UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Jean Everson Martina

Projeto de um Provedor de Serviços Criptográficos Embarcado para Infra-estrutura de Chaves Públicas e suas Aplicações

dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de mestre em Ciência da Computação.

Prof. Ricardo Felipe Custódio, Dr.
Orientador

Florianópolis, Março de 2005

Projeto de um Provedor de Serviços Criptográficos Embarcado para Infra-estrutura de Chaves Públicas e suas Aplicações

Jean Everson Martina

Esta dissertação foi julgada adequada para a obtenção do título de mestre em Ciência da Computação, área de concentração Sistema de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Prof. Raul Sidnei Wazlawick, Dr.
Coordenador do Curso

Banca Examinadora

Prof. Ricardo Felipe Custódio, Dr.
Orientador

Prof. Guido Costa Souza de Araújo, Phd.

Prof. Ricardo Dahab, Phd.

Prof Jeroen Antonius Maria van de Graaf, Phd.

Prof. Daniel Santana de Freitas, Dr.

"High risk insurance. The time is right. High risk insurance. The time is right. Got endurance, I was trained. I got my sights adjusted and my telescope aimed. Everybody wants an explanation. Got no love for the enemy nation. You gotta fight to stay independent. I got my pride and I'm gonna defend it."

The Ramones - High risk insurance - End of the Century - 1980

Ofereço às minhas 3 mulheres: minha Irmã, minha Mãe e minha Namorada.

Agradecimentos

Agradeço primeiramente aos meus pais pelas oportunidades que me propiciaram para chegar neste ponto e pela sua grande visão e persistência. Agradeço também a minha irmã Jane, pelas conversas e por compartilhar tudo o que vivemos até hoje.

Não por menor deve ser meu agradecimento ao Professor Ricardo Custódio, que me adotou neste caminho e sempre me fez permanecer nele, por mais adversas que as situações pudessem parecer. O agradecimento ao Professor Daniel Santana pela inúmeras horas e idéias compartilhadas, as quais sem dúvida me trouxeram a este ponto.

Também tenho que agradecer aos colegas de LabSec, os quais foram sempre muito encorajadores nos momentos difíceis, principalmente a pessoa do Júlio Dias que foi um companheiro em todo o caminho.

Gostaria também de agradecer a RNP - Rede Nacional de Pesquisa - pelo apoio financeiro através dos Projetos ICP-EDU e ICP-EDUII no decorrer de grande parte do meu curso.

Por fim gostaria de agradecer a Giseli, minha companheira, a qual sem dúvida foi a que mais sofreu nesta fase, viveu comigo o dia a dia e sempre me deu forças para superar.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas

AC	Autoridade Certificadora
ACD	Arquivo de Certificados Digitais
ACU	Autoridade Certificadora Única
AES	Advanced Encryption Alghorithm (Algorítmo de Cifragem Avançada)
AR	Autoridade de Registro
CBC	Cipher Block Chaining (Encadeamento de Blocos Cifrados)
CC	Common Criteria (Critérios Comuns)
DPC	Declaração de Práticas de Certificação
DES	Decryption and Encryption Standard (Padrão de Cifragem e Decifragem)
DPCPC	Dipositivo Próprio com Capacidade de Processamento Criptográfico
EAL	Evaluation Assurance Level (Nível de Avaliação de Garantia)
GPC	Gerador de Par de Chaves
GNA	Gerador de Números Aleatórios
ICP	Infra-Estrutura de Chaves Públicas
ISO	International Standards Organization (Organização Internacional de Padrões)
LCR	Lista de Certificados Revogados
LEC	Lista Estendida de Confiança
NIST	National Institute of Standarts (Instituto Nacional de Padrões)
PC	Política de Certificados
PP	Protection Profile (Perfil de Proteção)
PSC	Provedor de Serviços Criptográfios
RCD	Repositório de Certificados Digitais
RSA	Rivest, Shamir e Adelman

SEE Secure Execution Engine (Motor de Execução Segura)

SHA Secure Hash Algorithm (Algoritmo Seguro de Resumo Criptográfico)

SF Security Function (Função de Segurança)

SFP Security Function Policy (Política da Função de Segurança)

SOF Strength of Function (Força da Função)

SSL Secure Sockets Layer (Camada de Soquete Seguro)

ST Security Target (Alvo de Segurança)

TOE Target of Evaluation (Alvo de Avaliação)

TSC TSF Scope of Control (Escopo de Controle das Funções de Segurança do Alvo de Avaliação)

TSF TOE Security Functions (Funções de Segurança do Alvo de Avaliação)

TSFI TSF Interface (Interface das Funções de Segurança do Alvo de Avaliação)

TSP TOE Security (Segurança do Alvo de Avaliação)

UCD Usuário de Certificados Digitais

VPN Virtual Private Network (Rede Privada Virtual)

Lista de Símbolos

$\oplus$ - XOR/Ou exclusivo

$\leq$ - Menor ou igual que

$\geq$ - Maior ou igual que

Resumo

Esta dissertação de mestrado contém uma proposta para o projeto de um provedor de serviços criptográficos embarcado para uso em infra-estruturas de chaves públicas -ICP- e suas aplicações. O projeto consiste na especificação de um modelo de gerenciamento de chaves apoiado por um equipamento especialmente projetado para este fim

A dissertação apresenta os conceitos básicos de criptografia e o estado da arte da tecnologia das infra-estruturas de chaves públicas, mostrando a necessidade do uso de provedores de serviços criptográficos para a proteção de chaves critpográficas.

No trabalho, também são cobertas as normas internacionais relevantes à construção de dispositivos criptográficos, dando ao leitor o entendimento e abrangência das mesmas, relacionando diretamente os procedimentos que são necessários para prover os conjuntos de garantias necessárias ao projeto. Também são avaliados alguns produtos comerciais segundo estas normas, sendo feitas comparações qualitativas entre eles.

O cerne do trabalho consiste na construção de um modelo de gerenciamento de chaves adaptado para o uso em ambientes de ICP, levando em conta também as mais variadas aplicações existentes no mercado. Da definição deste modelo, partimos para o projeto de construção do equipamento, a qual consiste no levantamento dos requisitos de construção física, do detalhamento dos componentes de software e do projeto de testes do produto final do trabalho.

A dissertação apresenta em detalhes a construção de um protótipo do provedor de serviços criptográficos proposto, e conclui com detalhamentos dos projeto de implementação que ajudaram a validá-la.

Palavras Chaves: Gerenciamento de Chaves Criptográficas, Criptografia Assimétrica, Infra-Estrutura de Chaves Públicas, Módulos de Segurança Criptográfica.

Abstract

This master's dissertation proposes a project to implement a Hardware Security Module(HSM) focused to work with Public Key Infrastructures - PKI and its applications. The project consists of a key management scheme with a brief of hardware designed to protect it.

The dissertation presents the basic concepts of cryptography and the *state of art* of public key infrastructure technology, presenting the necessity of use of an HSM to protect keys and processes.

This work also covers international regulations on the matter of building an HSM and tries to show the reader how related they are in providing the warranties needed by the project. Some commercial products are studied, related to the regulations and also qualitatively compared.

The dissertation core consists in building a key management system adapted to use in PKI environments, taking care of a great sort of PKI aware applications. From this point we show how to construct an HSM that implements this key management system, developing requirements of hardware, software and test for it.

Finally, we detail the construction of an HSM prototype, concluding the work with its implementation

Keywords: Cryptographic Key Management, Asymmetric Cryptography, Public Key Infrastructure, Hardware Security Modules.

Capítulo 1

Introdução

É cada vez maior e mais crescente a dependência da sociedade em plataformas computacionais no seu quotidiano. Neste sentido as plataformas computacionais devem prover, além da agilidade e precisão na busca e zelo da informação, a segurança necessária para os documentos nelas armazenados.

Por segurança da informação entende-se os requisitos de integridade, tempestividade, sigilo e confiança no armazenamento [1]. Para obter estas garantias, fazemos uso de procedimentos criptográficos.

Os procedimentos criptográficos são métodos matemáticos evoluídos de técnicas de criptografia convencional, os quais são processados por sistemas computacionais, no intuito de prover os requisitos de segurança. Os procedimentos criptográficos podem ser divididos em 3 grandes classes: os de chave simétrica, os de chave assimétrica e os de resumo.

Nos procedimentos que fazem uso de chaves criptográficas, como os de chave simétrica e de chaves assimétricas, sua segurança é diretamente relacionada com a proteção das chaves criptográficas ou dos mecanismos que permitem o acesso às mesmas.

Os hardwares de proteção criptográfica são escudos que visam proteger as chaves e os processos criptográficos utilizados por sistemas de alto valor agregado. Entende-se por alto valor agregado os sistemas que utilizam chaves para cifrar e/ou assinar documentos atribuídos a transações que envolvem valores monetários significativos.

1.1 Contextualização

1.1.1 Criptografia Simétrica

A criptografia simétrica, também conhecida como criptografia de chave secreta, tem como base o uso da mesma chave para os processos de cifragem e de decifragem dos dados enviados.

Os mecanismos de criptografia simétrica, também chamados de algoritmos simétricos de criptografia, são métodos pelos quais uma determinada entrada sofre um embaralhamento através de sucessivas permutações e substituições, com a adição da chave ao processo de embaralhamento.

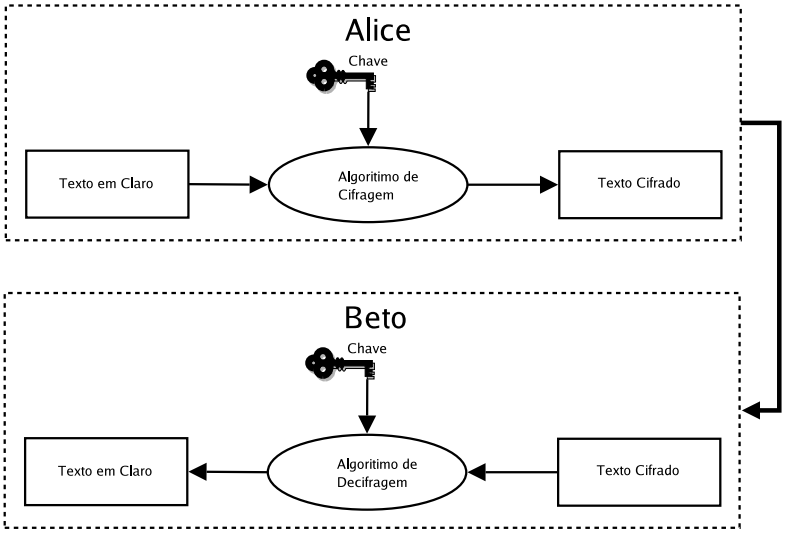


Figura 1.1: Cifragem e decifragem simétrica.

Quando Alice quer enviar alguma informação sigilosa para Beto, ela o fará usando um algoritmo simétrico de criptografia. Para tal, Alice deve previamente ter trocado, através de um canal seguro, com Beto uma chave secreta. Como vemos na figura 1.1, Alice, de posse da mensagem que quer enviar a Beto e de seu segredo compartilhado com Beto, transforma esta mensagem através do algoritmo de cifragem em uma mensagem cifrada, a qual pode trafegar de forma segura por um canal desprotegido. Beto, ao receber a mensagem cifrada, detecta que a mensagem vem de Alice e de posse do segredo que compartilha com Alice, através do algoritmo de decifragem, transforma a mensagem cifrada novamente em uma mensagem inteligível.

Para um atacante Charles, a descoberta do texto em claro deve sempre

ser mais custosa do que o ataque por força bruta, o qual consiste em, dado um par de entrada e saída conhecidas, varrer o espaço de chaves até encontrar a chave utilizada, para que o algoritmo seja considerado seguro.

Dentre os algoritmos de criptografia simétrica atuais se destacam o DES [2] e o AES [3], ambos padronizados para o uso pelo governo norte-americano. Para evitar ataques tal qual a análise de freqüência, os algoritmos simétricos são dispostos na forma de modos de operação [4].

Os modos de operação definem como os textos de saída ou de entrada dos algoritmos serão encadeados, de forma que de posse de pares de texto em claro e texto cifrado, um atacante não possa determinar a relação entre os textos.

1.1.2 Funções de Resumo Criptográfico

As funções de resumo criptográfico, também conhecidas como código de autenticação de mensagem ou funções simétricas de integridade, são valores agregados a mensagem original para garantir a sua integridade durante o transporte. Elas também podem autenticar a entidade que enviou a mensagem perante a entidade que a recebeu quando utilizadas com um segredo compartilhado entre as entidades.

Então se Alice quer prover além de confidencialidade, a integridade, antes de cifrar uma mensagem com um algoritmo simétrico de cifragem ela pode submeter o texto de entrada a uma função de resumo criptográfico, conforme a Figura 1.2, anexando o resultado ao texto cifrado.

A construção de uma função simétrica de integridade pode ser facilmente atingida, simplesmente utilizando um algoritmo de cifragem simétrica padrão no modo de operação de encadeamento de texto cifrado (CBC) e truncando a saída [5]. A utilização deste método, além de prover a integridade, provê a autenticação, mas não é suficiente para resolver uma disputa entre Alice e Beto, pois ambos possuem a chave que controla o processo, o que possibilita a alteração por ambas as partes.

Outra forma de prover integridade é o uso de funções hash de caminho único. Estas funções devem prover as seguintes características [6]:

- Ser computacionalmente inviável recriar a mensagem original a partir do resultado da função;

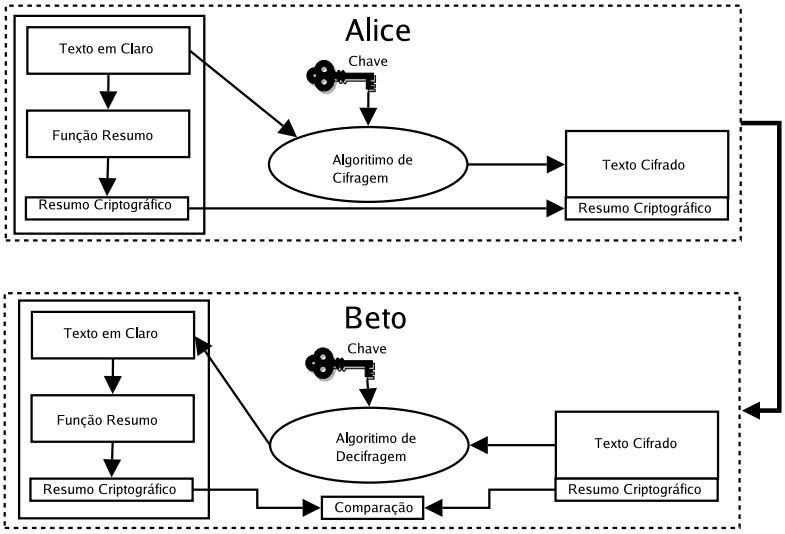


Figura 1.2: Cifragem simétrica com controle de integridade por função resumo.

