one-way* (ou seja, que não é possível deduzir o valor original a partir do resultado da função)[RIV 92]. O *digest* formado é criptografado com a chave privada do remetente para garantir, assim, a origem, impedindo que um intruso ou entidade mascarada produza uma mensagem em nome de outrem. Isto porque não há como criptografar corretamente o *digest* forjado sem conhecer a chave privada do remetente. E se isto não for feito, o destinatário não irá considerar a mensagem como válida. A figura 4.11 representa um mecanismo de proteção contra mensagens forjadas, utilizando chave pública.

No caso de uso de um esquema de chave secreta, o mecanismo é o mesmo, mas há uma restrição em relação à garantia de origem. Se a senha for única para diversas entidades comunicantes (que formam um grupo confiável entre si), a garantia de que a mensagem teve sua origem em uma entidade específica não vale. O que se pode garantir é que a mensagem teve origem dentro do grupo, considerado confiável, que detém a chave. Isto pode não representar um problema, uma vez que, se o grupo é verdadeiramente confiável, não há razões para suspeitar que entidades do grupo podem estar forjando mensagens. No caso de cada par comunicante manter uma senha secreta própria, esta restrição deixa de existir. O mesmo mecanismo de autenticação apresentado na figura 4.11 com a aplicação de esquema de chave secreta é apresentado na figura 4.12.

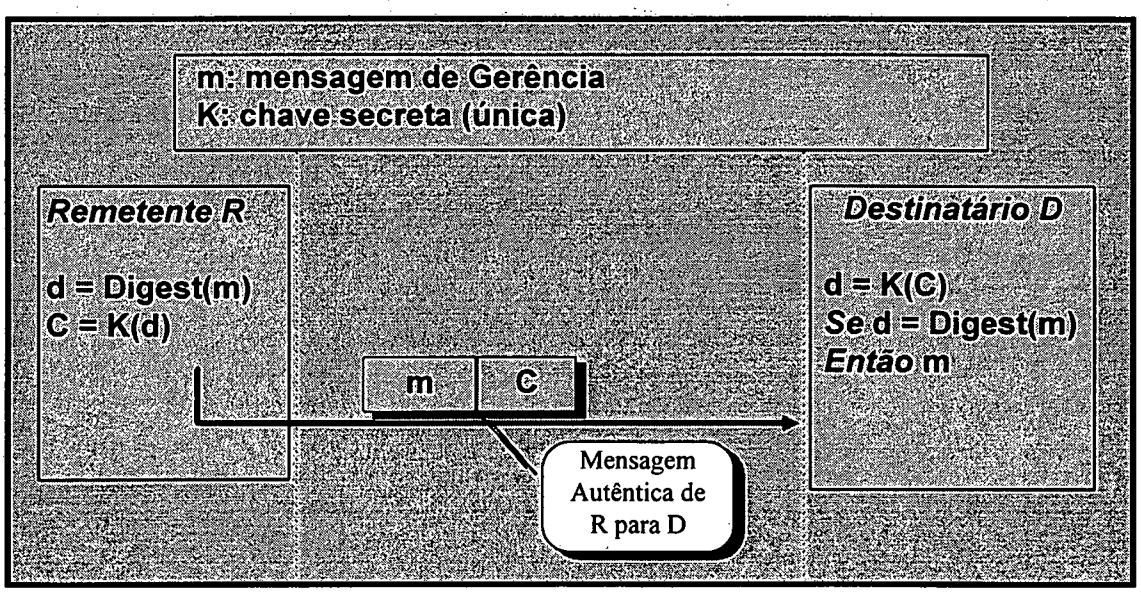

FIGURA 4.12: Autenticação com chave secreta

O **Serviço de Autenticação** deve, então, garantir que a origem das mensagens de gerenciamento são entidades legítimas para evitar a execução de ações indevidas ou o acesso a informações por terceiros. Uma observação atenta à providência citada na seção 4.5.2., referente a alteração de mensagens, permite concluir que a própria integridade oferece meios de aferir a autenticidade das mensagens, uma vez que somente o interlocutor autêntico conhecerá a chave que deve ser empregada e com isso poderá gerar os campos criptografados de acordo com o esperado. A autenticação, então, está automaticamente incluída no Serviço de Integridade. Esta vantagem é resultado da forma como foram analisados e equacionados os problemas de segurança que podem afetar Sistemas de Gerência.

4.5.4. Outras discussões

A abordagem sugerida para a solução dos problemas tem ainda a vantagem de permitir a instalação independente entre o Serviço de Confidencialidade e os outros dois Serviços (de Integridade e de Autenticação). Esta vantagem provém do fato de que, qualquer que seja o sistema de criptografia a ser empregado, a cifragem das mensagens completas sempre representa uma sobrecarga ao sistema e a Confidencialidade pode ser dispensável em determinadas situações, obtendo-se nestes casos, ganhos de desempenho.

Evidentemente, o desempenho final da comunicação do Sistema de Gerência será afetado pela carga de trabalho a mais que é incluída em função da implantação da Interface de Segurança em ambos os lados das comunicações. Mas esta sobrecarga é perfeitamente assimilável quando comparada aos benefícios oriundos da mesma. A proposta de submeter toda e qualquer mensagem do Sistema de Gerência à Interface de Segurança é sustentada por um único motivo: a importância das informações de gerência e do Sistema de Gerência como um todo para o bom funcionamento da rede e seus serviços. Levando isto em consideração, a carga de trabalho representada pela atividade de criptografia (na cifragem e decifragem de mensagens) é perfeitamente justificável, em função da segurança que oferece e sensibilidade desta área.

Um Serviço de Não-Repudiação, apesar de facilmente conseguido, é totalmente dispensável em um Sistema de Gerência, uma vez que este é um sistema cooperativo e perfeitamente confiável entre as partes que o compõem. Isto também elimina um problema relativo ao uso do DES, quando na implementação de assinatura digital. Em ambientes não confiáveis, DES não previne que um dos lados forje uma mensagem como sendo de um interlocutor. Uma situação como esta não é aceitável na arquitetura proposta (o que é aceitável em se tratando de um Sistema de Gerência), e, portanto considera-se viável o uso de DES nestas condições.

4.6. Conclusão

Neste capítulo, além de especificado um modelo para implementação de uma Arquitetura de Segurança para Gerência de Redes, foram levantadas diversas hipóteses associadas a possibilidades de implementação deste modelo, apresentando alternativas, características, vantagens e desvantagens; referentes a cada um dos serviços de segurança a implementar.

O Modelo da Arquitetura de Segurança oferece uma definição para levar segurança a Sistemas de Gerência de Redes. A Arquitetura define uma Interface de Segurança que é responsável por manter a segurança em sistemas de gerência de redes. A segurança oferecida diz respeito à confidencialidade das comunicações e das informações mantidas na MIB, à integridade de mensagens no conteúdo e no tempo e à autenticação das entidades pares, além de criar condições objetivas para outros serviços de segurança. Para realizar estas tarefas, são definidos Serviços de Segurança que devem ser implementados pela Interface de Segurança da Arquitetura.

O Modelo é genérico o suficiente para suportar diferentes implementações conforme idiossincrasias de sistemas de gerência ou de ambientes operacionais distintos. No capítulo seguinte, são apresentados aspectos que devem ser observados e decisões que devem ser tomadas quando da implementação do modelo.

5 Aspectos de Implementação

O Modelo Lógico da Arquitetura de Segurança apresentado no capítulo anterior é flexível o suficiente para proporcionar distintas formas de implementação da mesma. Isto permite a adaptação dos conceitos lá preconizados a diversas situações diferentes que podem ser encontradas quando da implantação de segurança em Sistemas de Gerência com características de comunicação próprias, e mesmo a escolha entre variações de implementação de modo a garantir melhor desempenho quando uma forma se mostrar mais adequada que outras.

Isto deriva de o Modelo Lógico da Arquitetura ser um conjunto de serviços, princípios e funcionalidades que não impõe uma estrutura. Este conjunto de serviços, princípios e funcionalidades é que garante que conceitos incorporados na Arquitetura serão aplicados, se as funcionalidades forem corretamente implementadas, os princípios respeitados e os serviços estabelecidos.

A proposta de análise da sobrecarga do sistema, idealizada e apresentada aqui, vai além de uma mera medição em um estudo de caso, onde a variável tempo poderia ser verificada. Isto porque a verificação de uma situação particular está sujeita a diversos fatores que interfeririam na perfeita avaliação do problema. Exemplos seriam as diferentes capacidades de processamentos das CPUs utilizadas, a carga de trabalho momentânea das CPUs, o fluxo momentâneo da rede e a própria localização de agentes e gerentes em redes mais complexas (com sub-redes e roteadores, por exemplo) e mesmo possíveis diferentes implementações de algoritmos de criptografia. A segurança no acesso à MIB, por envolver basicamente aspectos de criptografia local (sem trânsito de informações pela rede) não foi considerada na implementação de teste.

5.1. Opções de Implementação

Então, apesar de ter se realizado uma implementação piloto a fim de verificar a viabilidade/aplicabilidade do Modelo de Arquitetura de Segurança, a análise do trabalho extra

incluído será realizada a seguir com parâmetros genéricos e, por isso, isentos de interferências externas indesejáveis e menos sujeita a erros de interpretação.

5.1.1. Codificação

As possíveis variações de implementação que existem estão relacionadas a diversos fatores como, por exemplo, a disponibilidade de código fonte, a forma de interação entre agentes e gerentes e a escolha de determinado método de criptografia. Duas formas básicas se apresentam, então, para a implementação da Interface de Segurança, baseadas na disponibilidade ou não de código fonte dos agentes e gerentes envolvidos. Se o código fonte está disponível (ou se os agentes e gerentes ainda estão por serem implementados) é recomendável a implementação "embutida" no próprio código, por diversos motivos; como a maior eficiência, evidentemente, e a redução do número de interfaces de software necessárias, o que traz maior confiabilidade e reduz a complexidade da implementação, tornando-a menos sujeita a erros.

Se o sistema de gerência não disponibilizar o código fonte dos agentes e gerentes e/ou se pretender utilizar os mecanismos básicos oferecidos na forma de pacote fechado, a alternativa é partir para a implementação de uma camada completa de software, independente, e que se interponha entre o Sistema de Gerência e as Camadas de Suporte à Comunicação entre os elementos do Sistema.

A implementação da Interface deve levar em consideração a API (*Application Programming Interface* - interface de programação[1]) que é utilizada e, de algum modo, interferir neste processo de modo a filtrar todas as comunicações entre entidades cooperantes. Esta forma de implementação, apesar de ser bem mais complexa do que a apresentada acima, tem a vantagem de ser mais flexível, podendo também ser adotada na situação anterior (fontes disponíveis).

---

[1] *Conjunto de procedimentos que implementam uma metáfora pela qual um processo ocess comunica-se com outro(s) (IPC - Interprocess Communication - comunicação entre processos)*

5.1.2. Criptografia

Em relação ao método de criptografia, a primeira escolha diz respeito ao uso de um sistema de chave única (secreta) ou de um sistema de chave dupla (pública-privada). Ambos os sistemas podem ser utilizados em função das características e necessidades de um sistema de gerência (conforme já foi discutido nos capítulos anteriores), sem distinção entre os métodos. Todos os serviços de segurança necessários para Sistemas de Gerência (integridade, autenticidade e confidencialidade) são implementáveis por qualquer dos sistemas de chaves. A discussão então recai sobre a eficiência e robustez dos métodos que implementam os sistemas de criptografia.

Para cada classe de sistema de chaves, atualmente, há um método de criptografia que pode ser considerado padrão:

- DES - *Data Encryption Standard*: para sistema de chave única;
- RSA - *Rivest, Shamir e Adleman*: para sistema de chave dupla.

No Anexo A são apresentados maiores detalhes sobre ambos, que também podem ser encontrados em [PFL 89], [GAR 91] e [TAN 88].

Reconhecidamente, o método DES é mais eficiente para cifrar e decifrar mensagens do que o método RSA [PFL 89]. Isto porque, enquanto RSA exige que, para cada bloco de texto que se deseja cifrar, um número decimal muito grande (200 dígitos decimais é um tamanho comum) seja elevado a uma potência com o mesmo número de dígitos (o que não é tarefa trivial para os computadores convencionais de hoje); DES usa uma série de iterações aritméticas e lógicas, relativamente simples na implementação e execução.

Em relação à robustez dos métodos, ambos apresentam pequenas falhas como, por exemplo, algumas chaves geram textos cifrados mais facilmente decifráveis por cripto-análise do que outras, mas nenhuma falha séria que os invalidem. DES necessita, entretando, que a aplicação não seja pura, e sim, baseda em algum dos modos padronizados de utilização

do algoritmo básico, como os descritos no Anexo A. Apesar disso, DES e RSA são os dois únicos métodos matematicamente aceitos como confiáveis [PFL 89], para implantar sistemas simétricos (DES) e assimétricos (RSA).

5.1.3. Formato das mensagens

Ainda em relação à eficiência, pode-se comparar as técnicas em função da quantidade de informação 'redundante' que estas acrescentam às mensagens originais para atingir os objetivos. Claude Shannon estabeleceu alguns princípios de uma boa técnica de criptografia [PFL 89 citando SHA 49] e, dentre eles propõe que o tamanho do texto cifrado não deve ser maior do que o do texto original. Isto porque a quantidade de informação a mais não traz nenhuma informação útil e facilita o trabalho de inferência de um cripto-analista. Na verdade, ambos os métodos escolhidos para estudos (DES e RSA) seguem este princípio e não acrescentam informação alguma à mensagem original quando da cifragem, mantendo o tamanho inicial do texto inalterado. Mas, não é possível esquecer que os métodos citados oferecem, em princípio, somente a confidencialidade.

Por outro lado, a Interface de Segurança, para atingir seus objetivos de integridade e de autenticidade necessita acrescentar informações às mensagens de gerência (e sobre aquelas informações aplicar um método de criptografia). Estes aspectos serão analisados a seguir, de modo independente em relação a cada serviço de segurança e após, uma compilação da sobrecarga representada pelo conjunto de serviços. Serão também apresentadas outras decisões de implementação no que diz respeito aos mecanismos de *digest* e senhas, componentes do modelo, que são agregados pelos serviços implantados.

Em uma tentativa de generalização, é difícil (senão impossível) estimar o tamanho médio de uma mensagem de gerência em função da variedade de informações que esta pode conter e pela liberdade de implementação que é fornecida ao desenvolvedor/projetista das aplicações de gerência. Desta forma, as mensagens podem ser compostas por alguns poucos *bytes* ou podem ser mensagens grandes. Por conta disto, a melhor abordagem é tentar estimar qual seria um percentual de sobrecarga máximo admissível

para uma instalação. Este referencial pode ser inferido pelo administrador/gerente da instalação, baseado em constatações e necessidades tais como:

- o grau de sobrecarga que o Sistema de Gerência já causa, por si só, à rede. Os Sistemas de Gerência representam, reconhecidamente, um tráfego adicional que, se mal dimensionados, podem transtornar o uso da rede como um todo. Desta forma, é importante equacionar o percentual a mais de sobrecarga que uma rede ainda poderia suportar, mantendo-se utilizável.

- as diferentes necessidades de segurança podem influir decisivamente nas opções de implementação (principalmente nos aspectos relacionados com quantidade de informação agregada às mensagens de gerência), permitindo que a sobrecarga seja menor ou maior. Então, se a segurança for um fator crítico, uma sobrecarga maior é plenamente justificável.

Conclui-se então que, enquanto a capacidade ainda ociosa da rede é um fator limitante, a sensibilidade da rede representa um fator de estímulo de uso de serviços de segurança. A figura abaixo (5.1) representa a idéia da faixa disponível.

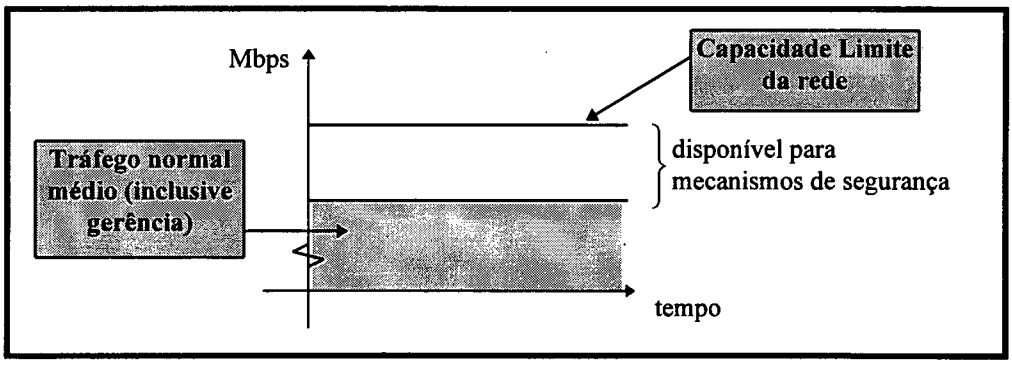

FIGURA 5.1: Sobrecarga máxima para Mecanismos de Segurança

A escolha do mecanismo de *digest* apropriado é de grande importância para a segurança, pois deve representar a mensagem original de forma única. As características desejadas de um algoritmo de *digest* são:

a) deve ser computacionalmente improvável que duas mensagens produzam o mesmo *digest;*
b) deve ser computacionalmente improvável produzir uma mensagem a partir de um *digest* especificado;

c) deve produzir *digests* de tamanho fixo, não importando o tamanho da entrada;
d) deve ser de cálculo rápido.

As características (a) e (b) são as que garantem a segurança do método, dando a ele funções de assinatura ou impressão digital. Cifrando este campo, torna-se altamente improvável a alteração da mensagem sem que isso possa ser percebido. A propriedade (c) é muito útil na implementação de protocolos seguros, uma vez que estabelece um padrão de tamanho para o campo; enquanto a característica (d), genérica, leva a que se tenha pouca sobrecarga de tempo de processamento.

O algoritmo escolhido para implementação de teste do Modelo é conhecido por MD5 - *Message Digest*, especificado no RFC 1321 [RIV 92]. MD5 toma como entrada uma mensagem de tamanho arbitrário e produz um *digest* de 128 *bits* (16 *bytes*) como saída. Atende também as características (a) e (b) acima e foi projetado para ser rápido especialmente em máquinas de 32 *bits*. Detalhes do algoritmo são apresentados no Anexo A.

Outros algoritmos, como CRC-16 ou CRC-32 [TAN 88] poderiam ser utilizados, pois atendem plenamente às exigências (a), (b), (c) e (d), gerando *digests* de 2 (dois) *bytes*. A única ressalva que se faz é em relação à característica (a), quando comparados ao MD5 (e outros da mesma família - MD2 e MD4, por exemplo); pois, mesmo considerando-os idempotentes entre si, a freqüência com que ocorrem repetições do *digest* para mensagens distintas é quatro vezes maior no caso de um CRC (em função do menos número de *bytes* utilizados). Também aqui a decisão tomada na especificação pesa na escolha do número de *bytes* a utilizar e, em decorrência disto, do algoritmo adotado e do grau de segurança atingido.

Em relação ao mecanismo de senhas, proposto no capítulo anterior, a decisão foi de utilizar 4 *bytes* para cada um dos campos (senha e próxima-senha). O Modelo de Interface não estabelece o tamanho que os campos devem ter, ficando então a critério do implementador a decisão. Evidentemente a decisão passa pela análise do nível de segurança contra a ameaça de re-submissão de mensagens que se deseja, bem como em relação ao

número de mensagens trocadas entre as entidades em questão, dentro de um período determinado. Uma breve análise das possibilidades é feita a seguir, com apoio da Tabela 5.1.

Número de *Bytes* (a)	Valores Possíveis (b)	Probabilidade de Repetição em amostra de 100 (c)	Probabilidade de Repetição em amostra de 256 (d)	Probabilidade de Repetição em amostra de 1024 (e)
1	256	0,39	1	4
2	65.536	$0{,}1536 \times 10^{-2}$	$0{,}3900 \times 10^{-2}$	$0{,}156 \times 10^{-1}$
3	16.777.216	$0{,}5960 \times 10^{-5}$	$0{,}1526 \times 10^{-4}$	$0{,}610 \times 10^{-4}$
4	4.294.967.296	$0{,}2328 \times 10^{-7}$	$0{,}5960 \times 10^{-7}$	$0{,}238 \times 10^{-6}$

Tabela 5.1: Análise do mecanismo de senhas

5.1.4. Geração de Senhas

O problema consiste na não repetição de valores no sub-campo senha dentro de um período considerado curto ou com freqüência curta. Se tomarmos uma lei de formação dos valores que somente repita os mesmos após todos terem sido utilizados, o número de mensagens que devem ser trocadas antes que haja uma repetição é mostrada na coluna (b) da Tabela 5.1. Por exemplo, uma progressão aritmética (PA) de razão ímpar[2].

A utilização de um algoritmo de geração de números aleatórios pode dar a possibilidade de verificar a probabilidade de repetição de um conjunto de valores (já gerados) dentro da faixa possível em função da quantidade de *bytes* adotada. Desta forma, pode-se ter uma boa indicação para a decisão sobre quantos *bytes* são necessários para implantar a segurança requerida. As colunas (c), (d) e (e) da Tabela 5.1 apresentam a **maior probabilidade** de repetição dos últimos 100, 256 e 1024 valores, respectivamente, em um algoritmo de geração de números randômicos com distribuição aleatória. A **menor probabilidade** é dada sempre pela função inversa dos valores da coluna (b).

---

[2] *Razão ímpar é importante para garantir a utilização de todos os valores desejados. Por* ...s. P... *propriedades matemáticas, uma PA de razão par produz um ciclo na exata proporção do valor da razão, gerando apenas K/R valores (com K sendo o universo possível e R a razão). Por sua vez, uma PA de razão ímpar evita este problema, gerando todos os valores possíveis (K valores) [KOL 87].*

A Tabela 5.1 pode ser facilmente aumentada tanto em número de *bytes* utilizados para representar a senha quanto para aumento do grau de segurança necessária, usando-se a expressão (1) da Equação 5.1, a seguir:

$$\frac{1}{2^b} \leq p \leq \frac{K}{2^b} \qquad (1)$$

Onde: p = probabilidade

b = número de *bits* utilizados para a senha

K = tamanho da série de valores que se deseja analisar quanto à repetição

EQUAÇÃO 5.1: Probabilidade de repetição em relação ao tamanho da senha

Um algoritmo de geração de números com distribuição aleatória dos valores não garante que um determinado valor não ocorrerá dentro de período arbitrário. Por outro lado, um algoritmo de geração linear (como uma P.A. de razão ímpar) é muito simples e garante o período mínimo a que se propõe, ou seja, todo o espectro de valores possíveis para o número de *bytes* utilizados são gerados antes da seqüência se repetir. Isto significa que, ao optar-se por 4 (quatro) *bytes* de senha, poderão ser enviadas mais de 4 (quatro) bilhões de mensagens de gerência entre cada par de entidades comunicantes do sistema, antes que se possa efetuar um ataque por re-submissão de mensagem.