- Ser computacionalmente inviável construir duas mensagens diferentes com a mesma saída da função.

Dentre as funções de caminho único atuais destaca-se o Secure Hash Algorithm (SHA) [7], o qual foi padronizado para uso pelo governo norte-americano como função de integridade e na sua versão padrão produz uma saída de 160 bits. O NIST da Secretaria de Comércio dos Estado Unidos da América definiu uma família de variantes do SHA, entre elas, o SHA-256, SHA-384 e o SHA-512, com respectivamente 256, 384 e 512 bits de saída.

1.1.3 Criptografia Assimétrica

A criptografia assimétrica, também conhecida como criptografia de chave pública, foi inicialmente proposta por Diffie e Hellman [8] em 1976, como um sistema revolucionário que veio para sanar um dos grandes problemas da criptografia simétrica, a distribuição de chaves de forma segura.

No sistema proposto por Diffie e Hellman, cada usuário do sistema criptográfico, ao invés de possuir uma única chave para se comunicar com outra parte, passaria a possuir duas chaves, uma distribuída publicamente, conhecida como chave pública, e uma para suas operações secretas, conhecida como chave privada. Não deve ser possível

a partir da chave tornada pública obter-se a chave privada.

O mecanismo de cifragem de decifragem de dados foi concebido de forma que as chaves pública e privada operem sempre de forma complementar, ou seja, tudo o que é cifrado com a chave pública só pode ser decifrado com a respectiva chave privada

Neste novo paradigma de criptografia, além da não necessidade de canais comprovadamente seguros para a troca das chaves, diminui-se drasticamente a quantidade de chaves que cada usuário deveria guardar para se comunicar com todos os possíveis outros usuários. Este novo mecanismo também passou a permitir que mesmo pessoas que não se conhecessem pudessem trocar mensagens de forma cifrada.

Para fazer uso do sistema de criptografia assimétrica, um individuo deve primeiramente gerar um par de chaves criptográficas, sendo que cada um será individualmente usado para os processos de cifragem e de decifragem.

O sistema de criptografia assimétrica mais amplamente conhecido na atualidade é o sistema desenvolvido por Rivest, Shamir e Adelman, e conhecido como RSA [9]. Este sistema se baseia nas idéias de Diffie e Hellman e implementa um difícil problema matemático, baseado na teoria de números, que é a fatoração de produtos de números primos em módulo n.

Neste novo sistema temos a independência do uso das chaves nos processos de cifragem e decifragem, podendo assim gerar as bases para um outro novo paradigma que é a assinatura digital de documentos.

A assinatura digital de documentos, consiste na criação de uma marca única na qual o assinante atesta o conhecimento do conteúdo do documento eletrônico, evidenciando que concorda com ele, a qual é obtida através do uso de sua chave privada, podendo ser facilmente verificável por qualquer usuário através da respectiva chave pública do assinante.

Dadas as características do uso da chave privada, a necessidade de proteção sobre ela é evidente. Para que possamos prover estas necessidades de proteção devemos armazená-la e controlar a sua operação de maneira segura. Para alcançar estes objetivos vamos fazer uso dos provedores de serviços criptográficos.

Definição: PSC - Provedor de Serviços Criptográficos, conforme a Figura 1.3, é uma camada de software que provê a um módulo de hardware criptográfico as funciona-

lidades de gerenciamento e controle do ciclo de vida das chaves por ele gerenciadas.

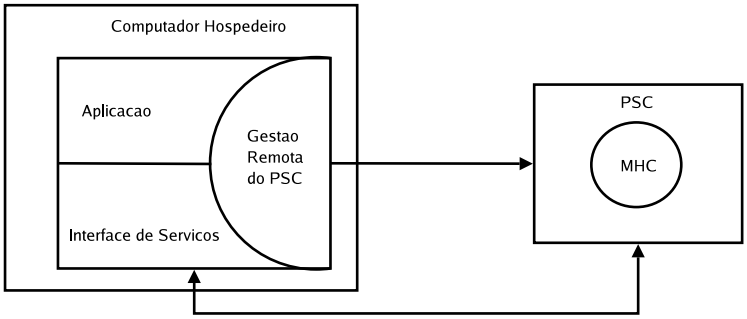


Figura 1.3: Estrutura Lógica e Ambiente de uso do PSC.

1.2 Objetivos

1.2.1 Objetivo Geral

O trabalho tem por objetivo geral o projeto e a especificação para a construção de um provedor de serviços criptográficos capaz de proteger e gerenciar processos e chaves criptográficas para ambientes de alto valor agregado da informação com auxílio de um módulo de hardware criptográfico.

Estão excluídos dos objetivos deste trabalho o detalhamento e construção de hardware específico, sendo somente abordado este tópico na definição dos requisitos necessários ao funcionamento e proteção do PSC, os quais serão providos pelo hardware.

1.2.2 Objetivos Específicos

- Levantamento dos requisitos de proteção de chaves por parte de uma Infra-Estrutura de Chaves Públicas;
- Levantamento das normas e padrões internacionalmente aceitos para a construção de mecanismos de guarda e zelo de chaves criptográficas;
- Levantamento de sistemas existentes capazes de exercer funções de guarda para chaves criptográficas;

- Projeto de um protocolo para o gerenciamento de chaves e processos criptográficos para uso na construção de um equipamento capaz de protegê-los;

- Projeto e Implementação de um protótipo de um provedor de serviços criptográficos usando hardware genérico;

- Estudo de sistemas de troca de mensagens criptográficas entre provedores de serviços criptográficos, bibliotecas de criptografia e sistemas existentes.

1.3 Justificativa e Motivação

O trabalho tem relevância estratégica para o Brasil no atual cenário de segurança da informação, uma vez que a maior parte das infra-estruturas de chaves públicas atualmente em funcionamento estão sendo protegidas por equipamentos estrangeiros. Isto impede qualquer procedimento de auditoria nos equipamentos, portanto, eles carecem dos necessários indícios de transparência e confiabilidade pela sociedade brasileira. Este trabalho contribuiu no sentido de que a sociedade possa estudar e realmente conhecer os equipamentos para proteção de chaves e processos criptográficos.

Definição: ICP - Infra-Estrutura de Chaves Públicas, é o conjunto de componentes necessários para suportar o uso de chaves públicas em ambientes heterogêneos e distribuídos.

Os esforços do trabalho são para a construção de provedores de serviços criptográficos voltados para o uso em ambientes de ICP, dando a eles características inerentes deste ambiente. No entanto o fruto do trabalho não se destina unicamente a este uso, podendo facilmente ser adaptado para uso em quaisquer componentes da ICP e também em aplicações que dependem dela, tais como túneis seguros.

1.4 Trabalhos Relacionados

Várias iniciativas de pesquisa no campo de provedores de serviços criptográficas tem sido iniciadas no Brasil atualmente. As pesquisas hoje levam em conta a realidade do cenário nacional, em uma estrutura de ICP hierárquica e de raiz única [10],

e buscam tratar os problemas de segurança endereçados especificamente a este tipo de plataforma.

No âmbito destas pesquisas, o Instituto Nacional de Tecnologia da Informação do Governo Federal Brasileiro, ITI, detentor da autoridade certificadora raiz Brasileira (ICP-Brasil), lançou em outubro de 2003 o projeto João de Barro [11] .

O projeto João de Barro consiste no esforço de construção de uma plataforma livre e segura para o uso no âmbito da AC-Raiz da ICP-Brasil, o que envolve a construção de um provedor de serviços criptográficos embarcado para a guarda de chaves e da construção de um sistema gestor de certificados digitais. Atualmente o Brasil é dependente na sua estrutura de ICP governamental de software proprietário, confiando sua segurança a sistemas estrangeiros e não auditáveis.

Segundo Pagliusi [12], o uso deste tipo de tecnologia estrangeira representa uma grande fonte de incerteza e de evasão de divisas, e necessita de uma nacionalização e pleno entendimento da sociedade devido à sua posição estratégica no atual cenário de segurança da informação no Brasil.

No campo acadêmico temos a colocação do projeto ICP-EDU II [13] da Rede Nacional de Pesquisa, e que é a segunda fase do desenvolvimento de uma infraestrutura tecnológica para a implementação de uma ICP acadêmica para todas as universidades brasileiras.

Durante a primeira fase do projeto ICP-EDU, foi desenvolvido um sistema gestor de certificados com código fonte aberto e livre para uso acadêmico [14]. Deste desenvolvimento surgiram as necessidade de uso de provedores de serviços criptográficos, os quais, conforme detalharemos no capítulo 4, são exorbitantemente caros para o cenário acadêmico brasileiro.

Com a proposta de implementação de um equipamento equivalente aos modelos comerciais, o ICP-EDU II, dá assim, a possibilidade da disseminação da cultura de certificados digitais e de segurança por todo o meio acadêmico nacional. Este projeto encontra-se atualmente em desenvolvimento e conta com parcerias das mais importantes universidades do meio acadêmico brasileiro e empresas do setor.

A grande evolução propiciada hoje no campo dos equipamentos para proteção de chaves e processos criptográficos são os propostos pela indústria, em especial os fabricados pelas empresas IBM e nCipher. Estas evoluções caminham tanto em

mecanismos de proteção mais evoluídos para os equipamentos, quanto para mecanismos lógicos para a proteção dos parâmetros críticos de segurança [15, 16].

Vistas especiais devem ser dadas aos rumos de desenvolvimento dado pela empresa inglesa nCipher, a qual, após ter obtido um alto grau de maturidade na proteção de parâmetros críticos de segurança, abre um novo paradigma, o da execução segura de código, através da SEE engine [17]. A SEE engine provê uma API para o desenvolvimento de aplicações e o embarque das mesmas no equipamento disponibilizado pela empresa, provendo áreas limitadas para a execução segura de código arbitrário desenvolvido pelos seus clientes.

Na área acadêmica, um trabalho que levanta o estado da arte atual na pesquisa sobre PSC, do ponto de vista da implementação das API de segurança é a tese de Bond [18]. Neste trabalho são levantados desde dados históricos sobre a evolução dos equipamentos, até mecanismos para validação formal das APIs de segurança desenhadas para eles, passando por vários ataques construídos pelo autor sobre equipamentos hoje existentes no mercado.

O trabalho de Bond propõe uma abertura deste novo campo de pesquisa que é a avaliação e validação das API de segurança criptográficas, dando base para avaliação das API existentes e também para a construção de novas API resistentes aos ataques nela expostos. Esta contribuição de Bond é adotada na construção da API resultante deste trabalho.

1.5 Estrutura do Trabalho

Este trabalho visa inicialmente estabelecer os conceitos e aplicações esperadas do projeto de um PSC, tendo no capítulo 2 uma revisão dos conceitos de Infra-Estruturas de Chaves Públicas para contextualizar a necessidade de proteção dos processos e chaves criptográficas, enfatizando a necessidade da proteção das chaves privadas nos ambientes de criptografia assimétrica.

Dando continuidade à explanação dos conceitos, o capítulo 3 revisa as normas existentes para a construção e a validação de dispositivos criptográficos, assim como de seus ambientes operacionais, deixando o leitor capacitado para entender posteriormente os requisitos atendidos pelo nosso projeto e pelos equipamentos comerciais,

monstrados no capítulo 4, onde fazemos um levantamento dos equipamentos disponíveis no mercado, avaliando e coletando suas mais importantes características. Estas características vão embasar a pesquisa, indicando quais são as medidas eficientes para a construção de um provedor de serviços criptográficos, e também evidenciando as deficiências dos equipamentos no mercado.

Prosseguindo, no capítulo 5, temos o projeto de construção de um provedor de serviços criptográficos embarcado em hardware, tratando de seus requisitos, os quais incluem, os projetos físico, lógico e de testes, e também todos os modelos e especificações de ambiente necessários para a construção do mesmo, dando base para a implementação do protocolo proposto no capitulo 6. No capítulo 7 temos os detalhes do projeto do protótipo implementado usando-se o protocolo proposto no trabalho.

O cerne do trabalho surge no capitulo 6, onde temos a modelagem e a especificação de um protocolo de gerenciamento do ciclo de vida das chaves gerenciadas pelo provedor de serviços criptográficos. Este protocolo será responsável pelas garantias de segurança no armazenamento e manipulação das chaves. A preocupação com o ambiente operacional de uma ICP é evidenciada, sendo garantidas as medidas para o atendimento dos requisitos de segurança para este ambiente.

Por fim, mostramos as considerações finais advindas da pesquisa para este trabalho no capítulo 8, evidenciando as possibilidades de trabalhos futuros nesta área de pesquisa.

Capítulo 2

Infra-Estruturas de Chaves Públicas

Este capítulo tem por principal objetivo esclarecer conceitos sobre a utilização dos provedores de serviços criptográficos para a proteção de processos e chaves criptográficas.

As chaves criptográficas são utilizadas por inúmeras aplicações, dentre as quais, destacam-se as infra-estruturas de chaves públicas e o estabelecimento de túneis de comunicação seguras, tais como o SSL e VPNs. Em qualquer uma destas aplicações, a definição da escolha do nível da proteção das chaves envolvidas deve levar em consideração o valor da informação protegida pelos componentes da estrutura como um todo, de forma a manter o custo do ataque ao sistema maior do que o valor da informação por ele protegida.

Neste capítulo, faremos uma revisão dos conceitos de ICP, de forma a apontar onde e de que forma são usados os provedores de serviços criptográficos para a proteção das chaves criptográficas, tornando evidente a inclusão do presente trabalho neste contexto.

2.1 Gerenciamento de Chaves Assimétricas

Com o nascimento da criptografia de chaves públicas, passamos a ter novas possibilidades de uso para a criptografia, e o conseqüente surgimento de novas necessidades. Dentre elas, está a questão de como associar uma chave pública e o seu efetivo controlador.

A proposta para a solução do problema veio de Loren Kohnfelder [19],

quando o mesmo propôs o uso de uma terceira parte confiável para atestar a relação de uma chave pública ao efetivo detentor da chave privada, criando assim a chamada Autoridade Certificadora - AC.

A AC é responsável pela identificação do usuário e por um atestado de que ele possui a chave privada correspondente à chave pública. Este processo é feito através da emissão de um documento assinado pela AC denominado certificado digital, contendo a identificação do usuário, sua chave pública e propriedades atribuidas a mesma.

2.2 Certificados Digitais de Chaves Públicas

Os Certificados Digitais não são um novo conceito para nós, e estão presentes em vários momentos do nosso dia a dia. Como, por exemplo, num cartão de visitas, que identifica uma pessoa e traz vários dados agregados a ele, assim como um cartão de crédito que relaciona um número a uma pessoa, uma assinatura a uma data de validade [6].

Quando pensamos num certificado digital, devemos fazê-lo levando em conta uma série de características que gostariamos que fossem atingidas. Segundo Housley e Polk [6], um certificado ideal deve ter as seguintes propriedades:

1. deve ser um objeto puramente digital, para que possamos enviá-lo pela rede e processá-lo automaticamente;

2. deve conter o nome do usuário que detém a chave privada, incluindo informações de contato e da organização a qual pertence;

3. deve ser possível determinar se o certificado foi recentemente emitido;

4. deve ser criado por um terceira parte confiável, ao invés do próprio usuário;

5. a mesma terceira parte deve ser capaz de criar vários certificados, até vários para o mesmo usuário e diferenciá-los;

6. deve ser fácil determinar se o certificado foi forjado ou se é genuíno;

7. deve ser auto-contido e à prova de alterações;

8. deve-se poder verificar, de forma imediata, se alguma informação no certificado não é mais válida;

9. deve-se poder determinar para qual aplicação o certificado é válido.

O certificado digital de chaves públicas, como especificado pelo ITU-T [20], provê diretamente sete das nove propriedades levantadas por Housley e Polk, sendo ele um objeto puramente digital, passível de envio pela rede e processável automaticamente. Ele também contém a identificação do dono da chave a ele relacionada, assim como uma data de validade e a assinatura da AC que o emitiu, o que garante sua auto-contenção e unicidade, assim como a característica de ser à prova de modificação. A Autoridade Certificadora é capaz de emitir certificados para vários usuários e inclusive vários para o mesmo usuário, através do uso de um número serial para cada certificado.

Numa terceira revisão por parte do ITU-T [21], os certificados digitais de chaves públicas ganharam uma nova característica, conhecida como extensões. Com o uso destas extensões, é possível agregar a qualquer certificado digital uma informação ou dado, e torná-lo parte do certificado, passando assim a ter as mesmas garantias inerentes ao próprio certificado digital.

As duas últimas propriedades são as que não podem ser diretamente providas pelo certificado digital, o que torna necessário a inserção de mecanismos adicionais para a sua obtenção: as listas de certificados revogados e as políticas de certificados.

2.3 Listas de Certificados Revogados

Da necessidade de se manter atualizados os dados de um certificado digital, surgiram as Listas de Certificados Revogados - LCR. Elas são listas emitidas periodicamente pela Autoridade Certificadora ou por outra entidade à qual foi delegada esta função, onde são relacionados os certificados não mais válidos.

Os motivos que podem levar à perda de validade de um certificado digital podem ser inúmeros, dentre eles o comprometimento da chave privada ou a mudança de algum dos dados constantes no certificado digital.

Com a inserção deste novo mecanismo, conseguimos alcançar o oitavo objetivo proposto por Housley e Polk, mas também incluímos mais uma peça na estrutura de validação de um certificado. Uma vez estando o certificado validado em sua

forma auto-contida, deve-se certificar que o mesmo não está incluído na LCR. Além dos certificados digitais, suas políticas e das LCRs, vários outros elementos devem ser disponibilizados para a correta validação de um certificado digital. Estes elementos devem ser providos adequadamente. Para tal, é necessário organizá-los na forma de uma infraestrutura de chaves públicas - ICP.

2.4 Políticas de Certificados

As políticas de certificados são usadas para se definir o propósito do certificado digital, ou seja o seu uso.

As políticas de emissão de certificados são documentos escritos pelos gerentes de uma Autoridade Certificadora, estabelecendo para qual fim serão emitidos os certificados digitais por parte de uma autoridade certificadora, assim limitando os possíveis usos destes certificados. Normalmente, os documentos criados são as Políticas de Certificados - PC - e as Declarações de Práticas de Certificação - DPC.

As PCS abrangem aspectos gerais da emissão de certificados, tais como as características que os certificados podem possuir através de extensões. Já as DPCs, tratam dos detalhes de como serão implementados os serviços de emissão e gerenciamento de certificados.

As políticas podem ser divididas em duas classes, as formalizáveis e as não formalizáveis. As formalizáveis são passíveis de inclusão direta no certificado digital, sendo automática a sua validação. Já, as políticas não formalizáveis, são unicamente expressas nos documentos que as estabelecem, não podendo ser validadas automaticamente, sendo necessário o conhecimento do usuário da sua aplicabilidade para a validação.

O estabelecimento das políticas formalizáveis pode ser feito através do uso de extensões acrescidas ao certificado. Mesmo assim, o entendimento das políticas depende da aplicação que usa o certificado para seu fim de autorização ou autenticação. Esta aplicação deve ser capaz de determinar se aquela extensão permite ou não o uso do certificado para um determinado fim.

2.5 Componentes de uma Infra-Estrutura de Chaves Públicas

A simples utilização de certificados digitais pode envolver determinadas tarefas bastante complexas, tais como, o gerenciamento da emissão dos certificados, e a disponibilização destes certificados para consulta e uso por parte de seus usuários.

A infra-estrutura de chaves públicas facilita e provê uma maneira eficaz para este gerenciamento, sendo composta dos seguintes componentes :

- Autoridade Certificadora - AC;
- Autoridade de Registro - AR;
- Repositório de Certificados Digitais - RCD;
- Arquivo de Certificados Digitais - ACD;
- Usuários de Certificados Digitais - UCD;

2.5.1 Autoridade Certificadora

A Autoridade Certificadora é o elemento chave na construção de uma Infra-estrutura de Chaves Públicas. Ela é uma composição de mecanismos de software e de hardware e tem como função principal a emissão de certificados digitais.

Uma Autoridade Certificadora usa sua chave privada para assinar digitalmente um certificado de uma outra parte, a qual pode ser um usuário final ou uma outra Autoridade Certificadora, atestando assim o conhecimento do conteúdo deste certificado.

Contudo, estes não são os únicos papéis das ACs. As ACs executam as seguintes tarefas:

- emitem Certificados;
- emitem periodicamente as Listas de Certificados Revogados;
- publicam os certificados e as LCRs atuais;
- mantém arquivos de certificados e LCRs antigas;

- delegam responsabilidades para outros componentes da ICP.

Com uma única AC, podemos criar uma infra-estrutura de chaves públicas e através dela, operarmos plenamente. Mas é também muito comum que as tarefas executadas pela AC sejam delegadas a outros componentes da ICP com fins e atividades específicas dentro da mesma.

O intuito desta delegação de tarefas é a minimização da complexidade do componente AC, o qual é demasiadamente complexo e difícil de ser implementado de forma única. Este procedimento de delegação vale para todos os elementos. Uma AC pode delegar a outra AC, denominada AC intermediária, emitir certificados em seu nome, tal como uma procuração, ou delegar a emissão da LCR a outra AC, e delegar o processo de identificação dos usuários para uma Autoridade de Registro - AR, a qual será detalhada na seção 2.5.2.

2.5.2 Autoridade de Registro

A autoridade de registro - AR, atua por delegação da AC, e é composta normalmente por componentes de software, de hardware e operadores, os quais são responsáveis pela verificação dos dados constantes na requisição de certificado, e atestam a sua veracidade para a AC.

A existência deste componente é justificada pela grande abrangência que uma AC pode ter, seja ela por um domínio territorial, ou seja pelo número efetivo de usuários. Em ambos os casos, quando a AC emite um certificado, ela atesta que os dados nele contidos são verdadeiros e para tal é necessário que o usuário comprove tais dados. Para evitar que cada usuário tenha que se deslocar até a AC, existe a figura da AR, a qual faz esta verificação da informação em nome da AC.

Uma AC pode delegar este papel de verificar a inúmeras ARs, as quais compartilham com ela a responsabilidade dos dados constantes no certificado emitido.

2.5.3 Repositório de Certificados Digitais

O Repositório de Certificados Digitais, assim como a AR, também atua por delegação da AC, e é normalmente composto por um software com o objetivo de

tornar públicos os certificados digitais e as listas de certificados revogados atuais emitidos por uma ou mais ACs.

A presença do Repositório de Certificados Digitais deve-se à necessidade de interação da AC com os seus usuários, e também da necessidade da obtenção dos certificados e das LCRs.

O Repositório de Certificados Digitais é uma parte da ICP que deve estar sempre disponível aos usuários, diferentemente das ACs e das ARs, as quais necessitam de medidas de segurança que normalmente incluem, em alguns casos, a sua desconexão de ambientes de rede de dados. Com esta delegação por parte da AC, é possível que a mesma consiga um maior zelo pela sua chave privada, pelo simples fato de não se comunicar com qualquer outro computador, e também consiga manter a disponibilidade necessária aos seus usuários, podendo replicar inúmeras vezes o repositório.

Os dados armazenados e disponibilizados pelo Repositório de Certificados Digitais são assinados pela AC que ele representa, garantindo sua integridade e sua autenticidade, o que torna imune o repositório a ataques de substituição e fabricação.

2.5.4 Arquivo de Certificados Digitais

O Arquivo de Certificados Digitais atua também por delegação das ACs, e são normalmente compostos de software e hardware, com o objetivo de armazenar os certificados digitais e as listas de certificados revogados emitidos por uma AC após o seu período de validade.

O Arquivo de Certificados Digitais tem por responsabilidade, por prazo indeterminado, a manutenção dos certificados digitais emitidos pela AC, garantindo o seu correto armazenamento e encadeamento. Em geral normas jurídicas explicitam o tempo que os documentos devem ser armazenados, fazendo com que os Arquivos de Certificados Digitais sejam necessários para o cumprimento de tais normas.

A presença do arquivo é importante no caso de uma disputa que envolva a existência de um certificado digital. Neste caso, o arquivo poderá dispor do certificado em questão e atestar a sua validade na data na qual a disputa se baseia.

2.5.5 Usuários de Certificados Digitais

Os usuários de certificados digitais são as peças chaves de todo o mecanismo de uma ICP, pois é para eles que são emitidos os certificados digitais. Eles podem ser divididos em duas classes:

Detentores de certificados - São usuários que possuem um certificado emitido pela ICP, e o utilizam para assinar, cifrar dados e se identificar através da sua chave privada.

Partes que confiam no certificado - São usuários que utilizam certificados de outras partes para validação e conferência dos dados. Em particular, os serviços que eles mais utilizam são: verificação de assinaturas, cifragem de dados, e estabelecimento de conexões seguras.

Em geral, quando usamos certificados emitidos por uma ICP, acabamos atuando alternadamente entre os papéis, pois sempre temos informações sendo recebidas e informações sendo produzidas, assim atuando como detentor ou como parte confiante dentro de um processo interativo de troca de informações.

2.6 Arquiteturas para ICP

Considerando o crescente avanço das tecnologias de ICP, assim como a sua adoção em escala cada vez maior, começamos a ter problemas com a interoperabilidade entre os certificados emitidos por várias ICPs diferentes, tornando assim a vida do usuário bastante complexa, pois o mesmo tem que decidir em quem deve confiar.

Os problemas de interoperabilidade entre certificados emitidos por ICPs diferentes começam pelo fato de nem sempre existir a confiança por parte dos usuários dos certificados nos componentes de uma outra estrutura de ICP.

Para resolver este e vários outros problemas advindos do uso em larga escala de inúmeras ICPs, as ICPs podem ser organizadas de diversas maneiras denominadas arquiteturas de ICP, que visam ajudar os usuários no estabelecimentos de estruturas de confiança para efetivar a interoperabilidade de uso dos certificados digitais emitidos por várias ICPs distintas.

Veremos primeiramente três arquiteturas que são consideradas básicas e comumente usadas em ambientes pequenos e restritos, e depois partiremos para arqui-

teturas mais complexas, principalmente voltadas para grandes organizações ou ambientes bastante distribuídos e ecléticos.

2.6.1 Autoridade Certificadora Única

Primeiramente, vamos ver a estrutura de Autoridade Certificadora Única - ACU, a qual simplesmente implementa-se com o uso de um conjunto simples dos componentes de uma ICP já vistos, sendo ela a única responsável pela emissão e gerência dos certificados digitais de todos os usuários da arquitetura, conforme a Figura 2.1.

Como só existe uma AC, para se confiar na infra-estrutura é necessário confiar-se na AC. Como todos os certificados são emitidos por uma única AC, para validar um certificado digital, basta estar de posse e confiar no certificado da AC da arquitetura.

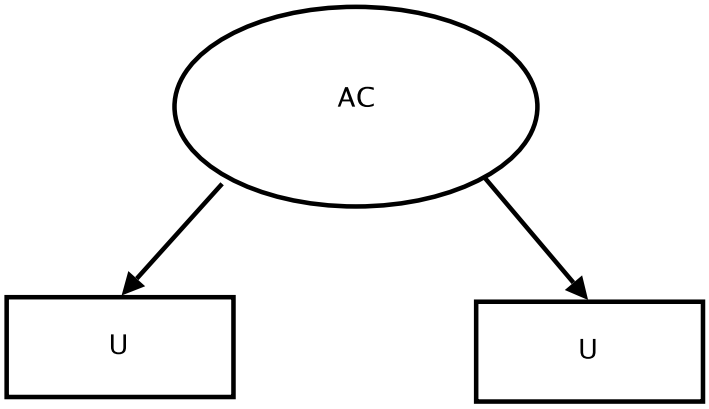


Figura 2.1: Arquitetura de ICP baseada em AC única.

Para Alice confiar no certificado digital de Beto emitido por uma AC, é necessário que Alice tenha e confie no certificado da AC. Para Alice deve ser possível determinar um caminho de certificação, como representado pela figura 2.2. A validação do caminho de certificação consiste em Alice confiar no certificado da AC, atestando de a chave pública Ku_{AC} pertence efetivamente à AC, e a partir dele validar a assinatura existente no certificado da AC e no certificado de Beto.