A coluna (b) da Tabela 5.1 também pode ser lida, então, como o período da série possível para a quantidade de *bytes* especificada. O tempo necessário para cálculo de um valor de senha utilizando-se uma P.A. como esta é desprezível (uma operação de adição).

5.2. Implementação

A sobrecarga global mínima em termos de agregação de dados às mensagens seria o uso de um algoritmo de CRC como *digest* (2 *bytes*), e mais 1 *byte* por mensagem para senha e 1 *byte* por mensagem para próxima-senha. Em relação a tempo, o algoritmo DES é o

mais eficiente para cifrar a mensagem, o *digest* e as senhas, uma vez que este conjunto é uma seqüência simples de bytes, e, para tal, o DES leva vantagem sobre o RSA (detalhamento no Anexo A). Para cálculo do *digest*, pode-se inferir dos algoritmos que o CRC-16 é mais rápido do que CRC-32 e MD5, em função da menor quantidade de ciclos necessários e maior simplicidade. Uma PA é a melhor opção para geração de senhas quando comparada a outros algoritmos de geração, como números aleatórios, conforme analisado na seção 5.1.4.

Esta "configuração", porém, não garante muita segurança, apesar de afastar invasores menos persistentes e/ou menos habilidosos. As escolhas mais seguras, como se pode perceber pela análise realizada nos ítens 5.1.3 e 5.1.4, direcionam para o uso de MD5 com 16 (dezesseis) *bytes* como algoritmo de digest e 4 (quatro) *bytes* para senhas e outros 4 (quatro) *bytes* para a próxima-senha. Quatro *bytes* podem ser considerados suficientes para senhas porque permitiria, juntamente com o algoritmo de Progressão Aritmética, o uso de 4 bilhões de mensagens de gerência entre cada par de entidades do sistema de gerência, antes que houvesse uma brecha para ataque por re-submissão, conforme já descrito na seção 5.1 e como pode ser analisado a partir da tabela 5.1. O algoritmo de criptografia RSA é considerado pela bibliografia mais robusto que o DES [PFL 89] [COO 89]. Em compensação, é menos eficiente, como já citado, além de ter implementação bem mais complexa. Já para o cálculo das senhas, o seguro é justamente o mais eficiente, ou seja a Progressão Aritmética, como já discutido anteriormente.

A abordagem adotada para implementação foi intermediária, garantido segurança com eficiência. Foram utilizados 16 *bytes* para *digest* (MD5) e 4 *bytes* para senha e próxima-senha. Para criptografia foi usado o DES e a já comentada P.A. como geradora de senhas. A sobrecarga dos 24 *bytes* adicionais é mais do que suficiente para garantir a integridade e autenticidade das mensagens por um tempo razoavelmente longo (o que significa até que elas não tenham mais valor). A codificação foi realizada a partir de fontes existentes, com a inclusão das rotinas necessárias, previstas no Modelo de Arquitetura de Segurança e é descrita com mais detalhes no item 5.4. A seguir serão apresentadas as estratégias de implementação de cada um dos serviços especificados para as Interfaces de Segurança.

5.2.1. Serviço de Integridade

O Serviço de Integridade, como já discutido, deve garantir que uma mensagem alterada não seja aceita e que uma mensagem antiga, se re-submetida, não seja processada. Para isso, na implementação realizada acrescentou-se informação às mensagens conforme descrito abaixo.

a) Não-alteração de mensagem:

Para garantir que uma mensagem chegue ao seu destino intacta (ou melhor, se esta mensagem for alterada no trajeto, esta alteração seja detectada) , optou-se por utilizar um *digest* MD5, de 16 *bytes*, cifrado com o algoritmo simétrico DES. O corpo original da mensagem não necessita alteração. A figura 5.2 apresenta uma mensagem qualquer alterada para incorporar este mecanismo.

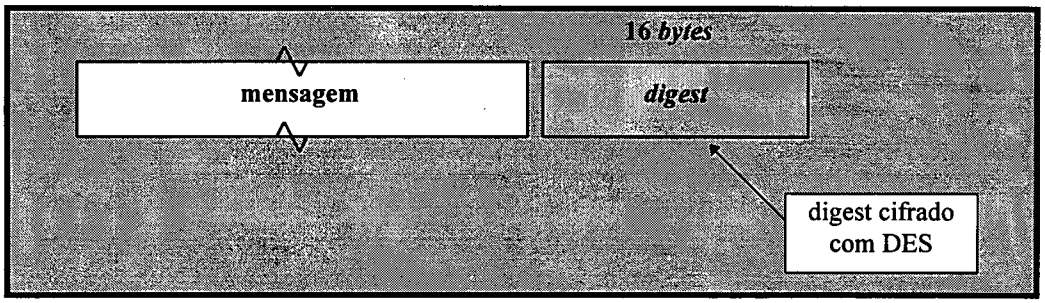

FIGURA 5.2: Mensagem com proteção contra alteração

b) Não re-submissão de mensagem

Da mesma forma, a incorporação do campo de sincronização com senha e próxima-senha se faz necessária para garantir que uma tentativa de re-submissão de uma mensagem antiga (válida) seja imediatamente detectada. A utilização de 4 *bytes* para cada um dos sub-campos (8 *bytes* de acréscimo, portanto) com cifragem também pelo algoritmo DES, e um algoritmo de geração de senhas baseada em PA (progressão aritmética) de razão 15, garante que só venha a ocorrer repetição da senha após 4 bilhões de mensagens em um sentido, entre cada par de interlocutores. O corpo original da mensagem não é alterado. A figura 5.3 apresenta o formato adotado.

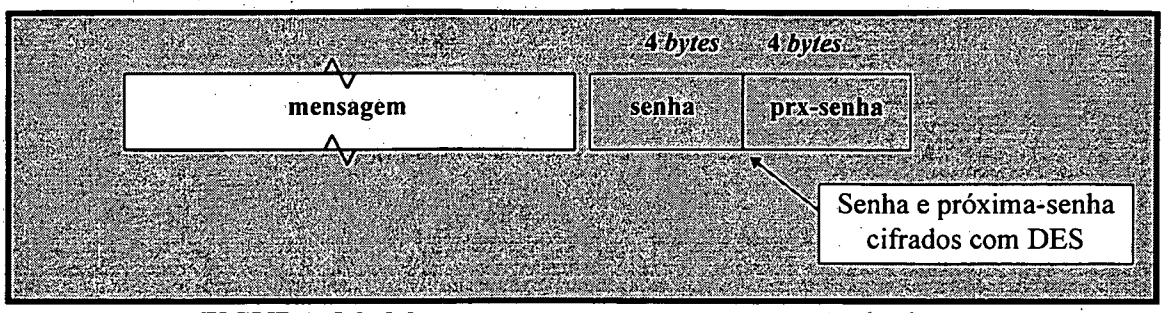

FIGURA 5.3: Mensagem com proteção contra re-submissão

5.2.2. Serviço de Autenticação

A autenticidade deve garantir que uma mensagem seja originária de um dos interlocutores confiáveis do Sistema de Gerência. A decisão de implementação fez valer a idéia de que a autenticidade pode ser conseguida como co-fator da integridade, utilizando os mesmos recursos, e reduzindo assim a sobrecarga total gerada pelos mecanismos de segurança. Utilizou-se, então, o *digest* MD5 de 16 *bytes*, cifrado com DES. A mensagem também manteve-se inalterada. Todas as entidades do sistema de gerência que interagem e, por isso, necessitam de uma chave para o algoritmo DES, foram consideradas confiáveis. Portanto, esse pressuposto deve ser obedecido estritamente para que a segurança (como um todo) e a autenticação (especificamente) sejam garantidas; uma vez que a mesma chave é utilizada por todas as entidades. A figura abaixo (5.4) apresenta o formato de uma mensagem com mecanismo de autenticação agregado.

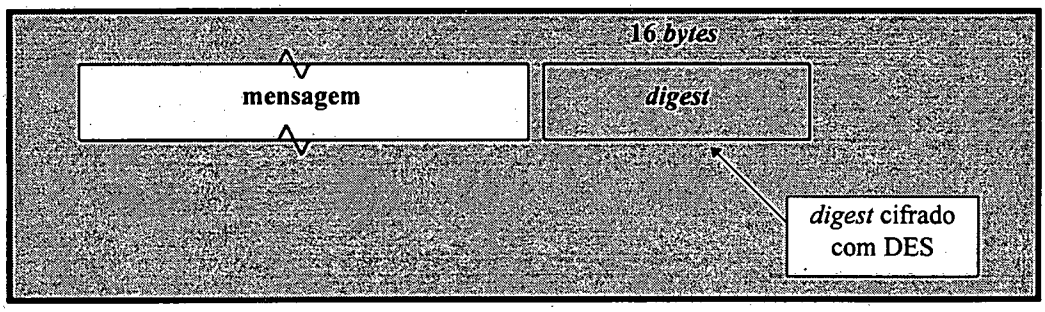

FIGURA 5.4: Mensagem com proteção quanto à autenticidade

5.2.3. Serviço de Confidencialidade

A confidencialidade evita que um interceptador de uma mensagem possa compreender o seu conteúdo. Para isto, basta cifrar a própria mensagem com um algoritmo de criptografia. O algoritmo mais eficiente, DES, foi escolhido para realizar esta tarefa. A figura

5.5 mostra que não é necessário agregar nada à mensagem para conseguir tal serviço, mas a mensagem já não é mais a original, pois passa pelo processo de criptografia.

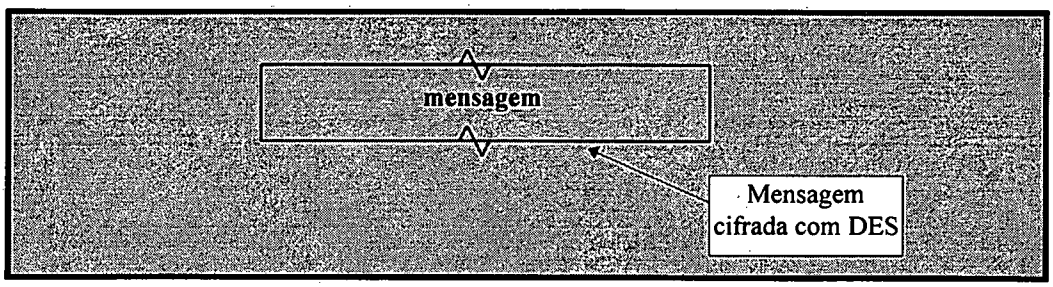

FIGURA 5.5: Mensagem com mecanismo de proteção quanto à privacidade

5.2.4. Reunindo os Serviços de Segurança

A união dos mecanismos descritos acima é que proverá a garantia de segurança para o Sistema de Gerência, como um todo. A figura 5.6 apresenta uma mensagem com todos os mecanismos de segurança agregados em uma mensagem.