Esta estrutura é bastante simples e de fácil implementação, mas é bastante vulnerável a um comprometimento da chave da AC. O comprometimento, seja por roubo, perda, ou qualquer outro fator que invalide o uso da chave, torna toda a arquitetura inválida, sendo necessário avisar todos os detentores de certificados e usuários confian-

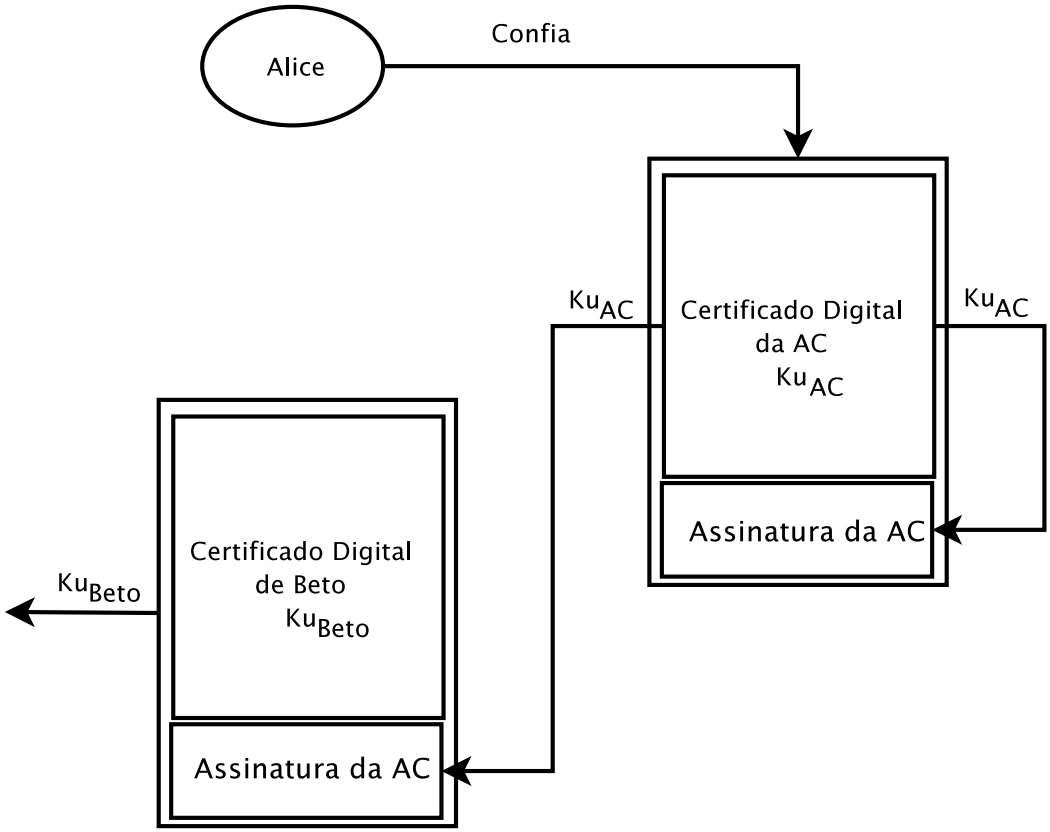


Figura 2.2: Exemplo de validação de um caminho de certificação.

tes que o comprometimento existiu, sendo também necessária a troca da chave e a nova emissão dos certificados tanto da AC quanto de todos os seus usuários registrados.

Nesta arquitetura, para a proteção da chave da AC que controla a arquitetura, deve ser feito uso de Provedores Criptográficos Seguros, garantindo assim uma maior resistência aos vários ataques possíveis.

2.6.2 Listas de Confiança

A evolução natural da arquitetura de Autoridade Certificadora Única é a sua implementação em vários setores ou empresas diferentes, e no caso da necessidade de confiança por parte de um usuário em uma outra ICP, o mesmo deve criar uma Lista de Confiança, assumindo assim como confiáveis os certificados de outras ACs que não a AC que emitiu o seu próprio certificado, conforme a Figura 2.3.

Como podemos ver na figura 2.3, a validação de um certificado digital emitido por uma outra ICP se dá da mesma forma que um certificado emitido pela própria AC do usuário registrado, pois partimos de um ponto de confiança diretamente confiável

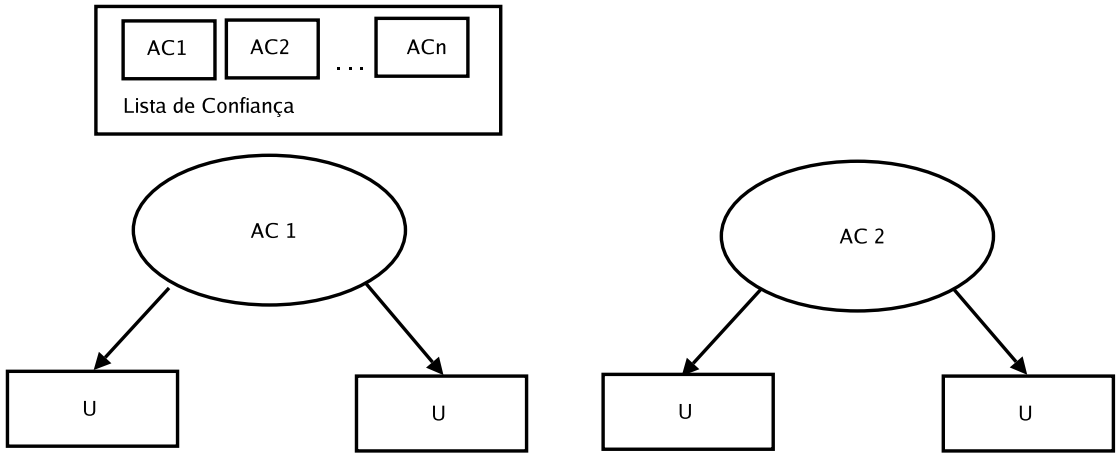


Figura 2.3: Exemplo de arquitetura de ICP baseada em Listas de Confiança.

pelo usuário.

Esta arquitetura apresenta os mesmos problemas de comprometimento de chaves que a arquitetura de Autoridade Certificadora Única, com um agravante: a AC não tem controle de quem confia nela, e no caso de um comprometimento de suas chaves, não pode informar a todos que nela confiam, principalmente os que não têm certificados emitidos por ela, tornando assim estes usuários vulneráveis enquanto eles não revogarem a sua confiança no certificado comprometido.

Nesta arquitetura o comprometimento das chaves de qualquer uma das ACs na qual um usuário confia torna-o vulnerável, tornando evidente que qualquer AC deve ter um considerável nível de proteção de suas chaves, uma vez que as listas de confiança não estão sob controle das ACs.

Uma forma de minimizar os ataques às várias AC componentes da lista de confiança é verificar se as mesmas possuem PSCs na sua estrutura, garantindo assim a resistência delas a ataques sobre a chave. Outra forma de utilizar um PSC neste contexto é armazenando a Lista de Confiança no próprio PSC.

2.6.3 Estrutura Hierárquica

Uma outra evolução da arquitetura de AC Única é a criação de estruturas hierárquicas de ICP, onde temos ACs subordinadas umas às outras, criando uma rede mais complexa de confiança.

Na arquitetura baseada em Estrutura Hierárquica, conforme podemos

observar na Figura 2.4, o usuário confia em um único ponto, o qual é o certificado auto-assinado da AC raiz da árvore da arquitetura. Ao confiar nesta AC raiz, ele passa a confiar em todos os certificados emitidos por ela para usuários e para ACs intermediárias. A confiança nos certificados emitidos por ACs intermediárias independe da vontade do usuário. O usuário necessita unicamente, na hora de validar o certificado, conferir um caminho de certificação mais longo, o qual deve incluir todas as ACs intermediárias que existem entre o certificado a ser validado e o ponto de confiança.

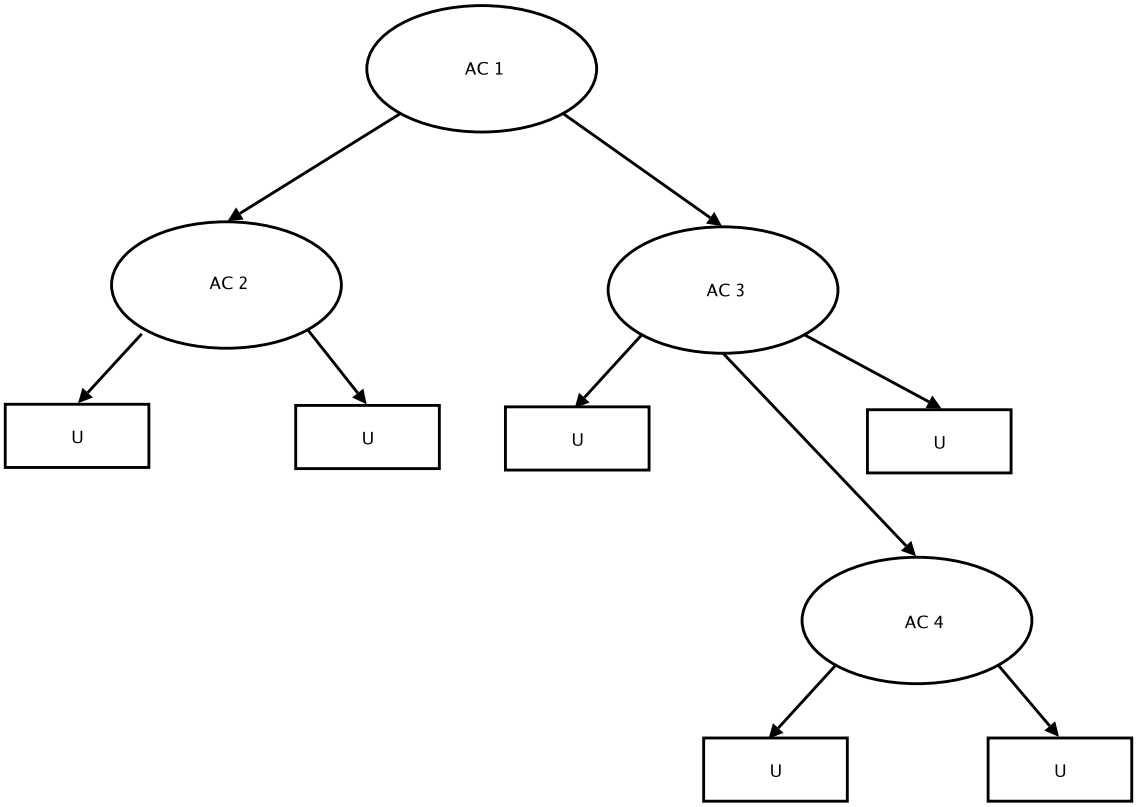


Figura 2.4: Exemplo de uma arquitetura de ICP baseada em Estrutura Hierárquica.

Esta estrutura é bastante comum e bastante utilizada em organizações com uma estrutura hierárquica e independente, pois torna mais fácil de representar a real estrutura de confiança dentro da organização.

Do ponto de vista do usuário, a arquitetura hierárquica agrega um certo grau de complexidade na validação dos certificados de usuários, pois passamos a ter a necessidade de validar também todos os certificados existentes no caminho de certificação que liga o ponto de confiança ao certificado do usuário.

Outro fator que pode levar à utilização de estruturas hierárquicas é a implementação de políticas de certificação, delegando competências de emissão de certi-

ficado com fins específicos a autoridades específicas.

Quanto à segurança das chaves criptográficas das ACs da arquitetura Hierárquica, o comprometimento de qualquer chave invalida todos os certificados emitidos por ACs abaixo do certificado comprometido. No entanto, todas as outras partes da estrutura continuam operando normalmente, e quando ocorre substituição das chaves comprometidas, basta emitir o certificado da AC comprometida e de todos os usuários abaixo dela. Como todos os usuários estão sob o mesmo ponto de confiança, todos podem ser notificados da revogação da confiança dos certificados da parte da estrutura comprometida.

O uso de PSC nesta arquitetura é recomendado em todos os níveis de AC, para minimizar os possíveis ataques às chaves privadas, podendo agora ser observados os níveis qualitativos de garantias que cada PSC pode prover. Em geral, quanto mais próximo à raiz da árvore, maiores são as necessidades de segurança que o PSC deve prover à AC.

2.6.4 Estrutura em Teia

A arquitetura usando Estrutura em Teia normalmente é considerada um boa alternativa para arquiteturas baseadas em estruturas hierárquicas mas com a confiança não diretamente ligada a uma única AC, mas a várias outras. Nela, como podemos ver na Figura 2.5, as relação de confiança não mais dependem do usuário, o qual confia em um único ponto de confiança, mais sim das relações entre as ACs.

As ACs de uma arquitetura com Estrutura em Teia se relacionam diferentemente das ACs de uma Estrutura Hierárquica, pois não existe uma relação direta de descendência, mais sim uma teia de confiança, a qual pode ser ainda unidirecional e em sentido não descendente e não necessáriamente única.

Estes fatores de relacionamento fazem com que a arquitetura usando Estrutura em Teia seja extremamente resistente a problemas de comprometimento de chaves nas ACs da ICP, pois no caso de um comprometimento, dependendo dos relacionamentos da estrutura, somente os usuários da AC comprometida são afetados. Nos piores casos de comprometimento, podemos deixar a estrutura em teia dividida em duas partes, quebrando uma passagem única do caminho de certificação.

Um problema inserido por esta arquitetura é quanto à validação de um

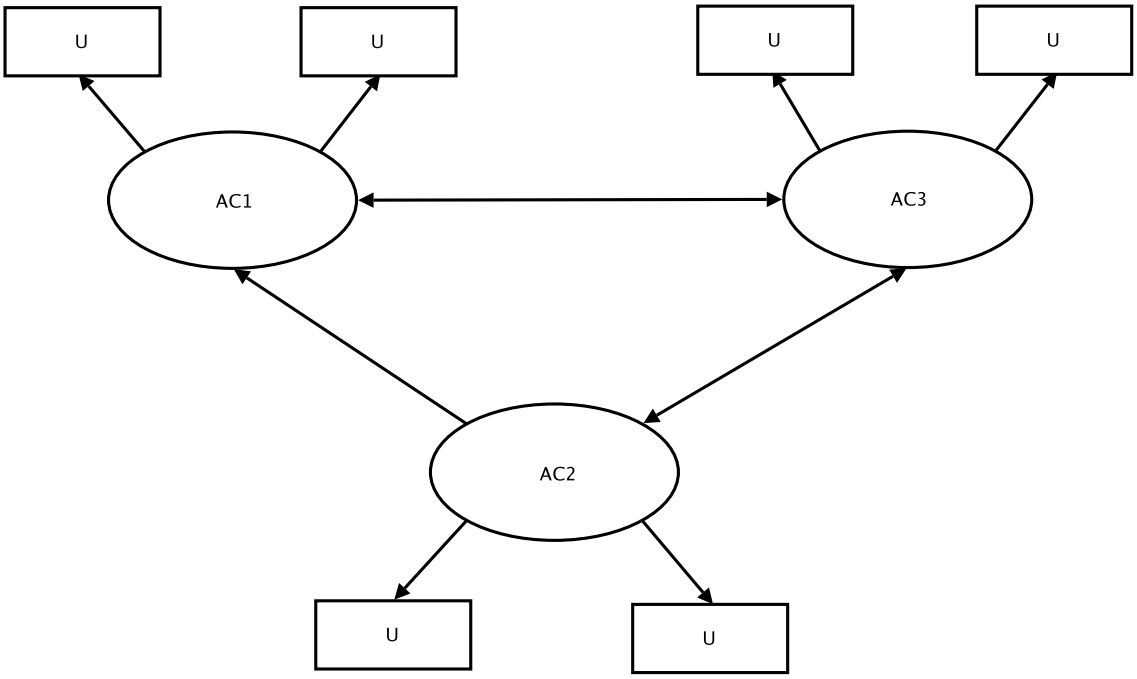


Figura 2.5: Arquitetura de ICP baseada em Estrutura em Teia.

certificado digital por parte de um usuário, pois o mesmo passa a não ter mais um caminho determinístico para esta validação. Este não determinismo faz com que a validação possa chegar em caminhos de certificação inválidos e seja necessário reiniciar o processo até determinar o caminho correto, ou no caso da exaustão dos caminhos, invalidar o certificado.

Uma forma de minimizar os ataques às várias ACs componentes da estrutura em teia é fazer com que as mesmas possuam PSCs na sua estrutura, garantindo assim a resistência delas a ataques sobre a chave. Esta minimização dos ataques tem por objetivo evitar o comprometimento das ACs e manter os caminhos de certificação sempre acessíveis.

2.6.5 Lista Estendida de Confiança

A arquitetura de Lista Estendida de Confiança - LEC é a evolução da estrutura de Lista de Confiança, com o fato de ser acrescentado à lista outras estruturas que não a Estrutura de AC Única, sendo assim considerada uma arquitetura híbrida de ICP.

Como podemos ver na Figura 2.6, ocorrem nas LECs o acréscimo de várias estruturas diferenciadas, podendo elas serem quaisquer arquiteturas de ACs. Com

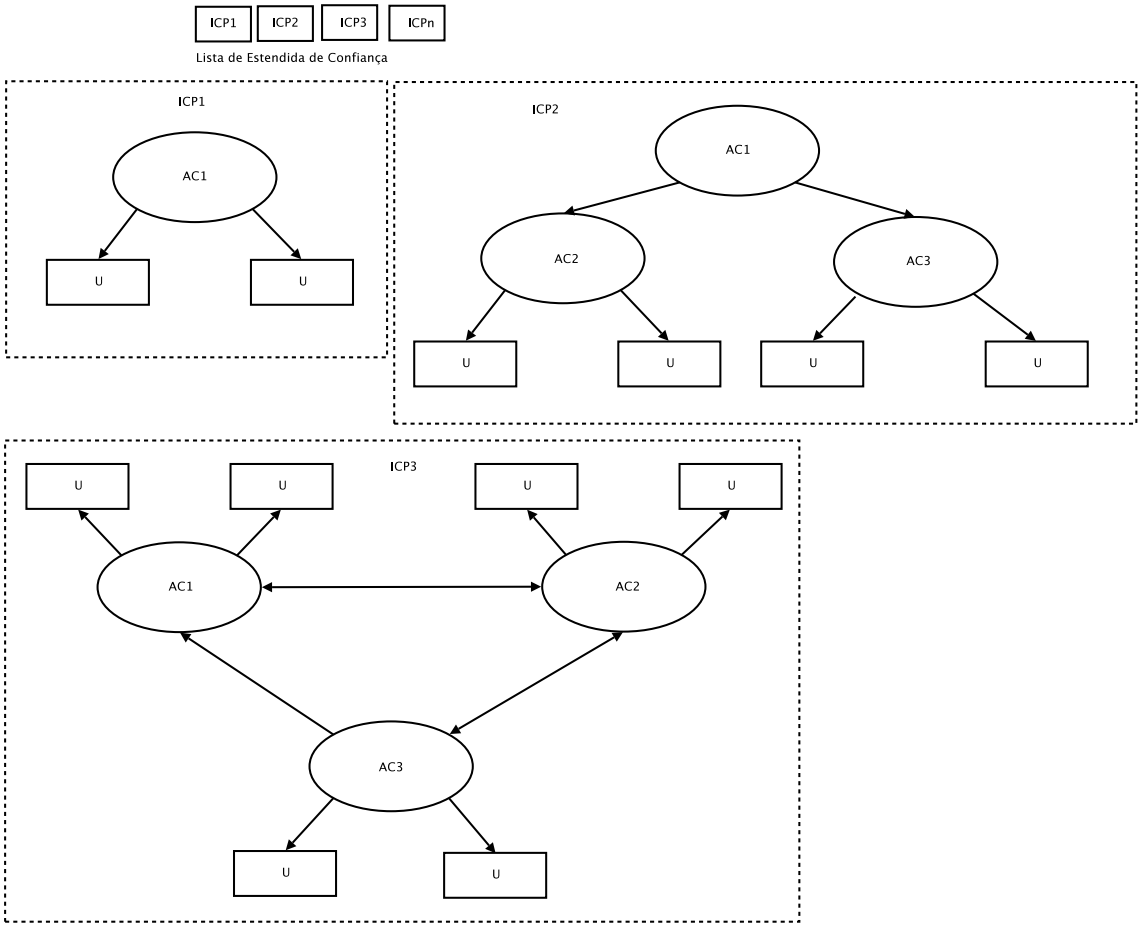


Figura 2.6: Exemplo de arquitetura de ICP baseada em Lista Estendida de Confiança.

esta possibilidade, passamos novamente ao usuário a escolha dos pontos de confiança, fazendo assim com que ele seja responsável por gerenciar a confiança ou a revogação dos certificados nas arquiteturas.

Ao usar uma LEC, um usuário sofre dos mesmos problemas de vulnerabilidades que sofreria com qualquer uma das arquiteturas que fazem partes da sua lista, com o agravante de que fica a seu cargo a revogação da confiança em determinado ponto de uma arquitetura para ele normalmente desconhecida.

Fica evidente que a abrangência das LECs é uma necessidade de evolução das arquiteturas de ICP, para que um usuário possa estabelecer a sua rede de confiança, mas o seu correto e pontual gerenciamento pode ser crítico para a segurança do mesmo. Isso novamente evidencia que qualquer AC componente de qualquer estrutura deve ter como principio básico a proteção de suas chaves, visto que os problemas causados por um comprometimento podem ser imensuráveis e isso pode ser feito adequadamente por um PSC.

2.6.6 Certificação Cruzada

Na arquitetura de Certificação Cruzada, temos a emissão de certificados entre ICPs, estabelecendo assim relações de confiança entre diferentes ICPs, conforme demonstra a Figura 2.7.

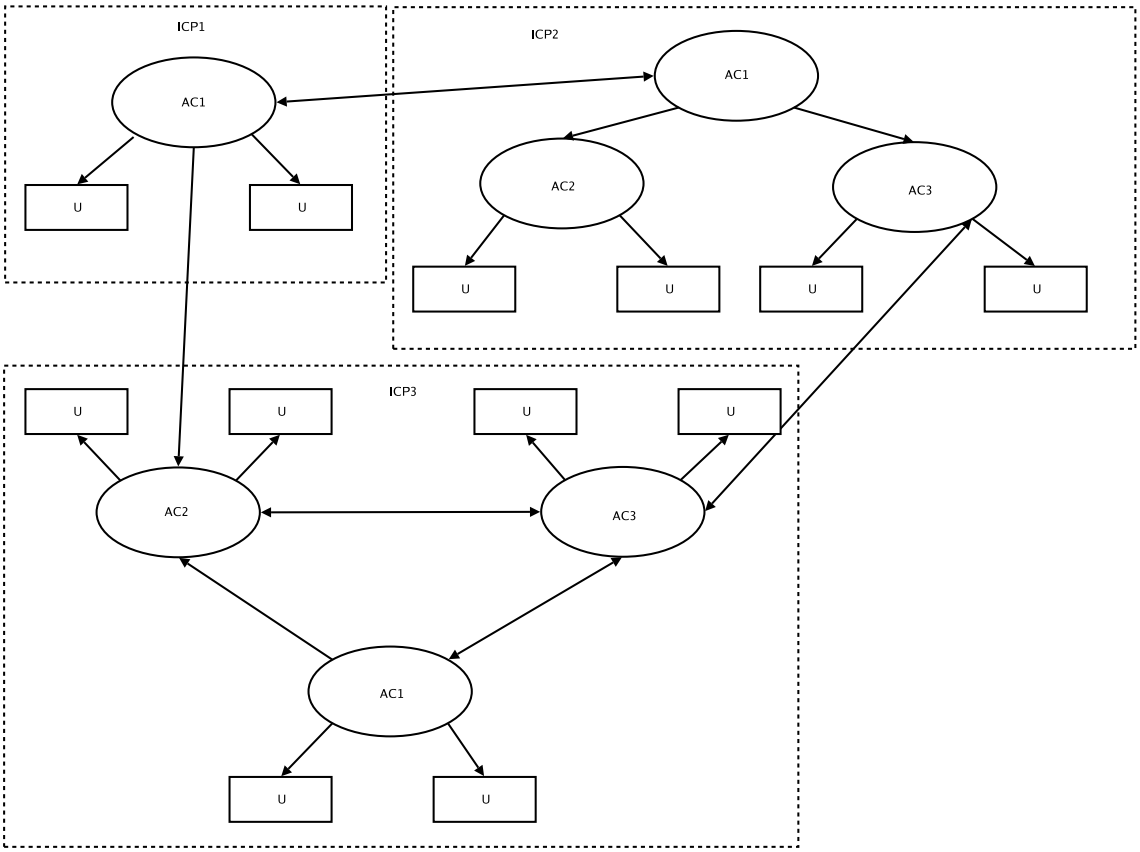


Figura 2.7: Exemplo de arquitetura de ICPs baseada em Certificação Cruzada.

A arquitetura de Certificação Cruzada vem para retirar a carga de decidir os relacionamentos de confiança por parte do usuário com certificados de diferentes ICPs. Esta técnica se assemelha bastante com a de Listas Estendidas de Confiança, mas com a confiança sendo estabelecida pelo Administrador da ICP, e não mais pelo usuário, fazendo assim com que a decisão seja válida para todos os usuários da ICP.

Esta arquitetura também pode ser considerada híbrida, uma vez que as ICPs interconectadas pela certificação cruzada podem variar em sua estrutura básica. Este fator também torna a validação de certificados uma tarefa bastante árdua, uma vez que os algoritmos devem levar em conta todas as arquiteturas de ICP, assim como implementar métodos não determinísticos para determinar o caminho de certificação.

Nesta arquitetura, o comprometimento das chaves de uma AC é tratado

diretamente pelos administradores, e normalmente a confiança é negociada entre eles, ficando assim o aviso do comprometimento mais fácil e transparente para um usuário, o qual só tem um único ponto de confiança. Este fato não isenta uma boa política de proteção de chaves, pois este comprometimento pode fazer com que a necessidade de troca de informação segura entre as ICPs esteja indisponível, o que pode, por exemplo, inviabilizar toda uma estrutura de negócio.

A minimização dos ataques às várias ACs, pode ser feita com o uso de PSCs na sua estrutura, garantindo assim a resistência de todas as ACs a ataques sobre as suas respectivas chaves. A minimização dos ataques evita o comprometimento das ACs e mantém os caminhos de certificação sempre acessíveis.

2.6.7 Certificação em Ponte

A arquitetura de Certificação em Ponte tem como objetivo resolver os problemas estruturais que as arquiteturas de Lista Estendida de Confiança e de Certificação Cruzada possuem. Estes problemas estão basicamente na necessidade da criação de extensas teias de confiança entre ACs e da possibilidade de caminhos de certificação não autorizados entre as ICPs componentes.

Como podemos ver na Figura 2.8, todos os estabelecimentos de confiança entre as ICPs passam por uma AC que é responsável unicamente pela ponte de certificação entre elas e torna-se um árbitro de confiança entre as partes. Desta forma, quando existe o comprometimento de uma ICP componente, uma única alteração deve ser feita pela AC da ponte, a qual desconecta a ICP comprometida da estrutura como um todo.

Nesta arquitetura, novamente desoneramos o usuário de estabelecer pontos de confiança alternativos; a ele basta possuir um ponto de confiança com uma AC de uma ICP.

O uso desta arquitetura não nos isenta dos problemas de validação dos caminhos de certificação, uma vez que esta estrutura é tão complexa quanto uma arquitetura em Teia, podendo ser constituída, também, por várias ICPs em Teia.

O comprometimento das chaves de uma AC nesta estrutura normalmente será tão grave quanto for a abrangência da estrutura de ICP coligada a ponte, tornando somente necessário ao administrador da ICP o seu restabelecimento. Já no caso do

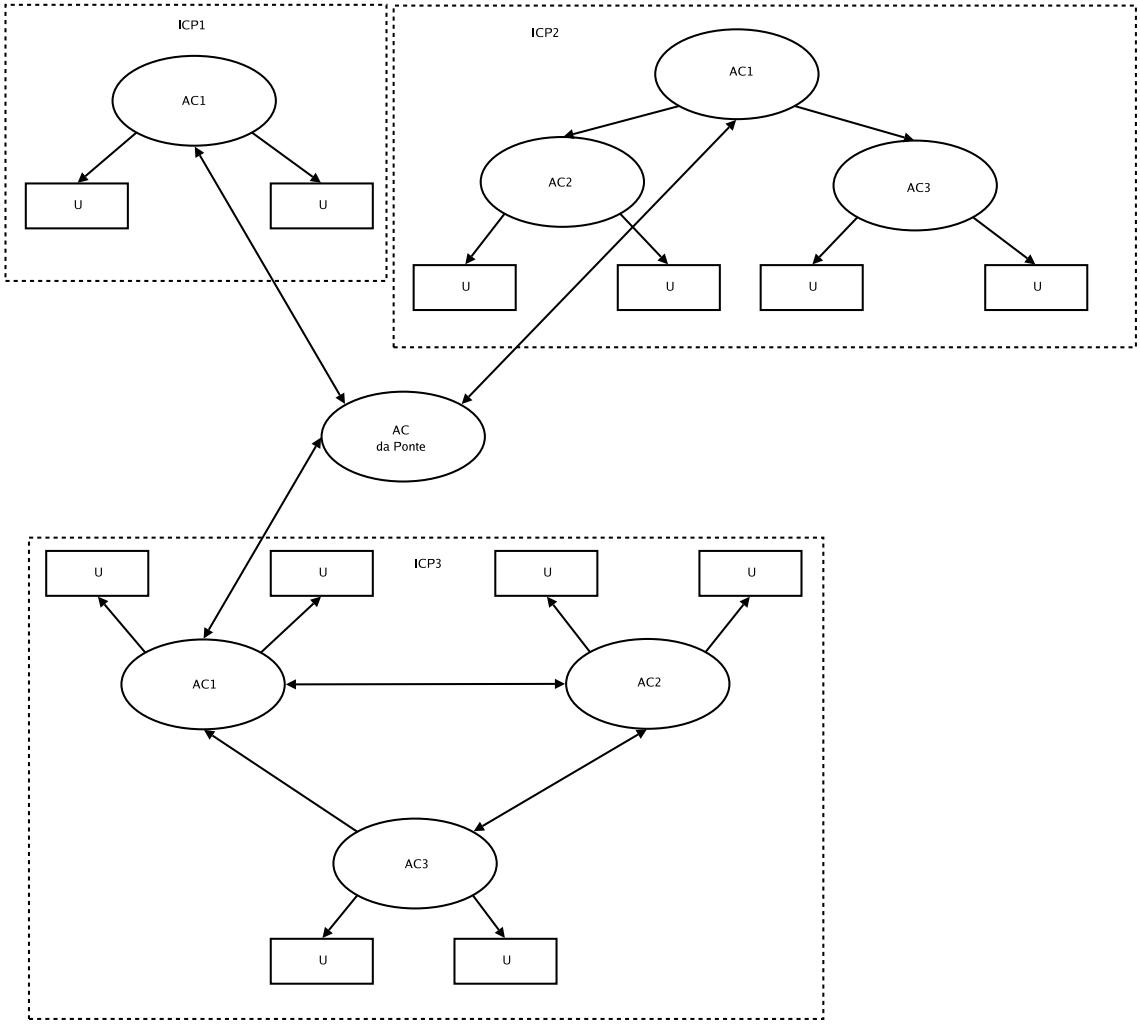


Figura 2.8: Exemplo de arquitetura de ICPs baseada em Certificação em Ponte.

comprometimento da estrutura como um todo, a incumbência de revogação da confiança cabe unicamente ao administrador da AC da ponte.