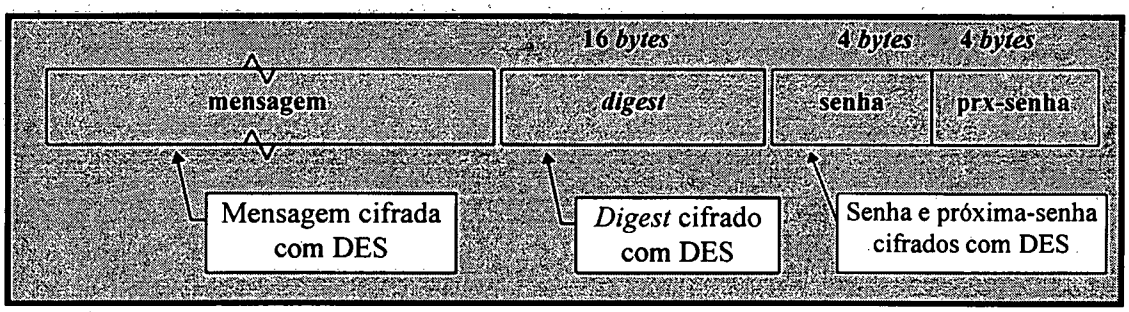

FIGURA 5.6: Mensagem com todos os mecanismos de proteção da Arquitetura

Os 24 *bytes* acrescidos à mensagem original representam uma sobrecarga para transmissão e também uma sobrecarga de processamento para a geração dos valores (de *digest* e senhas) e para a cifragem dos mesmos. O tempo de processamento para geração das senhas, considerando o método utilizado (PA de razão 15), é desprezível. O tempo de geração do *digest* já deve ser considerado, bem como o tempo processamento para a criptografia, tanto dos 24 *bytes* agregados quanto da própria mensagem. O tempo de cifragem do algoritmo DES é linear, tomando a entrada de 8 em 8 *bytes*; enquanto o algoritmo MD5 é linear baseado em múltiplos de 64 *bytes*. As equações 5.2 e 5.3 representam o tempo relativo necessário para

submeter a mensagem aos algoritmos DES e MD5, respectivamente, deduzidos a partir de análise do tamanho das mensagens e das características lineares dos algoritmos.

- Obtenção das equações
  - Seja **utd** a unidade básica de tempo para aplicação do algoritmo DES sobre 8 *bytes*. Para aplicação do algoritmo sobre o campo de *digest* de 16 bytes, são necessárias 2 **utd**, já que o algoritmo é linear. Para aplicação sobre os dois campos de senhas de 4 bytes cada, é necessária 1 **utd** . Seja **k** o tamanho da mensagem original que será transmitida, são necessários 1/8 (um oitavo) do tamanho da mensagem em unidades de tempo **utd**, ou seja, **k**/8 **utd**s para a cifragem com o algoritmo DES. Desta forma, para calcular o tempo total necessário para a cifragem de toda a mensagem com DES, basta adicionar o tempo necessário para cada uma das parcelas da mensagem final, conforme apresentado na equação 5.2.
  - Seja **utm** a unidade básica de tempo para aplicação do algoritmo MD5 sobre 64 *bytes*, uma mensagem de tamanho arbitrário **k** exige **k**/64 **utm**s para aplicação sobre toda a mensagem. Isto basta porque o algoritmo MD5 é aplicado somente sobre a mensagem original e gera o campo *digest* que é acrescentado na mensagem a ser enviada. Sendo assim, a equação 5.3 representa o tempo total requerido para aplicação do algoritmo MD5 na mensagem.

$$TT_{DES} = 3\,ut_{DES} + \frac{k}{8}\,ut_{DES} \qquad (2)$$

$$TT_{MD5} = \frac{k}{64}\,ut_{MD5} \qquad (3)$$

Onde: TT_{XXX} é o Tempo Total para aplicação do algoritmo XXX
ut_{XXX} é 1 unidade de tempo utilizada para aplicação do algoritmo XXX
k é o tamanho da mensagem original

EQUAÇÕES 5.2 e 5.3: Tempo Relativo para incorporação de mecanismos de segurança

As equações 5.2 e 5.3 tem comportamento simples, e podem ser descritos pelos gráficos abaixo (figuras 5.7a e 5.7b), aplicado o esquema descrito neste item (24 bytes de redundância para segurança), e com tamanho da mensagem variando. Pode-se perceber, no

gráfico referente à inclusão do DES, que a equação é linear. No caso da aplicação do MD5, o comportamento da curva é composto de patamares, sempre maiores, na medida em que a mensagem cresce.

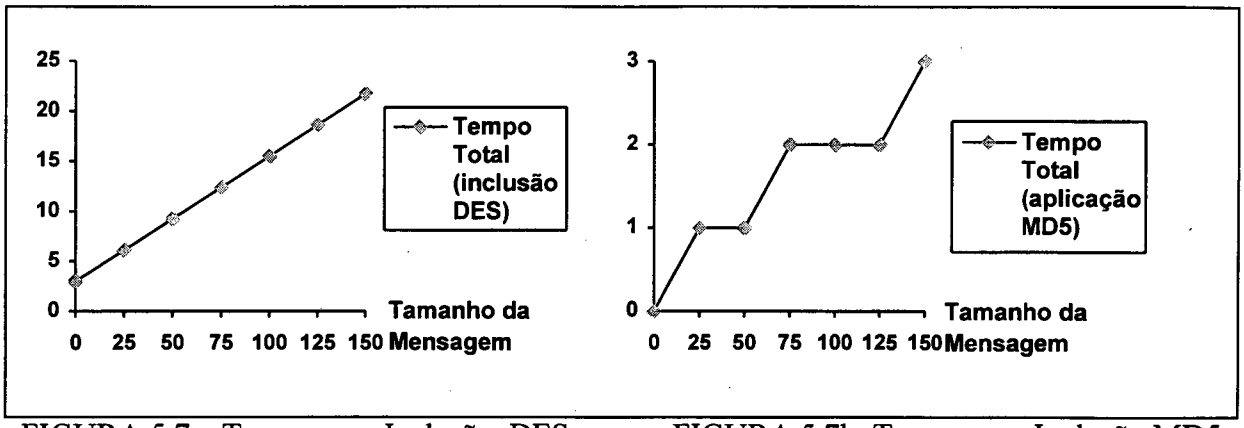

FIGURA 5.7a: Tempo para Inclusão: DES FIGURA 5.7b: Tempo para Inclusão MD5

5.3. Algoritmos da Interface de Segurança

Os algoritmos utilizados para a implementação das Interfaces de Segurança são divididos em Algoritmo para Envio de Mensagens Seguras, Algoritmo para Recepção de Mensagens Seguras e Algoritmo de Confidencialidade de Acesso à MIB. Estes algoritmos são apresentados a seguir.

5.3.1. Algoritmos para Envio e Recepção de Mensagens Seguras

Para se ter a garantia de **confidencialidade**, **autenticidade** e **integridade** das mensagens que são trocadas entre as entidades componentes do Sistema de Gerência, cada Interface de Segurança implementa (indistintamente para agentes e gerentes) os algoritmos para o envio e o recebimento de mensagens a seguir (as figuras 5.8 e 5.9 apresentam os diagramas de transição de estados dos algoritmos).

- Envio de Mensagens Seguras:

1. Recepção de mensagem da entidade "pura"[3] destinada a uma entidade par;

---

[3] *Por entidade "pura" se entende agente ou gerente do Sistema de Gerência sem os mecanismos de segurança aqui descritos, ou seja, conforme originalmente implementados. Toda comunicação entre entidades puras agora se dá através da Interface de Segurança apresentada.*

2. Criptografar toda a mensagem original, gerando uma Mensagem Cifrada;
3. Calcular o *digest* da Mensagem Cifrada;
4. Cifrar o *digest* calculado com a chave secreta comum, gerando o *Digest* Cifrado;
5. Gerar o sub-campo Próxima-Senha e agregar ao sub-campo Senha, gerando o Campo de Sincronismo;
6. Cifrar o Campo de Sincronismo com a chave secreta comum;
7. Montar a Mensagem Segura, com a agregação da Mensagem Cifrada, do *Digest* Cifrado e do Campo de Sincronismo Cifrado;
8. Enviar a Mensagem Segura para a Interface de Segurança homóloga.

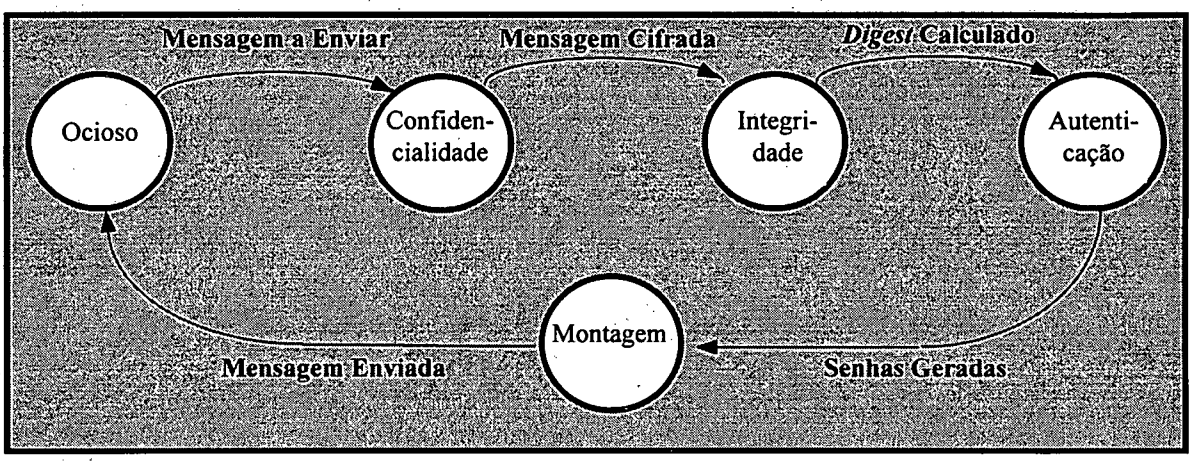

FIGURA 5.8: Envio de Mensagens Seguras

- Recepção de Mensagens Seguras:

1. Recepção de mensagem da Interface de Segurança homóloga;
2. Decifrar Campo de Sincronismo Cifrado, usando a chave secreta comum;
3. Verificar sub-campo Senha conforme negociação realizada durante interação anterior;
4. Decifrar *Digest* Cifrado;
5. Calcular *Digest* da Mensagem Cifrada;
6. Verificar *Digest* Decifrado com *Digest* Calculado;
7. Decifrar Mensagem Cifrada com chave secreta comum;
8. Enviar Mensagem Decifrada para a entidade "pura" associada.