2.7 Como confiar em um Certificado Digital

Para um usuário confiar em um certificado digital, ele deve primeiramente confiar na AC emitente do certificado. Dependendo da arquitetura, a confiança não precisa ser diretamente estabelecida com a AC emitente, mais sim com qualquer outra AC que possua uma relação de confiança com esta. Confiando neste certificado da AC, devemos sempre estabelecer um caminho de certificação entre o certificado que queremos validar e o ponto de confiança.

Em geral a administração das listas de confiança e o processamento do

caminho de certificação é feito pelo sistema operacional ou pelas aplicações que utilizam os certificados digitais. Eles criam uma base de dados que contém todos os certificados dos pontos de confiança, certificados de usuários e certificados de ACs intermediárias necessários à validação do caminho de certificação, conforme a Figura 2.9.

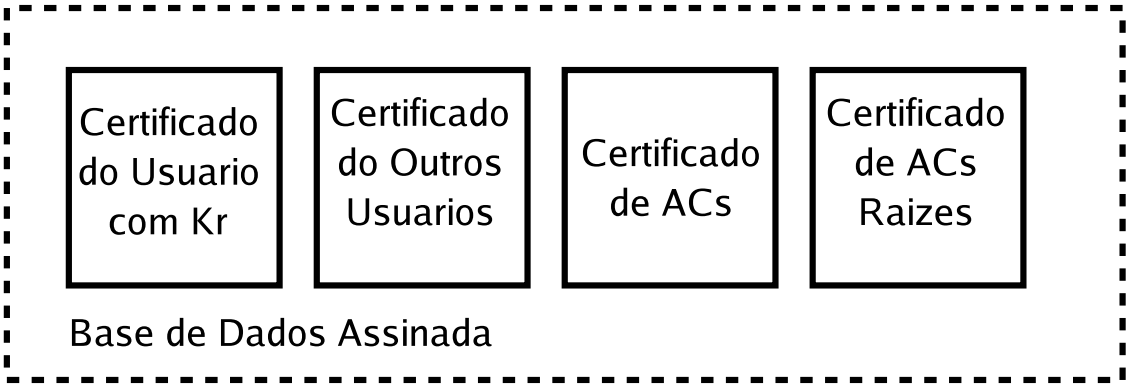


Figura 2.9: Exemplo de Base de Dados de Certificados Digitais.

Cabe também ao sistema operacional ou à aplicação usuária de certificados digitais a gerência e segurança desta base de dados. É preciso salientar que esta base é um dos alicerces necessários à confiança em um certificado digital, e para tal estas aplicações usam técnicas de assinatura digital para garantir a integridade e autenticidade da base.

Ao usar assinaturas digitais, para esta manutenção, é evidenciada a necessidade da proteção das chaves que foram usadas para este processo, deixando assim claro a necessidade da existência de um PSC.

2.8 Uso das Chaves em Infra-estruturas de Chaves Públicas Hierárquicas

Numa infra-estrutura de chaves públicas, temos diferentes necessidades de proteção de chaves e diferentes necessidades de velocidade do uso das mesmas.

Como ilustra a figura 2.10 para o caso particular de uma ICP hierárquica, vemos que, dada a importância de uma Autoridade Certificadora Raiz, seu uso se restringe normalmente à assinatura de certificados de Autoridades intermediárias e à emissão periódica de Listas de Certificados Revogados (LCR). Estas ACs, por serem o

ponto de confiança da estrutura, normalmente tem seu ambiente operacional localizado em locais seguros, tais como salas cofre, os quais provêm um ambiente de acesso restrito e controlado de todos os operadores. Sua necessidade de aceleração criptográfica é mínima, visto que o número de vezes que a chave é usada é pequeno e, mesmo quando usada, o tempo gasto para uso da chave não é importante, mas em contrapartida a segurança das chave é crucial para a segurança da ICP como um todo.

As ACs intermediárias têm um perfil diferente de uso das chaves, visto que estas ACs podem assinar outras ACs e em alguns casos emitem LCRs com mais freqüência que a AC Raiz da ICP em questão. Neste caso, a necessidade de proteção da chaves da AC também é garantido normalmente por ambientes seguros, mas seus requisitos de segurança não são tão fortes quanto os da AC-Raiz, e devem ter provedores que sejam mais rápidos do que os das ACs raízes.

As ACs finais, como seu nome já indica, são autoridades que emitem certificados digitais para usuários finais, podendo emitir milhares, e até milhões de certificados. Estas ACs também são protegidas por ambientes seguros, mas seus requisitos de velocidade criptográfica são muito maiores por outro lado, seu comprometimento afeta somente os certificados dos seus usuários finais. Sendo assim, as necessidades de proteção de chaves, não tão rígidas quanto as das ACs anteriores. Mesmo assim, toda AC deve proteger sua chave privada através do uso de um PSC e de ambientes seguros. Isso se dá uma vez que, na prática, a AC pode ser facilmente inserida nestes ambientes.

Já, usuários finais, precisam da proteção das suas chaves privadas de uma forma muito mais genérica que as ACs, visto que o seu comprometimento é facilmente resolvido (não afeta nenhum outro componente da ICP) e, dependendo dos seus requisitos de mobilidade, o uso de ambientes seguros pode ser inviabilizado. Já, requisitos de velocidade, são extremos, por exemplo, visto que um usuário de certificado digital pode ser uma pessoa que assina 1 documento por mês ou um servidor SSL que necessita responder a mais de 1000 desafios por segundo através do protocolo [9, 22].

Um equipamento PSC tem maior uso pelos usuários finais do que por ACs, visto que eles existem em maior número. Em função disso, os fabricantes de PSCs preocupam-se mais com os requisitos dos usuários finais do que aqueles necessários às ACs. Assim, os PSCs disponíveis no mercado são normalmente aceleradores criptográficos, que podem realizar muitas operações criptográficas por segundo. A característica

de proteção dos parâmetros criptográficos sensíveis é normalmente vista pelos fabricantes como um adicional que pode agregar algum valor ao produto colocado no mercado, uma vez que seu foco é a aceleração criptográfica proporcionada pelo equipamento.

2.9 Conclusões

Como pudemos ver no decorrer deste capítulo, a importância do uso dos métodos de criptografia nos levou à criação de estruturas para o seu gerenciamento. Estas estruturas, por sua vez, adquirem grande complexidade do ponto de vista organizacional.

O ponto de vista da segurança, normalmente o foco de uma ICP, é garantido pelo zelo e boa guarda das chaves de todas as ACs componentes da referida ICP. Este zelo normalmente é obtido através do uso de provedores de serviços criptográficos, os quais são devidamente preparados para resistir a ataques comumente utilizados contra as chaves.

No Brasil, as ICPs criadas à nível nacional, normalmente tem seguido o propostos pelo governo federal [10], e tem feito uso de estruturas hierárquicas, sejam com raíz única, ou com certificação cruzada estabelecidas pelas suas raízes.

Este ponto ajuda a endereçar a aplicação dos estudos desta dissertação, fazendo assim como demonstra a importância da existência de métodos eficazes de proteção e zelo das chaves assimétricas.

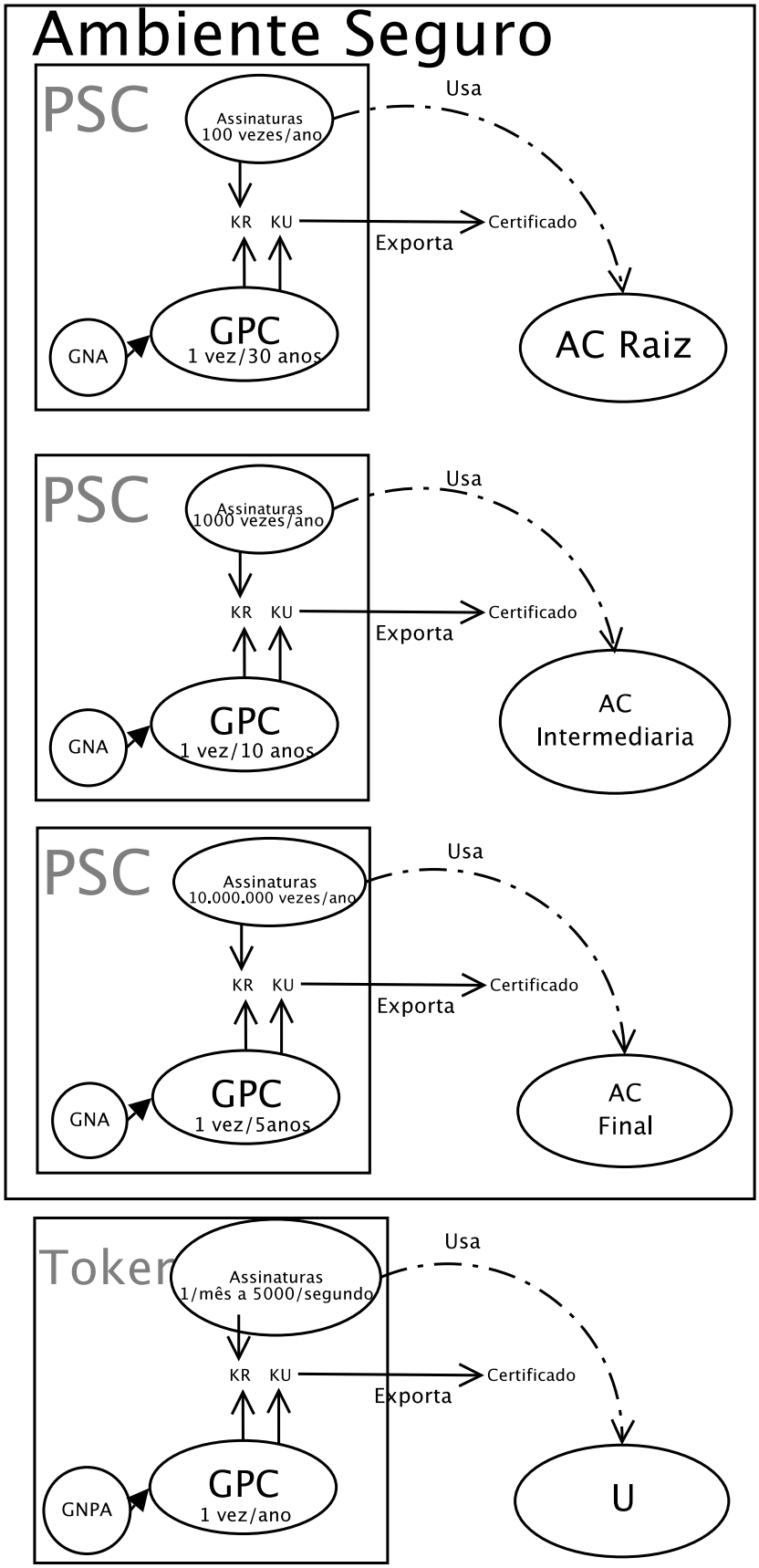


Figura 2.10: Uso das chaves em infra-estruturas de chaves públicas hierárquicas.

Capítulo 3

Normas para Construção de Dispositivos Criptográficos

A construção de mecanismos criptográficos deve normalmente levar em conta os mais variados ataques que estes mecanismos podem sofrer.

Como guia para o desenvolvimento de mecanismos criptográficos baseados em hardware temos a norma FIPS PUB 140-2 [23], estabelecida pelo Instituto Nacional de Padrões e Tecnologia do governo norte-americano, que trata diretamente de características que estes equipamentos devem ter para serem validados num programa de certificação mantido pelo próprio instituto.

Já no desenvolvimento de sistemas voltados para a segurança, temos o estabelecimento de um critério comum internacional, o ISO/IEC 15408 [24–26], também conhecido como Common Criteria, pelos órgãos responsáveis pela padronização e segurança do Canadá, França, Alemanha, Holanda, Inglaterra e Estados Unidos da América. Este critério provê um enquadramento de trabalho para a determinação de objetivos, especificação de requisitos e modelos de implementação para sistemas voltados para a área de segurança da informação.

Na área específica de dispositivos de proteção de chaves criptográficas, os critérios comuns, através da especificação de um perfil de proteção, designam o Perfil de Proteção para Dispositivos de Criação de Assinaturas Seguras e foi estabelecido pelo Comitê Europeu para Padronização/Sistema de Padronização da Sociedade da Informação (CEN/ISSS) [27–29].

O presente capítulo apresenta as normas que regem a indústria que

desenvolve equipamentos criptográficos, dando ênfase às normas FIPS PUB 140-2 e ISO/IEC 15408, as quais são amplamente aceitas por governos e pela inciativa privada.

No decorrer do capítulo, vamos ver o detalhamento da FIPS PUB 140-2, passando por uma visão geral, pelos quatro níveis de requisitos e por condições para a obtenção da certificação. Prosseguindo vamos tratar a ISO/IEC 15408, explicando conceitos gerais, mostrando as famílias de funcionalidades de segurança e de garantias de avaliação, detalhando os níveis de avaliação e correlacionando as duas normas.

3.1 FIPS PUB 140-2

A FIPS PUB 140-2 é uma publicação do NIST, Instituto Nacional de Padrões e Tecnologia do governo norte-americano, e trata dos requisitos de segurança para módulos de hardware criptográfico. Esta publicação substitui a anterior do mesmo tema, chamada FIPS PUB 140 [30].

Nesta publicação, o NIST especifica os requisitos que devem ser satisfeitos para que um módulo de hardware criptográfico a ser usado para processamento de informação do governo norte-americano, seja aprovado e classificado para uso de acordo com a sensibilidade da informação por ele protegida.

A FIPS PUB 140-2 classifica os módulos de hardware criptográfico em quatro níveis crescentes de segurança. Os requisitos necessários a cada nível são divididos em várias áreas [23], conforme a Tabela 3.1:

Cada nível tem requisitos diferentes para cada uma das áreas, e para suprir os requisitos de um determinado nível é necessário que se cumpram os requisitos dos níveis anteriores.