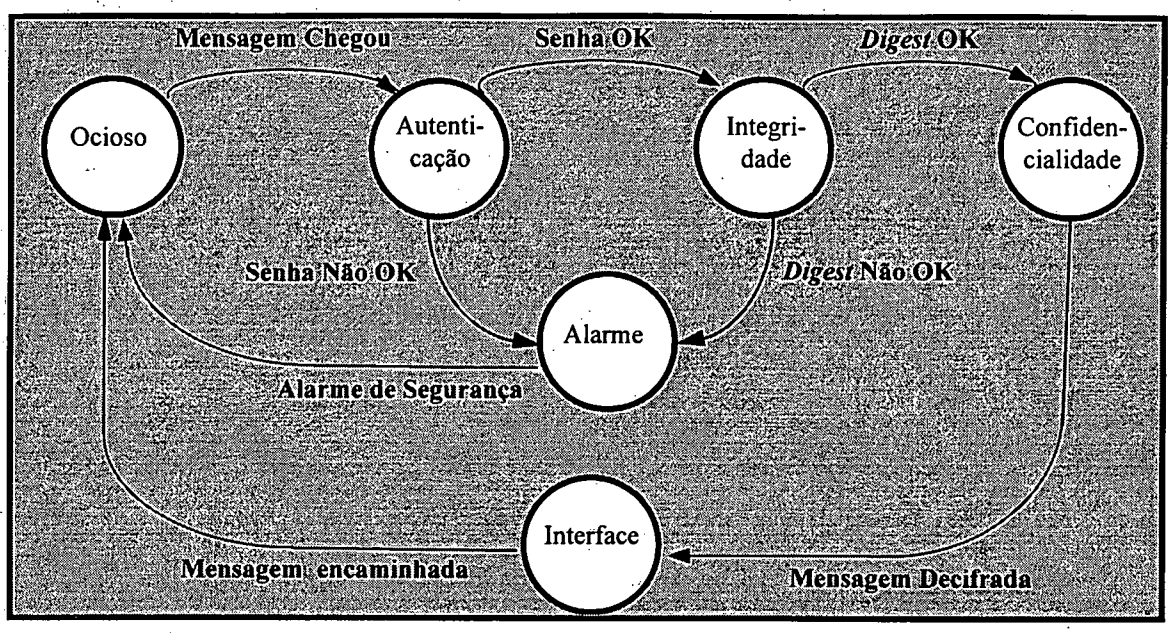

FIGURA 5.9: Recepção de Mensagens Seguras

5.3.2. Algoritmo para Confidencialidade de Acesso à MIB

Conforme apresentado anteriormente, há dois momentos onde é necessária a confidencialidade: na comunicação (acima) e no acesso à MIB. Se a MIB reside em disco, é importante que ela esteja garantida contra acessos indevidos. Para isto, a Interface de Segurança dos Agentes implementa os seguintes algoritmos, que também são ilustrados na figura 5.10:

- Manutenção de Informação (Operação de Escrita):

1. Recebe solicitação de manutenção de informação (alteração/inclusão) da Entidade "pura";
2. Criptografa informação com chave secreta;
3. Armazena/Altera informação na MIB.

- Acesso à Informação da MIB (Operação de Leitura):

1. Recebe solicitação de acesso à MIB da Entidade "pura";
2. Busca na MIB a informação desejada, que estará criptografada;
3. Decifra a informação, com a chave secreta;
4. Envia informação para Entidade "pura".

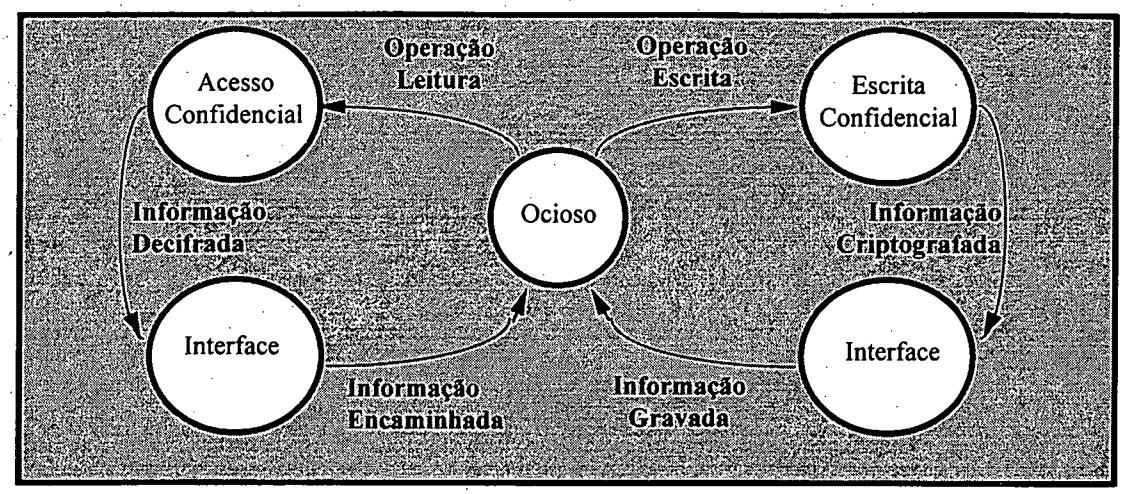

FIGURA 5.10: Controle de Acesso à MIB

5.4. Ambiente de Teste

A implementação que serviu aos testes de verificação da viabilidade do modelo proposto foi realizada em ambiente de gerência SunNet Manager v1.1 [SUN 91], instalado na rede do Departamento de Informática e de Estatística (INE) da Universidade Federal de Santa Catarina (UFSC) - Rede INF, que é composta por estações de trabalho Sun Microsystems, de modelos variados, além de microcomputadores tipo PC e diversos periféricos como impressoras e discos. O SunNet Manager (SNM) define um ambiente de desenvolvimento que permite a criação de agentes e gerentes que interagem.

A comunicação entre as entidades de gerência dá-se por meio de RPC's (*Remote Procedure Calls* - chamada de procedimentos remotos), que é uma forma disciplinada e padronizada de IPC (*InterProcesses Communication* - comunicação entre processos), disponível em ambientes Unix [SUN 90a]. A plataforma de software padrão da Rede INF é o SunOS 4.1.3 (4.3 BSD Unix), sistema operacional multitarefa de propriedade da Sun Microsystems [SUN 90b], que implementa RPC segundo o RFC 1050.

O código fonte de entidades de gerência estava disponível e, por isso, rotinas que implementam os diversos serviços que compõem o modelo foram incorporadas ao software original. O código original utilizado compõe o Sistema de Alertas, desenvolvido por Ewerton Longoni Madruga, em seu trabalho de mestrado [MAD 94], realizado no Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Rio Grande do Sul, sob orientação da Profa. Dra. Liane Margarida R. Tarouco. As interações entre as entidades do

Sistema de Alertas foram encapsuladas pelos serviços de segurança. Estas rotinas incorporadas implementam os algoritmos de Envio e de Recepção de Mensagens Seguras e de Confidencialidade de Acesso à MIB (quando armazenada em disco), conforme descritos na seção 5.3; e que utilizam as funções de criptografia DES e *digest* MD5, conforme apresentado na seção 5.2 e descritos no Anexo A.

O Sistema de Alertas é uma ferramenta de auxílio à administração de rede nas áreas de Gerência de Falhas e de Desempenho. Sua tarefa é coletar e analisar dados sobre equipamentos da rede e gerar eventos que, segundo um conjunto de regras, podem levar à geração de alertas que informam ao operador que atitudes devem ser tomadas para que o nível dos serviços se mantenham. A interação entre os módulos que compõem o Sistema de Alertas é apresentada na figura 5.11. O Sistema de Alertas fornece ao SNM uma função que deve ser evocada automaticamente a cada amostra (relatório de dados) ou *trap* recebidos (módulo "Analisador de Amostras"). As interações entre as entidades de gerência que se valem da rede foram encapsuladas por serviços de segurança, garantindo sua autenticidade, integridade e confidencialidade. Isto significa que, antes de mensagens de gerência circularem na rede, elas são submetidas aos serviços necessários e então enviadas. Do outro lado, antes de ingressarem no sistema, as mensagens são tratadas e validadas pelas rotinas competentes que implementam os complementares dos serviços incorporados.

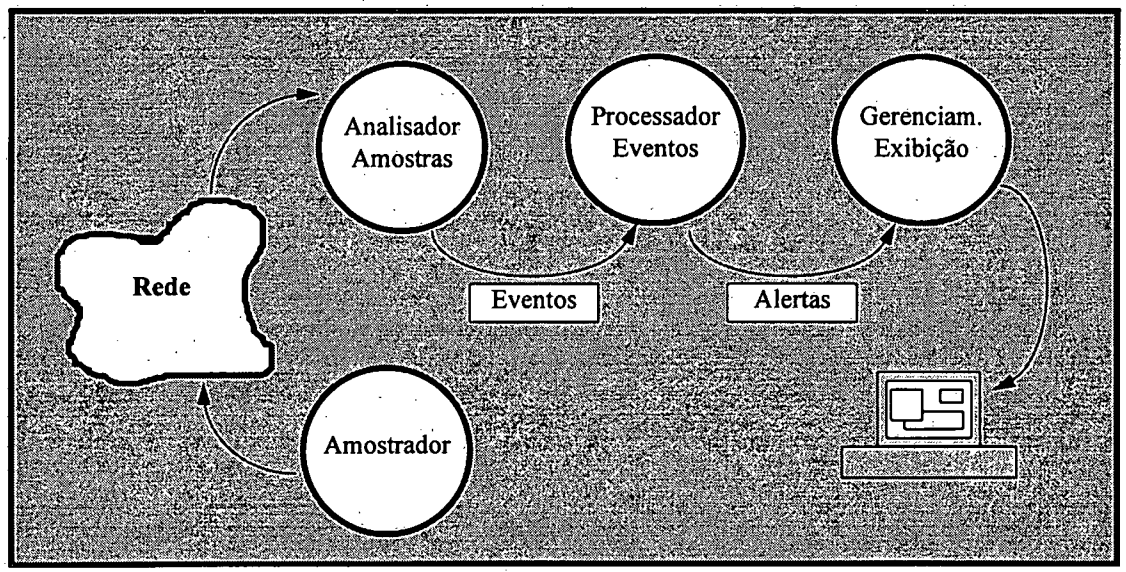

FIGURA 5.11: Organização dos módulos do Sistema de Alertas [MAD 94]

A figura 5.12 ilustra um agente padrão com a incorporação das rotinas necessárias para a agregação de segurança no envio de um relatório de dados, através do *dispatch_request* (rotina padrão de tratamento de requisições de dados). O agente utilizado no exemplo, chamado *etherif*, está incluído originalmente no SunNet Manager. A figura 5.13, por sua vez, apresenta um trecho do código do Sistema de Alertas que trata os dados recebidos (rotina *report_handler*, também padrão em gerentes SNM), com as rotinas de validação, que são utilizadas quando da chegada de um relatório de dados ou evento[4].

```
typedef struct {  caddr_t valorOriginal;
                  char[16] digest;
                  char[4]  senha;
                  char[4]  pxsenha
                } valorSeguro;

typedef struct {  char *name;
                  u_int type;
                  u_int length
                } semi_Netmgt_data;

static void putargs (name, type, size, value)
{
struct valorSeguro *Seguro;
struct semi_Netmgt_data q;
      ...
      DefineSenhaDES ( SENHA );      /* SENHA secreta comum */        /*1*/
      strcpy ( q.name, name );
      q.type = type;
      q.length = size;
      Cifrar ( q );                  /* resultado no proprio q */     /*2*/
      Seguro->valorOriginal = value;
      Seguro->digest        = MD5 ( q );   /*calcula MD5 */          /*3*/
      Seguro->senha         = SenhaAtual;  /*valor global*/          /*4*/
      SenhaAtual            = prox_senha ( minha_anterior );        /*5*/
                                 /*gera proxima senha*/
      Seguro->pxsenha       = SenhaAtual;                           /*6*/
      Cifrar ( Seguro );                                            /*7*/
      strcpy ( p->name, q.name );
      p->type   = q.type;
      p->length = q.length;
      p->value  = Seguro;

      /* Enviando um report */
      netmgt_build_report ( p, &event );
      ...
```

FIGURA 5.12: Código de Envio de Mensagem Segura

[4] *O código foi simplificado para melhor compreensão. Nas duas figuras (5.12 e 5.13), o código original aparece em itálico. O código relativo à segurança aparece de forma normal.*

Seguindo os algoritmos da seção 5.3, pode-se analisar o código da figura 5.12 e verificar a seqüência da aplicação dos mecanismos: criptografia para garantir confidencialidade (referências 1, 2 e 7), *digest* para garantir autenticidade e não-alteração (referência 3) e senhas para garantir não re-submissão (referências 4, 5 e 6). Também na figura 5.12 são apresentadas duas estruturas definidas pela Interface de Segurança para permitir a realição das operações: valorSeguro e semi_Netmgt_data.

A figura 5.13 também segue os algoritmos apresentados na seção 5.3, e pode-se distinguir as operações de descriptografar (referências 1, 2 e 6) para restaurar os valores originais, verificação do *digest* (referência 3) e verificação e atualização das senhas (referências 4 e 6).

```
struct valorSeguro *Seguro;
struct semi_Netmgt_data q;
     . . .
     netmgt_fetch_data(&data)     /* recebe relatorio dados do agente */
     DefineSenhaDES ( SENHA );                                    /*1*/
     strcpy ( q.name, data.name );
     q.type   = data.type;
     q.length = data.length;
     Seguro   = data.value;
     Decifrar ( Seguro );                                         /*2*/
     if ((Seguro.digest != MD5(q))||                              /*3*/
         (Seguro.senha!=SenhaAtual)) {                            /*4*/
           fprintf ( stderr, "Erro na validacao de seguranca");
           unregister_application();
           exit(0);     /* poderia levantar alarme de seguranca */
     }
     SenhaAtual = Seguro.pxsenha;                                 /*5*/
     Decifrar ( q );                                              /*6*/
     data.type = q.type;
     data.length = q.length;
     strcpy ( q.name, data.name );
     data.value = Seguro.valorOriginal;
     ...
```

FIGURA 5.13: Código de Recepção de Mensagem Segura

Desta forma, percebe-se que a abordagem utilizada para incorporar a segurança nestas rotinas foi a manipulação dos dados antes que estes fossem submetidos às primitivas básicas de emissão de mensagens do SunNet Manager (*netmgt_build_report*, por exemplo).

Também no caso da recepção de mensagens, assim que as mesmas são recebidas pelas primitivas do SunNet Manager (no exemplo, *netmgt_fetch_data*), passam por rotinas de validação como as apresentadas na figura 5.13, para confirmar a legitimidade da mensagem.

5.5 Avaliação

Sobre a implementação realizada, foram levantados dados referentes à sobrecarga de tempo que foi acrescida ao sistema como um todo pela incorporação dos serviços de segurança. Ressalte-se, desde o princípio, que tal medição serve exclusivamente para demonstração da carga incluída pela implementação piloto, não devendo ser considerada como referência para nenhuma outra implementação. Isto se deve ao fato de a implementação realizada ter como principal objetivo a demonstração de viabilidade da Arquitetura de Segurança tal como proposta e não sua eficiência na realização das tarefas. Portanto, o código não foi submetido a sessões de depuração/otimização e não tem como característica a eficiência em termos de tempo de processamento. Isto se aplica tanto ao código referente ao algoritmo de criptografia DES quanto ao algoritmo de *digest* MD5. Ambos foram submetidos, sim, a testes de conformidade, onde implementações distintas eram utilizadas sobre os mesmos dados para verificar se geravam as mesmas respostas.

Para a realização das avaliações, o algoritmo DES foi utilizado de dois modos, seguindo os modos padronizados ECB (*Electronic Code Book*) e CBC (*Cipher Block Chaining*) [TAN 91].

Já o algoritmo MD5 não permite variações quanto a forma de implementação, tendo sido então implementado conforme a especificação básica e confrontado com resultados esperados também apresentados no seu documento de padronização [RIV 92]. Da mesma forma, o MD5 trabalha sempre com dados de dimensão múltipla de 64, tornando seu desempenho diretamente relacionado ao tamanho da mensagem módulo 64, como já pôde ser observado na figura 5.7b.

A metodologia utilizada no levantamento dos dados foi a seguinte:

1. Um par agente-gerente sem mecanismos de segurança foi acompanhado, sendo tomados os tempos[5] de cada um, referentes a:
   a) agente: tempo entre a chegada de uma requisição e o envio de uma resposta
   b) gerente: tempo entre o despacho de uma requisição e o recebimento de uma resposta
2. Foram tomadas estas amostras em dias típicos, em momentos variados, onde a rede INF apresenta diferentes cargas
3. O mesmo par agente-gerente, agora com mecanismos de segurança incorporados foram submetidos às mesmas amostragens em momentos equivalentes. Como a Arquitetura de Segurança é flexível neste ponto, realizou-se a análise com duas configurações distintas:
   a) Segurança Máxima: 24 *bytes* redundantes de segurança, sendo 16 *bytes* de *digest* MD5, e 8 *bytes* de campo de sincronização (4 + 4 *bytes*).
   b) Segurança Mínima: 4 *bytes* redundantes, sendo 2 do *digest* CRC-16 e 2 de sincronização.
4. Fez-se um comparativo entre os tempos médios das quatro variantes de segurança implementadas (DES-ECB com Segurança Máxima, DES-ECB com Segurança Mínima, CBC com Segurança Máxima e CBC com Segurança Mínima) com o tempo médio das amostras tomadas no sistema que não dispunha de mecanismos de segurança.

Configuração de Segurança	DES ECB	DES CBC
Segurança Mínima	+7,0%	+8,1%
Segurança Máxima	+14.2%	+15.5%

TABELA 5.2: Diferenças sobre o Sistema de Gerência sem segurança

Tomou-se o cuidado de somar os tempos das atividades dos gerentes e agentes que eram relacionadas, de modo a considerar uma operação de gerência uma atividade indivisível, pois, apesar de ser realizada de maneira distribuída sobre uma rede, uma operação que envolve requisição e resposta é una e coesa. Então, a média destes tempos somados é que foi usada como parâmetro de comparação.

---

[5] *Em todos os casos foram utilizados os tempos de CPU e não o tempo conforme parece ao usuário, uma vez que a máquina do usuário pode estar sobrecarregada com outras tarefas em alguns momentos de medição e não em outros, o que prejudicaria a análise.*

Também procurou-se evitar a interpretação incorreta dos dados em função de situações atípicas/anormais que porventura pudessem ter ocorrido, gerando valores espúrios. Para tanto, levantou-se a moda dos tempos somados, o que se mostrou desnecessário pois a análise comparativa entre as modas obtidas trouxe um resultado (percentual de diferença) muito próximo do que as médias apresentaram, ratificando assim os dados.

A tabela 5.2 apresenta o resultado das comparações segundo a equação abaixo:

$$p = \frac{100 * CS}{SS} + SS$$

Onde: p = percentual (%) de diferença

CS = média das amostras Com Segurança

SS = média das amostras Sem Segurança

EQUAÇÃO 5.4: Percentual de diferença entre sistema com e sem segurança

Permite, desta forma, calcular o percentual da diferença entre o sistema sem segurança e com as diferentes configurações de segurança.

A análise da tabela 5.2, dá conta de que:

a) a diferença entre os 2 modos de criptografia DES utilizados (ECB e CBC) é significativa. O método ECB é o mais simples de todos (como se pode perceber da sua descrição e implementação) e, por não incorporar nenhum encadeamento ou realimentação do algoritmo (como fazem os outros modos), é também o mais frágil deles. A vantagem da segurança trazida pelo modo CBC está diretamente representado pelo tempo a mais que é necessário para sua aplicação.
b) o uso do MD5 tanto aumenta o tamanho da mensagem a ser transmitida como também o tempo de cálculo (em relação ao CRC-16, usado na linha "Segurança Mínima"). A análise custoXbenefício desta configuração é mais adequada do que a configuração de segurança mínima pois a segurança apresentada tem crescimento exponencial no que se refere ao número de bytes do campo de sincronização (tabela 5.1, coluna b) e o acréscimo de tempo

é linear. Apesar de a parte de segurança da mensagem ter crescido 6 vezes (de 4 para 24 *bytes*), o tempo necessário para uso da arquitetura cresceu apenas 2 vezes.

Desta forma, pode-se perceber que, mesmo não havendo sido planejada para eficiência, pode-se perceber as alternativas disponíveis e visualiza a mais adequada para as situações particulares.

5.6 Conclusão

As principais características detectadas no modelo são a simplicidade e flexibilidade da Arquitetura, ao mesmo tempo demonstrando capacidade de atender aos requisitos de segurança para Gerência de Redes. A implementação realizada se baseou nestas características, permitindo a identificação e compreensão dos fatores envolvidos para uma análise objetiva dos mesmos.

As decisões de implementação mostraram-se corretas no sentido que, se de um lado não comprometeram a eficiência do sistema como um todo, de outro inseriram a segurança pretendida, permitindo um rígido controle sobre todo o sistema sem afetar a usabilidade do mesmo.

6 Conclusões

Os aspectos relacionados com Segurança em Redes de Computadores e, em especial, em Segurança da Gerência de Redes foram apresentados com detalhes, objetivando demonstrar que o estudo destes temas são importantes no mundo de hoje e que exigem um significativo esforço em comum da comunidade de redes contra as ameaças que estão presentes diariamente nos sistemas. Também pretende demonstrar que a Gerência de Redes é um dos pontos frágeis no que concerne a segurança. Esta foi a motivação do trabalho.

Foi apresentada, então, uma Arquitetura de Segurança genérica para aplicação em Sistemas de Gerência de Redes. Tal arquitetura é extremamente flexível, uma vez que permite sua aplicação em Sistemas de Gerência que usam agentes e gerentes e, ao mesmo tempo, não exige que tais entidades sofram alterações para suportá-la, tornando transparente a sua instalação. Também permite que os Serviços de Segurança disponibilizados por ela sejam seletivamente implantados, conforme necessidades de cada instalação e restrições de desempenho.

Apesar de possuir uma estrutura simples e não prever a inclusão de grande sobrecarga ao sistema como um todo, são oferecidos pela Arquitetura os Serviços de Autenticação, Integridade e Confidencialidade (de comunicação e de acesso a dados). As novidades da abordagem utilizada dizem respeito à utilização de seqüenciamento constante (a cada mensagem trocada entre entidades é realizada uma verificação da origem - autenticação), a ampliação do conceito de assinatura digital (que permite não só validar a origem mas também a integridade das mensagens) e ao controle de acesso à MIB ser feito através do serviço de confidencialidade (que não impede o acesso mas não revela as informações senão para o legítimo proprietário).

Além disso, a Arquitetura é genérica, podendo ser aplicada para segurança em redes em geral, e não somente para sistemas de gerência de redes. Qualquer aplicação pode se valer do modelo proposto para implementar uma arquitetura que responda às necessidades

de segurança específicas, bastando analisar os requisitos de segurança da aplicação em vista e implementar os serviços adequados junto à Interface de Segurança. Isto porque a idéia de uma interface que filtra as mensagens pode ser utilizada sem restrições, apenas adaptando às necessidades específicas.

A Arquitetura permite criar um ambiente confiável para Sistemas de Gerência de Redes; mas como tudo relacionado com segurança[1], somente após muitos testes e análises e a submissão da implementação a sistemas em produção, vulneráveis a ataques de toda ordem, é que será possível a sua validação. Além disso, a validação estará fortemente associada com os algoritmos de criptografia utilizados pelos diversos serviços integrantes da Arquitetura; muito mais do que em relação ao modelo proposto. Testes da implementação em ambiente com carga regular de tráfego de gerência em uma instalação em produção e sujeita a ameaças reais é que podem, então, determinar com precisão a eficácia dos algoritmos e a sobrecarga real causada pela implementação.