3.1.1 Nível 1

Este é o nível mais básico de segurança que pode ser propiciado por um módulo de hardware criptográfico. Os requisitos são bastantes elementares, sendo necessário apenas que ele opere com algoritmos e funções já previamente aprovadas e publicadas em outras FIPS PUB, não sendo necessário nenhum requisito de segurança física a não ser o fato de que os componentes devem ser amplamente disponíveis no mercado. Neste nível, todo o software e componentes de firmware podem ser executados em

Tabela 3.1: Requisitos analisados na conformidade com a FIPS 140-2

Área:	**Descrição:**
Especificação do módulo	Inclui a especificação de todos os componentes de software, hardware e firmware, além da definição da política de segurança do módulo.
Portas e interfaces de comunicação	Especificação de todas as interfaces e de todos os caminhos de dados de entrada e saída, incluindo requisitos de separação de caminhos de dados.
Papeis, Serviços e Autenticação	Especificação dos papéis e serviços desempenhados durante a operação do módulo, assim como os mecanismos de autenticação para acesso a eles.
Máquina de estados finitos	Deve existir a definição formal através de uma máquina de estados finitos de todos os estados alcançáveis pelo módulo, incluindo erros, testes e atividades de manutenção.
Segurança física	Devem haver mecanismos para prevenir modificação e acesso não autorizado ao módulo, garantindo a integridade e segurança de todos os parâmetros de segurança por ele protegidos.
Ambiente operacional	Devem existir definições que garantam que a operação do módulo vai ocorrer da forma esperada, estabelecendo requisitos no ambiente operacional no qual o mesmo vai estar inserido.
Gerenciamento de chaves	Especificação de mecanismos para o gerenciamento do ciclo de vida completo das chaves protegidas pelo módulo, o que inclui geração de chaves, armazenamento, entrada e saída, e a sua devida eliminação
Interferência e compatibilidade eletromagnética	Devem ser especificados mecanismos para evitar o vazamento e a interferência eletromagnética entre o módulo e outros equipamentos que possam vir a ser operados no mesmo ambiente que o mesmo.
Segurança de Projeto	Deve-se demonstrar que foram utilizadas as melhores práticas e técnicas para garantir que o módulo foi desenvolvido de acordo com as especificações apresentadas.
Contenção de ataques	Deve constar em projeto mecanismos para mitigar ataques que por ventura possam vir a existir, através do monitoramento do ambiente no qual o módulo se encontra inserido.

qualquer computador, não sendo necessário o uso de um sistema operacional comprovadamente seguro. Módulos enquadrados neste nível são adequados a ambientes onde o custo da informação protegida não é muito alto, ou em ambientes em que não se exigem normatizações severas quanto aos procedimentos administrativos.

3.1.2 Nível 2

O nível 2, estabelecido pela FIPS PUB 140-2, aumenta os requisitos do nível anterior, principalmente nos quesitos de segurança física, incluindo requisitos para a detecção de acesso físico ao dispositivo, tais como revestimentos com detecção de

violação, ou travas mecânicas nas vias de acesso ao interior do equipamento.

Ainda como requisito deste nível, temos a necessidade de autenticação baseada em papeis, na qual o módulo autentica o operador para assumir um determinado papel e o libera para executar um conjunto de operações de acordo com o seu papel.

Todo o software criptográfico, chaves, parâmetros críticos de segurança e informação de controle de estado devem estar sob o controle de um sistema operacional avaliado como nível 2 ou superior do critério comum de avaliação de segurança[1], ou de um sistema operacional equivalente em segurança.

3.1.3 Nível 3

Em adição aos requisitos do nível 2, a segurança física deste nível é mais elaborada, tentando evitar que algum invasor tenha acesso a Parâmetros Críticos de Segurança (PCSs); os mecanismos de segurança tentam evitar e responder às tentativas de invasão.

Os principais mecanismos presentes são: invólucros resistentes a invasão e com detecção de intrusão; circuitos zeradores de dados, que apagam todos os parâmetros críticos quando um módulo é aberto; mecanismos de identificação baseados na identidade, para que posteriormente sejam assumidos papéis; a necessidade da separação dos caminhos de dados usados para entrada e saída de dados do módulo de forma física ou lógica e todos os dados que entram ou saem do módulo devem estar na forma cifrada.

Todo o software criptográfico, chaves, parâmetros críticos de segurança e informação de controle de estado devem estar sob o controle de um sistema operacional avaliado como nível 3 ou superior do critério comum de avaliação de segurança[2] incluidas as características adicionais de caminhos seguros[3] e um modelo de política de segurança[4]. O sistema operacional pode ser um equivalente em termos de segurança.

---

[1] EAL2
[2] EAL3
[3] $FTP\_TRP.1$
[4] $ADV\_SPM.1$

3.1.4 Nível 4

O nível 4 provê o maior grau de segurança em módulos de hardware criptográfico. Neste nível, os mecanismos de segurança provêem um envelope completo de proteção ao módulo de cifragem, com o intuito de detectar e reagir a quaisquer tentativas de acesso não autorizado ao módulo.

A penetração no módulo deve sempre habilitar o circuito zerador, para que nenhum parâmetro crítico de segurança seja acessado. Os módulos com este nível de segurança são comumente recomendados para ambientes onde não existem possibilidades de segurança física.

Os módulos deste nível devem ser capazes de detectar mudanças ambientais e flutuações do ambiente externo, e, em determinados níveis de flutuação do ambiente, chamar os circuitos zeradores de dados. Também é necessário que o módulo realize testes rigorosos de suas condições internas para que nada seja comprometido internamente por variações externas ao módulo.

Todo o software criptográfico, chaves, parâmetros críticos de segurança e informação de controle de estado devem estar sob o controle de um sistema operacional avaliado como nível 4 ou superior do critério comum de avaliação de segurança[5], ou de um sistema operacional equivalente em segurança.

3.1.5 Aprovação de Conformidade

Como condições a todos os níveis de segurança especificados, tem-se a necessidade da total especificação do módulo para que o mesmo possa ser avaliado pelo NIST, e seja atestado em um determinado nível de segurança. Nestas especificações devem constar as implementações de software, hardware, e firmware, assim como as definições das barreiras criptográficas destes componentes. Devem ser especificadas todas as portas e modos de operação que elas podem assumir, os controles lógicos e manuais, os controles de estado, e as características físicas, lógicas e eletrônicas de todo o equipamento.

A documentação deve prover uma máquina finita de estados com todos os possíveis estados assumidos pelo módulo, organizados em classes [23], conforme mostra a Tabela 3.2

[5]EAL4

Tabela 3.2: Possíveis estados assimidos pelo módulo

Classe:	**Descrição:**
Inicialização/finalização	Estados que serão atingidos durante os processos de inicialização e finalização do módulo criptográfico.
Inicialização criptográfica	Estados onde as operações criptográficas são efetuadas, assim como a sua efetiva inicialização.
Uso comum	Estados nos quais os usuários podem obter serviços do módulo.
Auto-teste	Estados nos quais o módulo está se auto testando contra falhas ambientais e de procedimentos internos.
Erros	Estados onde o módulo possa encontrar erros que proíbam a sua correta operação.
Repasse	Estados onde é possível a operação do módulo sem a necessidade de processamento criptográficos
Manutenção	Estados necessários para qualquer tipo de manutenção no equipamento.

Um ponto importante para a aprovação de conformidade do módulo com o programa de validação da FIPS PUB 140-2 é quanto ao gerenciamento de chaves criptográficas, o qual deve abranger todo o ciclo de vida das chaves, e quaisquer outros componentes criptográficos relativos ao módulo.

O gerenciamento de chaves inclui a especificação do gerador de números aleatórios, o gerador de chaves, o estabelecimento de chaves, a distribuição de chaves, a entrada e saída de chaves, o armazenamento de chaves e a eliminação de chaves. Todos os processos relativos às chaves e a cifragem e decifragem de parâmetros críticos de segurança devem ser feitos usando algoritmos previamente aprovados. Os textos cifrados por métodos não aprovados são considerados textos claros no processo de obtenção do nível de conformidade.

Os geradores de números aleatórios (GNA) devem passar por testes de aleatoriedade especificados na norma. O módulo deve possuir métodos de teste na inicialização e no seu desligamento, de forma a garantir que a semente aleatória não seja reaproveitada. O gerador de chaves deve fazer uso de um gerador de números aleatórios aprovado e deve fazer todo o processo de geração de chaves internamente. Como requisito, a determinação de uma chave gerada por um módulo criptográfico deve ser sempre mais custosa que a busca exaustiva da chave no espaço de chaves. Se o módulo liberar alguma forma intermediária de chave para o mundo externo, esta liberação deve ser feita sempre na forma cifrada ou então na forma de segredo compartilhado [31].

A troca de chaves ou o estabelecimento das mesmas pode ser feita de forma automatizada, ou de forma manual, ou ainda por um método intermediário que use os dois processos. A entrada e saída de chaves nos módulos criptográficos deve ser sempre feita na forma cifrada para chaves privadas e sementes aleatórias, e para os níveis 3 e 4 a proteção da chave privada enquanto fora do módulo deve ser feita através de segredo compartilhado, e a re-entrada de uma chave em um módulo deve ter todo o seu processo validado e descrito na documentação. O armazenamento das chaves nos módulos criptográficos pode ser feito tanto na forma cifrada quanto na forma aberta, mas devem ser considerados parâmetros críticos de segurança (PCS) sempre que estiverem na forma aberta, ou seja, devem sofrer a ação do circuito zerador sobre quaisquer tentativa de invasão. O circuito zerador do módulo criptográfico não deve zerar a chave criptográfica na forma cifrada.

Por fim, os módulos aprovados pelo Programa de Validação de Módulos Criptográficos [32] tem que ter implementados e descritos na sua documentação, métodos de auto-teste em conformidade com os explanados na norma, assim como teste de conformidade do ambiente para o caso dos módulos dos níveis 3 e 4.

3.2 Critérios Comuns - ISO/IEC 15408

Os critérios comuns para avaliação de segurança da informação, também conhecidos como CC [24–26], formam uma estrutura para definição de requisitos de segurança para produtos voltados para a área de segurança da informação.

Definição - Perfil de Proteção - Um conjunto de requisitos de segurança independente de implementação feito para uma categoria de produtos para avaliação. Seu foco esta nas necessidades específicas dos consumidores destes produtos.

Definição - Alvo de Segurança - Um conjunto de especificações e requisitos que é usado como base de avaliação de um produto.

Definição - Componente - O menor conjunto de elementos que pode ser incluído em um Perfil de Proteção ou um Alvo de Segurança.

Esta estrutura é composta de Perfis de Proteção, Alvos de Segurança e de componentes que proverão as Funcionalidades de Segurança ou darão as Garantias de

Compatibilidade com os níveis de avaliação de garantias, EAL[6].

Os componentes de segurança funcional expressam os requisitos de segurança para as ameaças no ambiente operacional do produto alvo da avaliação. Os componentes de garantia devem garantir os projetos e execução de produtos alvo de avaliação. Os níveis de garantia foram criados para servir de critério para a avaliação de perfis de proteção e de alvos de segurança.

Os EAL por sua vez determinam o grau de garantia que teremos no desenvolvimento e uso de produtos voltados para a segurança da informação, através de uma análise metódica e bem definida do projeto, dos testes e das revisões necessárias às funções de segurança dos produtos alvo de avaliação.

Os componentes são distribuidos formando conjuntos. A estes conjuntos chamamos: Classes, Famílias e Pacotes. As classes são grupos de famílias que compartilham um foco em comum. As famílias por sua vez, agrupam componentes que compartilham os mesmos objetivos de segurança, mas diferem em ênfase e rigor. Já os pacotes são conjuntos de componente reutilizáveis combinados para satisfazer um conjunto de objetivos de segurança.

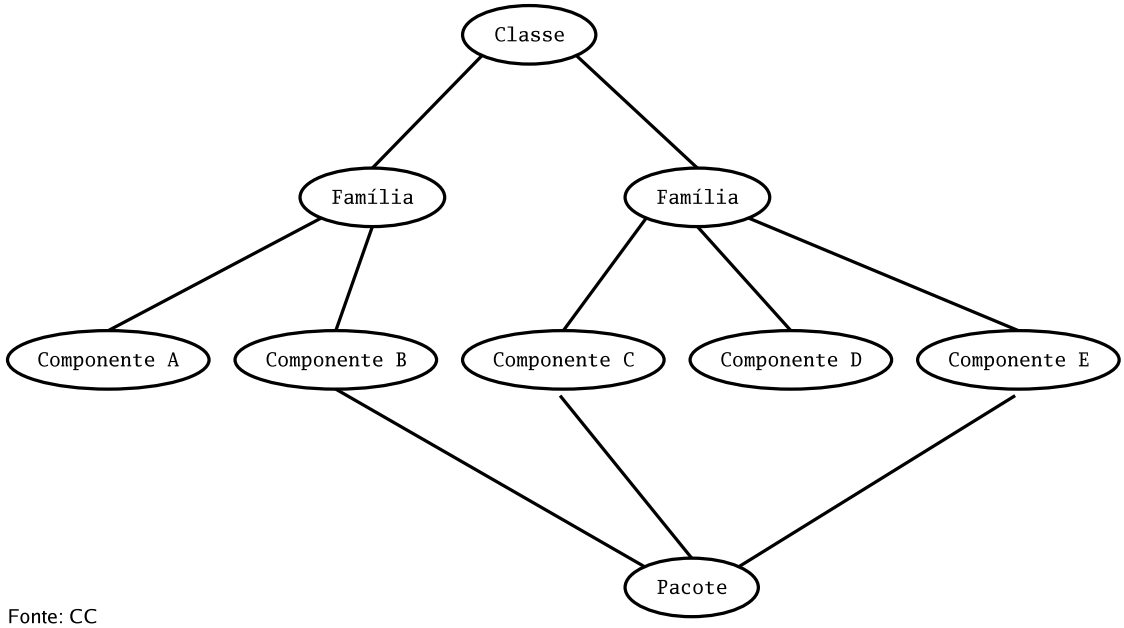
Para denominar uma classe de componentes, usamos três letras seguidas de um traço inferior, e mais três letras que representarão o nome da família, seguidas de um ponto e um número que indica a hierarquia do componente dentro da família, tal como o componente de caminho seguro, denominado *FTP_TRP.1*. Quando queremos representar uma classe como um todo normalmente usa-se as três primeiras letras, por exemplo *FTP*.

Para um melhor entendimento, a Figura 3.1 mostra os relacionamentos entre as entidades definidas pelo CC. Ela mostra a a composição dos pacotes, famílias e classes de famílias de forma hierárquica, conforme especificado pela CC [24–26]

3.2.1 Funcionalidades de Segurança

Os critérios comuns na sua segunda parte [25] especificam um catálogo de componentes de segurança funcional, os quais são usados com base na especificação dos perfis de proteção e alvos de segurança como requisitos funcionais de segurança, e eles descrevem como deve ser o comportamento esperado de um produto alvo de avaliação

[6]Evaluation Assurance Level



Fonte: CC

Figura 3.1: Relacionamento entre as entidades do CC.

para cumprir os objetivos dos perfis de proteção.

Os componentes de segurança funcional respondem aos requisitos de segurança para as ameaças no ambiente operacional do alvo de avaliação e para cobrir políticas de segurança já identificadas. Estes componentes são um conjunto ordenado de elementos funcionais e são agrupados em famílias com objetivos comuns e classes com intenções em comum, podendo existir uma relação hierárquica entre eles.