Outras variações também podem ser verificadas como a utilização de mecanismos de criptografia assimétrica (como o RSA) e a implementação da Interface de Segurança de modo autônomo, como uma pseudo-camada entre o Sistema de Gerência e as camadas de rede que provêem serviços de comunicação. Estas variações, na verdade, estão todas contempladas no modelo proposto sendo, então, questão de decisão de projeto a opção por uma ou outra. Este é um dos pontos importantes do trabalho e sobre o qual mais se trabalhou: a sua adequabilidade à diversas filosofias de comunicação entre processos e implementações distintas entre sistemas de gerência. Tudo o que poderia ser caracterizado como idiossincrasia ou restrição foi removido do modelo, sempre optando-se pelas soluções mais gerais que atendiam a diferentes tipos de algoritmos de criptografia (não apenas diferentes algoritmos, mas tipos).

---

[1] *Algoritmos de criptografia e segurança em geral nunca são considerados totalmente confiáveis. O máximo que se afirma acerca de um mecanismo de segurança é que "até aquele momento" não existem ou não foram publicadas formas de se sobrepujar os mesmos [PFL 89] [DOD 83].*

Deve-se ressaltar ainda o fator custo no resultado final, porque é sabido que, se mal dimensionado, o tráfego de gerência em redes pode sobrecarregar por si só a própria rede. Com a inclusão de mecanismos de segurança, que inserem tanto tempo de processamento quanto (e principalmente) bytes nas mensagens de gerência, a sobrecarga é ainda maior. Disso decorre então a necessidade de um bom balanceamento entre os benefícios requeridos e as dificuldades inseridas.

A extensão do modelo proposto para suportar outras formas de implementação de entidades de gerência, como por exemplo agentes *proxy*, e mesmo equipamentos de comunicação gerenciáveis com capacidade de processamento limitada ou restrita, e que permitem pouca ou nenhuma interferência nos seus agentes, como *hubs*, é uma possível continuação deste trabalho, que tornará o Modelo ainda mais consistente.

A seqüência natural deste trabalho passa, inevitavelmente, pela implementação da Arquitetura em ambientes de gerência distintos, inclusive dentro do domínio OSI. Também é necessária a evolução das análises e testes para ambientes mais complexos, envolvendo diversas redes e sub-redes com sistemas de gerência distintos convivendo e interagindo com eles. Ainda mais ousado seria uma iniciativa no sentido de expandir a abrangência da Arquitetura de Segurança, atingindo as comunicações entre entidades da rede, trazendo segurança não apenas para aplicações específicas (como os sistemas de gerência) mas generalizando-a e suportando todas as interações entre processos que utilizem redes.

Anexo A

Algoritmos de Criptografia

Neste anexo são apresentados com alguns detalhes os algoritmos de criptografia mais comumente utilizados atualmente, DES e RSA, bem como o algoritmo de *digest*, MD5.

A.1. Algoritmo DES - Data Encryption Standard

Um dos sistemas de criptografia simétrica mais amplamente utilizado hoje em dia é o *Data Encryption Standard* (DES - Padrão de Criptografia de Dados), foi desenvolvido nos anos 70 e patenteado pela IBM, que posteriormente o tornou disponível para uso público. Em 1977 foi publicada a primeira norma técnica descrevendo o algoritmo e o *status* de padrão ANSI foi adquirido em 1981 (X 3.92-1981/R 1987).

DES é um algoritmo baseado em funções de permutação, substituição e recombinação, a nível de *bits*, de blocos de 64 *bits* com chave de 56 *bits* (8 caracteres ASCII - 7 *bits*). O algoritmo é estruturado de forma que, se qualquer bit da entrada for alterado, diversos *bits* da saída serão afetados.

O método DES é considerado robusto o suficiente para prover segurança às mais variadas aplicações; não sendo facilmente suscetível a ataques do tipo força-bruta (tentativa de suposição da chave); pois ou o custo e/ou o tempo necessários para concretizar tal ataque é muito elevado.

Existem quatro modos padronizados para emprego do algoritmo básico [GAR 92] [TAN 91]:

- ECB - *Electronic Code Book* (Livro de Código Eletrônico): neste modo, cada bloco de entrada (64 *bits*) é cifrado usando a chave (56 *bits*), e a saída é escrita como um bloco. Este método é a cifragem simples de uma mensagem, um bloco por vez.

- CBC - *Cipher Block Chaining* (Encadeamento de Blocos Cifrados): o texto original é submetido a uma operação lógica binária XOR com o valor cifrado do bloco prévio. Um valor conhecido é usado para o primeiro bloco. O resultado é então cifrado usando a chave. O último bloco pode ser usado como *checksum* para garantir que o texto cifrado não foi alterado.

- CFB - *Cipher FeedBack* (Realimentação Cifrada): a saída é re-inserida no mecanismo. Após cada bloco ser cifrado, parte dele é deslocado para um registrador. O conteúdo deste registrador é cifrado com a chave do usuário usando o modo ECB, e é feito um XOR desta saída com a cadeia de dados para produzir o resultado.

- OFB - *Output FeedBack* (Realimentação da Saída): a saída também é re-inserida no mecanismo. Um registrador é cifrado pelo modo ECB usando a chave do usuário. O resultado é usado como uma chave para cifrar os blocos de dados (usando XOR) e é armazenado também no registrador para ser usado com o próximo bloco.

O algoritmo DES é uma combinação cuidadosa e complexa de dois dos blocos básicos construtores da criptografia: substituição e permutação (ou transposição). O algoritmo deriva sua robustez da aplicação repetida destas duas técnicas, uma sobre a outra, em dezesseis ciclos sucessivos. Tais características trazem grande complexidade para o rastreamento de um *bit* desde a entrada até a saída, pois este passa por 16 ciclos de substituição e transposição.

As substituições provêem confusão pela substituição sistemática de alguns padrões de *bits* por outros. As transposições promovem a difusão pela reordenação dos *bits*. Os textos planos são, portanto, afetados por uma série de ciclos de substituição e transposição. As iterações de substituições e permutações são realizadas como delineadas na figura A.1.

O algoritmo usa somente operações aritméticas e lógicas padronizadas sobre números de 64 *bits*, tornando-o adequado tanto para implementação em software quanto em

hardware, estando diversas implementações e *chips* disponíveis para a criação de dispositivos e mecanismos de segurança.

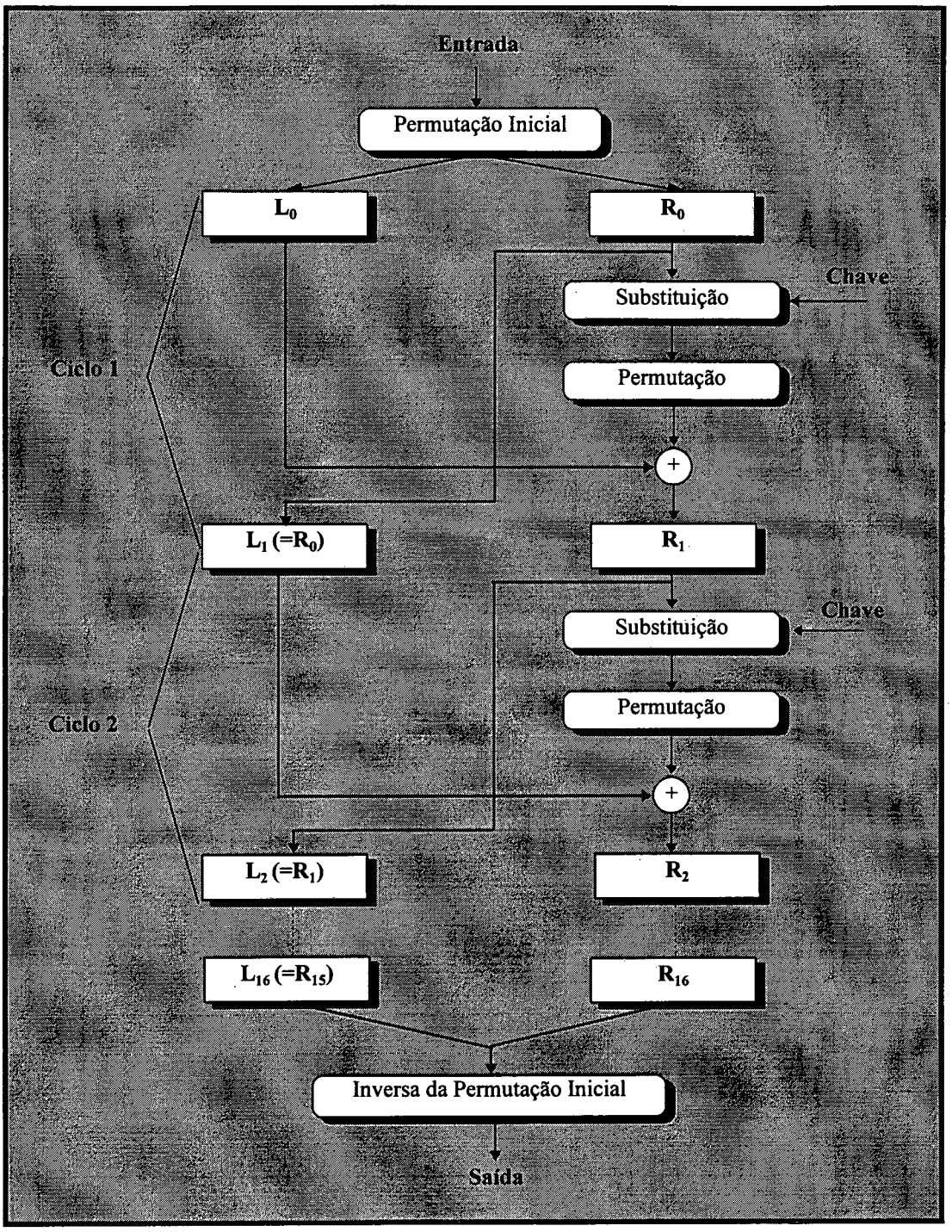

Figura A.1: Ciclos do algoritmo DES

A.2. Algoritmo RSA - *Rivest, Shamir e Adleman*

RSA também é um sistema de criptografia muito difundido, mas é da modalidade assimétrica, também conhecida como criptografia de chave pública. RSA são as iniciais dos autores do método: R.L.Rivest, A.Shamir e L.Adleman. Diferentemente de sistemas de chaves secretas, criptografia por chave pública usa duas chaves: uma pública e uma privada. A chave pública é usada para cifrar uma mensagem e a privada para decifrá-la. O inverso também é possível. RSA é patenteado nos EUA e não está disponível para uso sem licença.

A robustez do método RSA está baseada na dificuldade de fatorar números muito grandes. Propriedades de aritmética inteira são usadas para conseguir as características desejadas do algoritmo. A fatoração de grandes números é tarefa trabalhosa e aumenta-se a robustez do método quanto maior for o tamanho das chaves que serão utilizadas.

Uma chave típica para ser empregada com RSA pode conter 200 dígitos decimais [GAR 92]. Todos os valores de 200 dígitos podem ser representados por 665 *bits* (2^X possui aproximadamente $X(\log_{10} 2) + 1$ dígitos decimais). Para fatorar tal número, seriam necessários aproximadamente 1.2×10^{23} operações. Se uma máquina é capaz de realizar 10^{10} operações por segundo (mais do que as melhores máquinas de hoje), seriam necessários 1.2×10^{13} segundos (ou 380.267 anos) de tempo de computação. Dobrando o tamanho da chave para 400 dígitos, requerer-se-ia 8.6×10^{15} anos para fatorá-la (para efeito de comparação, Stephen Hawking em "Uma Breve História do Tempo" estima que o universo tenha apenas 2×10^{10} anos [HAW 90]). Então, a menos que se descubra um caminho através da teoria matemática, o mecanismo RSA pode ser considerado seguro nos dias de hoje.

Com o algoritmo RSA, existem duas chaves, *d* e *e*, que trabalham em pares para decifrar e cifrar, respectivamente. Um texto plano[1] *P* é criptografada para um texto cifrado[2] pela função:

$$C = P^e \bmod n \qquad (A.1)$$

O texto plano é recuperado por:

$$P = C^d \bmod n \qquad (A.2)$$

Em função da simetria na aritmética modular, a encriptação e a decriptação são mutuamente inversas e comutativas. Portanto, de A.1 e A.2, deriva-se:

$$P = C^d \bmod n = (P^e)^d \bmod n = (P^d)^e \bmod n \qquad (A.3)$$

Isto significa que pode ser aplicada a transformação de criptografia e então a de descriptografia, ou a de descriptografia seguida pela de criptografia.

A chave de criptografia consiste de um par de inteiros (*e*, *n*), e a chave de descriptografia é (*d*, *n*). O ponto de partida para encontrar chaves para este algoritmo é selecionar um valor para *n*. O valor de *n* deve ser suficientemente grande, um produto de dois primos *p* e *q*. Ambos devem ser grandes também. Normalmente, *p* e *q* possuem aproximadamente 100 dígitos cada, de modo que *n* tem um tamanho de aproximadamente 200 dígitos, o inibe a fatoração de *n* e a inferência de *p* e *q*.

A seguir, um inteiro relativamente grande ser escolhido para *e* de modo que seja relativamente primo[3] a (p-1)*(q-1). Um modo fácil de garantir tal característica é escolher *e* como um número primo que é maior do que (p-1) e do que (q-1).

---

[1] *Texto Plano (Plaintext): Texto original, não submetido a qualquer processo de criptografia [PFL 89].*

[2] *Texto Cifrado (Ciphertext): Texto criptografado, gerado a partir da submissão de um plaintext a um processo de criptografia [PFL 89].*

[3] *Relativamente primo significa que* e *não têm fatores em comum com (p-1)*(q-1) [PFL 89]*

Finalmente, seleciona-se *d* tal que:

$$e * d \equiv 1 \bmod (p-1)*(q-1) \qquad (A.4)$$

Assim, obtém-se os valores que serão utilizados como chaves para cifrar e decifrar textos. Deve-se observar também que qualquer uma delas pode ser utilizada como pública ou privada, uma vez que são intercambiáveis, apresentando as mesmas características.

A.3. Algoritmo MD5 - Message Digest 5

O algoritmo toma como entrada uma mensagem de tamanho arbitrário e produz como saída um *digest* (ou *fingerprint*) de 128 *bits* da entrada. Conjectura-se que é computacionalmente inviável produzir duas mensagens que tenham o mesmo *digest*, ou para produzir qualquer mensagem dado um *digest* alvo pré-especificado. O algoritmo MD5 foi desenvolvido para aplicações de assinatura digital, especialmente idealizado para máquinas de 32 *bits*, onde o seu desempenho é otimizado. O algoritmo MD5 é de simples codificação e compacto.

O MD5 faz parte de uma família de algoritmos de *digest*, onde outros algoritmos como o MD2 e MD4 também fazem parte, mas tem um projeto mais "conservador" (especialmente se comparado com MD4) para garantir mais segurança em detrimento de um pouco da velocidade. O algoritmo MD5 é de domínio público.

A dificuldade de gerar duas mensagens com o mesmo *digest* é da ordem de 2^{64} operações, e a dificuldade de produzir uma mensagem a partir de um *digest* original é calculado como sendo da ordem de 2^{128} operações. Entretanto, por ser um algoritmo novo, ainda não está sedimentado como um padrão de segurança.

O algoritmo MD5 inicia-se com algumas definições [RIV 92]. Supondo uma mensagem de *b bits* (com *b* sendo um inteiro não negativo), não necessariamente múltiplo de 8. Os *bits* estão na ordem m_0, m_1, ..., m_{b-1}. A mensagem deve ser estendida (*padded*) de modo que seu tamanho em *bits* fique congruente a 448, módulo 512, tornando-a apenas 64 *bits* "distante" de ter um tamanho múltiplo de 512 *bits*. O *padding* é realizado pelo acréscimo de um único *bit* 1 seguido de *bits* 0 tantos quantos necessários para a mensagem chegar a um tamanho congruente a 448, módulo 512. Nestes 64 *bits* ainda necessários para completar o tamanho múltiplo de 512, é acrescentado o tamanho *b*, original, representado em 64 *bits*.

Sobre esta mensagem estendida, são aplicadas sucessivamente quatro funções lógicas de transformação entremeadas por rotações e operações aritméticas, utilizando ainda uma tabela constante gerada a partir de funções trigonométricas. São realizados quatro ciclos destes, com pequenas alterações nas funções que são aplicadas. O resultado destas operações são quatro blocos de 32 *bits* cada que são concatenados, formando o *digest* de 128 *bits* requerido.

BIBLIOGRAFIA

[BAL 92] BALL, L. L. *Cost-Efficient Network Management*, McGraw-Hill, 1992.

[BRI 93] BRISA, *Gerenciamento de Redes: Uma Abordagem de Sistemas Abertos*, Makron Books, 1993.

[CAS 90] CASE, J., FEDOR, M., SCHOFFSTALL, M & DAVIN, J., *A Simple Network Management Protocol (SNMP)*, RFC 1157, 1990.

[COL 89] COLLINS, W. *OSI Management Services Elements, Protocols and Application Layer Structure (ALS)*, Procedings of Integrated Network Management I, 1989.

[COO 89] COOPER, J. A., *Computer and Communications Security*, McGraw-Hill, 1989.

[DEL 94a] DE LUCCA, J. E., SPECIALSKI, E. S. & WESTPHALL, C. B., *Arquitetura de Segurança para Gerência de Redes* - PANEL'94 - XX Conferência Latino-Americana de Informática - Cidade do México, México, 1994.

[DEL 94b] DE LUCCA, J. E. & SPECIALSKI, E. S., *Arquitetura de Segurança em Redes de Computadores* - CCBol'94 - Conferência de Ciências de la Computación - Cochabamba, Bolívia, 1994.

[DEL 94c] DE LUCCA, J. E., *Arquitetura de Segurança para Gerência de Redes*, Relatório de Trabalho Individual - CPGCC/UFSC - Florianópolis, SC - 1994.

[DOD 83] USA DEPARTMENT OF DEFENSE COMPUTER SECURITY CENTER, *Trusted Computer System Evaluation Criteria - Orange Book - CSC-STD-001-83*, USA-DoD, 1983.

[GAL 91] GALVIN, J. M. & McCLOGHRIE, K. & DAVIN, J.R., *Secure Management of SNMP Networks*, Proccedings of Integrated Network Management II, 1991.

[GAR 92] GARFINKEL, S. & SPAFFORD, G., *Practical UNIX Security*, O'Reilly & Associates, 1992.

[HAW 90] HAWKING, S., *Uma Breve História do Tempo*, Ed. Rocco, 1990.

[ISO 89] ISO/TC 97, *IS-ISO 7498-2: Information processing systems - Open Systems Interconnection - Basic Reference Model - Part 2: Security Architecture*, ISO, 1989.

[ISO 90a] ISO/TC 97, *DSI-ISO 10164-7: Information Technology - Open Systems Interconnection - Systems Management - Part 7: Security Alarm Reporting Function*, ISO, 1990.

[ISO 90b] ISO/TC 97, *DSI-ISO 10164-8: Information Technology - Open Systems Interconnection - Systems Management - Part 8: Security Audit Trail Function*, ISO, 1990.

[ISO 90c] ISO/TC 97, *DSI-ISO 10164-9: Information Technology - Open Systems Interconnection - Systems Management - Part 9: Object and Attributes for Access Control*, ISO, 1990.

[KAR 91] KARILLA, A. T., *Open Systems Security: An Architectural Framework*, Helsink, Finlândia, 1991. (Tese de Doutorado)

[KLE 88] KLERER, S. M, *The OSI Management Architecture: an Overview*, IEEE Network, vol.2, núm. 2, pp 20-29, março 1988.

[KOL 87] KOLMAN, B. & BUSBY, R. C., *Discrete Mathematical Structures for Computer Science*, 2a. ed., Prentice-Hall, 1987.

[MAD 94] MADRUGA, E. L., *Ferramentas de Apoio à Gerência de Falhas e Desempenho em Contexto Distribuído*, Porto Alegre, Brasil, 1994. (Dissertação de Mestrado)

[MCL 90] MCLOGHRIE, K. e ROSE, M., *Management Information Base for Network Management of TCP/IP-based Internet*, RFC 1156, 1990.

[NEC 90] NECHVATAL, J., *Public Key Cryptography,* National Institute of Standards and Technology - NIST, 1990.

[OMU 90] OMURA, J., *Novel Applications of Cryptography in Digital Communications*, IEEE Communications Magazine, vol. 4, núm. 2, pp 21-34, maio 1990.

[PIN 94] PINTO, A. C., *Proposição de Funções para Gerência de Segurança em Redes*, Relatório de Trabalho Individual 411, CPGCC-UFRGS, 1994.

[PFL 89] PFLEEGER, C. P., *Security in Computing*, Prentice-Hall, 1989.

[RAM 94] RAMOS, A. M., *Método de Controle de Acesso para Gerenciamento de Segurança*, Dissertação de Mestrado, CPGCC - UFSC, 1994.

[RIV 92] RIVEST, R., *The MD5 Message Digest Algorithm*, RFC 1321, MIT e RSA Data Security Inc, abril 1992.

[ROT 93] ROTEMBERG, M., *Communications Privacy Implications for Network Design, Communications of the ACM,* vol 36, n. 8, pp 87-95, agosto 1993.

[ROZ 89] ROZIER, M. et alli., CHORUS Distributed Operating Systems IN *Computing Systems*, vol.1, num.4, fevereiro 1989.

[SEV 89] SEVCIK, P. J. & KORN, L. K., *A Network Monitoring and Control Security Architecture*, Procedings of Integrated Network Management I, 1989.

[SHA 49] SHANNON, C., *Communication Theory of Secrecy Systems*, Bell System Technical Journal, v.28, pp 656-715, 1949.

[SOU 92] SOUZA, R., *Arquitetura e Gerência de Segurança no OSI*, OSI'92 - Seminário de Conectividade e Interoperabilidade OSI - BRISA, São Paulo, 1992.

[SUN 89] SUN MICROSYSTEMS, Inc, *SunNet Manager Tutorial - How to write an Agent*, 1989.

[SUN 90a] SUN MICROSYSTEMS, Inc, *Network Programming Guide*, 1990.

[SUN 90b] SUN MICROSYSTEMS, Inc, *SunOS Reference Manual*, vol. I, 1990.

[SUN 91] SUN MICROSYSTEMS, Inc, *SunNet Manager 1.1: Installation and User's Guide*, 1991.

[TAN 91] TANENBAUM, A. *Computer Networks*, 2a. ed., Prentice-Hall, 1991.

[TAN 92] TANENBAUM, A. *Modern Operating Systems*, Prentice-Hall, 1992.

[WES 92] WESTPHALL, C. B. & ASSOUL, S. *Management Architecture for Networks of the Future*. IEEE/IFIP Distributed Systems: Operation and Management. Munich, Alemanha, 1992.

[WES 93] WESTPHALL, C. B. & SILVA, A.C.B., *Desenvolvimento e Integração de Agentes para Gerência de Redes Locais,* Relatório de Pesquisa 209, CPGCC-UFRGS, Porto Alegre, 1993.