Os componentes podem ser estendidos, através de uma interface de aplicação - API definida pelos critérios comuns, simplesmente aproveitando a estrutura atual e colocando novos mecanismos dirigidos a um problema especifico, mas estas extensões não são consideradas oficiais e para sua avaliação segundo o CC, deve-se requerer a aprovação prévia por parte de um comitê.

As classes de componentes de funcionalidades de segurança são as apresentadas na Tabela 3.3:

Vamos agora dar uma breve descrição e listar cada uma das classes de funcionalidades.

Tabela 3.3: Classes de componentes de funcionalidades de segurança.

Sigla:	Descrição:
FAU	Componentes de Auditoria
FCS	Componentes de Suporte Criptográfico
FCO	Componentes de Comunicação
FDP	Componentes de Proteção de Dados de Usuários
FIA	Componentes de Identificação e Autenticação
FMT	Componentes de Gerenciamento de Segurança
FPR	Componentes de Privacidade
FPT	Componentes de Proteção de Funções Confiáveis
FPT	Componentes de Separação de Domínios
FRU	Componentes de Utilização de Recursos
FTA	Componentes de Acesso ao Alvo de Avaliação
FTP	Componentes de Caminhos e Canais Seguros

3.2.1.1 Componentes de Auditoria (FAU)

Os componentes da classe de auditoria são componentes que envolvem atividades de reconhecimento, captura, guarda e análise das informações relacionadas com as atividades de segurança.

Os registros de auditoria são gerenciados pelos componentes das famílias desta classe desde sua geração até sua análise, e nesta classe encontramos componentes que definem os requisitos para a seleção de que eventos auditar, a análise dos eventos de auditoria, a proteção dos eventos de auditoria e o correto armazenamento dos mesmos.

3.2.1.2 Componentes de Suporte Criptográfico (FCS)

Os componentes desta classe são responsáveis pela aplicação de funcionalidades criptográficas para ajudar a satisfazer os objetivos de segurança, os quais incluem: identificação, autenticação, não repúdio, caminhos e canais seguros, e separação de dados.

A classe é composta de duas famílias, uma responsável pelo gerenciamento de chaves e outra para suporte de operações criptográficas.

3.2.1.3 Componentes de Comunicação (FCO)

Esta classe de componentes possui unicamente duas famílias que estão diretamente ligadas a identificação das partes em uma comunicação de troca de dados.

A primeira família é responsável por garantir a identidade do originador

de uma transmissão, garantindo o não repúdio de envio por sua parte. A segunda família é responsável por garantir a identidade do destinatário de uma transmissão, garantindo assim a entrega da mensagem e o não repúdio do recebimento.

3.2.1.4 Componentes de Proteção de Dados de Usuários (FDP)

A classe de componentes de proteção de dados de usuários especifica as funções de segurança e as políticas de proteção de dados de usuários. Ela é dividida em quatro grupos de famílias que cuidam da segurança de dados de usuários durante a importação, exportação, armazenamento e alteração de atributos dos dados.

Os quatro grupos de famílias são assim classificadas:

Famílias de funções de políticas de proteção de dados de usuários - É responsável pela definição das políticas relacionadas com os dados de usuários e pela definição do escopo de controle da política necessária para endereçar a segurança dos dados;

Famílias de formas de proteção de dados de usuários - É responsável pela funções de controle de acesso, funções de controle do fluxo de informação, transferências internas, proteção de informação residual, e controle de integridade;

Famílias de funções de importação, exportação e armazenamento - É responsável pela autenticação dos dados, exportação para o exterior e importação do exterior para o alvo de avaliação;

Famílias de funções de comunicação inter-funções - É responsável pela comunicação segura entre todas as funções da classe de proteção de dados de usuários e pela comunicação com outros produtos certificados.

3.2.1.5 Componentes de Identificação e Autenticação (FIA)

As famílias nesta classe de componentes endereçam os requisitos para funções que estabelecem e verificam a identidade de um determinado usuário, e requerem que estes usuários estejam associados a atributos de segurança corretos. Elas lidam com o problema da correta identificação e com a correta definição de atributos para os usuários do alvo de avaliação.

A identificação de usuários e a correta definição de atributos para usuários autenticados é primordial para o funcionamento de qualquer política de segurança.

3.2.1.6 Componentes de Gerenciamento de Segurança (FMT)

Esta classe de componentes especifica o gerenciamento de vários aspectos relacionados com os dados, atributos e funções de segurança do alvo de avaliação, assim como o gerenciamento de papéis e sua interação.

Nas famílias desta classe, temos o gerenciamento de dados das funções de segurança, gerenciamento de atributos de segurança, tais como listas de controle de acesso ou lista de funcionalidades, o gerenciamento de funções de segurança e a definição dos papéis de segurança no alvo de avaliação.

3.2.1.7 Componentes de Privacidade (FPR)

A classe de componentes de privacidade é responsável por prover proteção para os usuários contra o descobrimento ou mau uso da identidade por outros usuários.

Esta classe está dividida em quatro famílias que provém:

Anonimato - A identidade do usuário não é revelada para os processos ou funções de segurança, sendo garantida a sua autenticidade;

Pseudônimo - O usuário pode usar um recurso ou serviço sem liberar sua identidade, mas mesmo assim o uso é contabilizado para si;

Não Ligação - O usuário pode usar múltiplos recursos simultaneamente sem revelar este uso para outros;

Inobservabilidade - Permite ao usuário utilizar um recurso ou processo sem que outros possam ver que o recurso está em uso.

3.2.1.8 Componentes de Proteção de Funções Confiáveis (FPT)

A classe de componentes de proteção de funções confiáveis provê os requisitos funcionais que relacionam a integridade e o gerenciamento de mecanismos das funções de segurança e dos dados das funções de segurança do alvo de avaliação.

Em alguns casos os componentes das famílias desta classe podem parecer confusos e duplicados com os da classe de proteção de dados de usuários, mas os últimos focam unicamente os dados do usuários, enquanto os primeiros focam os dados das funções de segurança.

Do ponto de vista desta classe existem três porções significantes das funções de segurança do alvo de avaliação: a máquina abstrata das funções de segurança, a implementação das funções de segurança e os dados das funções de segurança.

3.2.1.9 Componentes de Separação de Domínios (FPT)

Os componentes das famílias desta classe garantem que sempre existirá pelo menos um domínio disponível para a execução das funções de segurança e que estas funções de segurança estão protegidas de interferência e violação externa através de objetos não confiáveis.

A satisfação dos requerimentos desta classe garante que as funções de segurança do alvo de avaliação são auto protegidas. Assim uma interferência externa no conjunto de funções de segurança não pode modificar ou danificar as funções de segurança do alvo de avaliação.

Na criação de domínios temos a separação do conjunto de funções de segurança, tornando-as inobserváveis e inalteráveis de fora do domínio, desta forma passam a existir controles de entrada e saída de dados entre os domínios.

3.2.1.10 Componentes de Utilização de Recursos (FRU)

Esta classe de componentes provê três famílias para dar suporte à disponibilidade de recursos, quando os mesmo são solicitados.

A família de tolerância a falhas provê proteção contra indisponibilidade de funcionalidades causadas por falhas do alvo de avaliação. A família de priorização de serviços garante que os recursos serão alocados de forma prioritária para as tarefas mais importantes e pontuais e que não podem ser monopolizados por tarefas de mais baixo nível. Por fim, a família de alocação de recursos provê limites no uso de recursos por parte de usuários ou procedimentos, fazendo assim com que eles não possam monopolizar recursos.

3.2.1.11 Componentes de Acesso ao Alvo de Avaliação (FTA)

Esta classe de componentes especifica famílias que controlam o estabelecimento de sessões de usuários, garantindo o escopo de atributos selecionáveis, o controle de sessões concorrentes, o travamento de sessões, os avisos de acesso, os históricos de acesso e o estabelecimento da sessão.

3.2.1.12 Componentes de Caminhos e Canais Seguros (FTP)

As famílias nesta classe servem para a construção de caminhos seguros de comunicação entre usuários e funções de segurança, e para o estabelecimento de canais de comunicação seguros entre as funções de segurança e outros produtos certificados.

Os canais e caminhos seguros possuem as seguintes características:

- Os caminhos de comunicação são construídos usando canais de comunicação que isolam um subconjunto identificado de dados e comandos das funções de segurança do alvo de avaliação;
- O uso de caminho de comunicação pode ser iniciado pelo usuário ou pelas funções de segurança;
- Os caminhos de comunicação são capazes de prover garantias de que o usuário está se comunicando com a função correta e vice-versa.

  Assim, um canal seguro é um canal de comunicação que pode ser iniciado por quaisquer lados da comunicação e provê características de não repúdio da identidade dos dois lados do canal. Um caminho seguro provê os meios para um usuário interagir com o alvo de avaliação de forma garantida e direta. As respostas dos usuários através de um caminho seguro são garantidas contra modificação e revelação a terceiras partes não confiáveis ao alvo de avaliação.

3.2.2 Componentes de Garantia

Os componentes de garantia são as famílias e classes de componentes que ajudam a garantir o projeto e a execução de alvos de avaliação. As ameaças de segurança e a efetivação de políticas de segurança organizacionais devem ser bem articuladas e as medidas de segurança tomadas devem demonstrar suficiência para os seus propósitos.

Estes componentes vão ser base para a construção dos níveis de valiação de garantias, os EAL, os quais servem para qualificarmos produtos de segurança da tecnologia da informação através do cumprimento dos requisitos de segurança funcional e de garantias. Eles provêm requisitos para que tenhamos garantias de gerenciamento de configuração, de desenvolvimento, de entrega e operação, de documentação do ciclo de vida, de teste e de vulnerabilidades, tornando assim o produto, que é o alvo de avaliação, muito mais robusto e factível de certificação.

As classes de componentes de garantia são as apresentadas na Tabela 3.4:

Tabela 3.4: Classes de requisitos de garantias.

Sigla:	**Descrição:**
ACM	Garantias de Gerenciamento de Configuração
ADO	Garantias de Entrega e Operação
ADV	Garantias de Desenvolvimento
ACM	Garantias de Documentação
ALC	Garantias de Suporte do Ciclo de Vida
ATE	Garantias de Testes
AVA	Garantias de Avaliação de Vulnerabilidades

3.2.2.1 Garantias de Gerenciamento de Configuração (ACM)

A classe de garantias de gerenciamento de configuração nos ajuda a garantir que a integridade do alvo de avaliação sempre será preservada, requerendo disciplina e controle nos processos de refinamento e modificação.

A gerência de configuração nos ajuda a prevenir modificações, adições e apagamentos do alvo de avaliação, garantindo assim que a documentação de avaliação é a documentação final do produto. Ela está dividida em famílias de:

gerência de automação - responsáveis por automatizar o controle usado nos ítens de configuração.

funcionalidades de gerenciamento de configuração - responsáveis por definir as características do sistema de gerenciamento de configuração

escopo do gerenciamento de configuração - responsáveis por indicar os ítens que devem ser controlados pelo gerenciamento de configuração.

3.2.2.2 Garantias de Entrega e Operação (ADO)

Os componentes desta classe de garantias definem os requisitos para medidas, procedimentos e padrões voltados para a entrega, instalação e uso operacional do alvo de avaliação, garantindo que a segurança não será comprometida durante o transporte, instalação, inicialização ou operação.

Esta classe é dividida em duas famílias: entrega, que cobre os procedimentos usados para manter a segurança durante o processo de entrega para o usuário; e instalação, geração e inicialização, as quais definem os requisitos para que uma cópia do alvo de avaliação, ao ser configurada e ativada pelo seu administrador, mantenha as mesmas propriedades do alvo de avaliação propriamente dito, dando subsídio para que o administrador saiba como proceder as configurações.

3.2.2.3 Garantias de Desenvolvimento (ADV)

A classe de garantias de desenvolvimento define os requisitos para o refinamento dos requisitos de segurança, a partir da especificação inicial do alvo de avaliação até a sua implementação. Cada uma das famílias ajuda no provimento de informações para o avaliador determinar quando os requisitos funcionais foram devidamente cumpridos durante o desenvolvimento.

Uma das famílias é a de especificação funcional, a qual deve instanciar de forma apurada e completa os requisitos funcionais, especificando também detalhes da interface onde os usuários devem operar com o alvo de avaliação. As outras famílias são de projetos de alto e baixo nível, as quais são repensáveis pelos detalhamentos das macroestruturas no caso do projeto de alto nível e dos detalhes necessários à implementação de software ou hardware no caso de baixo nível.

Ainda nesta classe, temos mais duas famílias importantes que são a de representação de correspondência, responsável por mostrar a correspondência entre todos os pares adjacentes de projetos desenvolvidos neste nível, e a família de modelagem de políticas, a qual define os requisitos de modelagem das políticas de segurança do alvo de avaliação, e aumenta a correspondência entre as políticas e os requisitos funcionais.

3.2.2.4 Garantias de Documentação (AGD)

A classe de garantias de documentação define os requisitos necessários ao entendimento, cobertura e completude da documentação operacional provida pelo desenvolvedor. As documentações estão divididas em duas categorias, uma para os usuários e a outra para os administradores.

Os requisitos para a documentação dos administradores são focados para o entendimento de todo o ambiente operacional do alvo de avaliação, dando completa informação ao administrador de como administrar o módulo de uma maneira segura e de coma fazer uso efetivo das funções de proteção e privilégios. Já os requisitos para a documentação do usuário ajudam a garantir que os usuários serão capazes de operar o alvo de avaliação de maneira segura, informando e explicando a ele quais são as funções visíveis ao seu nível de acesso, e como é seu uso, de forma que as proteções às informações sejam adequadamente garantidas.

3.2.2.5 Garantias de Suporte do Ciclo de Vida (ALC)

As classes de garantias de suporte ao ciclo de vida do alvo de avaliação definem os requisitos para garantir, através de modelos bem definidos de gerenciamento de ciclo de vida, todos os passos do desenvolvimento, incluídos remediação de problemas e políticas, o correto uso de ferramentas e técnicas e as medidas de segurança necessárias para garantir o processo de desenvolvimento do alvo de avaliação.

Dentro desta classe temos famílias para o controle de desenvolvimento seguro, remediação de problemas, definição do ciclo de vida e a definição de ferramentas e técnicas que garantirão o ciclo de vida do alvo de avaliação.

3.2.2.6 Garantias de Testes (ATE)

Os componentes desta classe respondem aos requisitos dos testes para demonstrar que as funções de segurança correspondem aos requisitos funcionais do alvo de avaliação.

Esta classe está dividida em 4 famílias que são:

testes de cobertura cuidam para que todos os testes funcionais tenham sido efetuados pelo desenvolvedor.

teste de profundidade garantem os detalhes com os quais os testes foram especificados pelo desenvolvedor.

testes funcionais garantem que as funções de segurança exibem as propriedades necessárias para satisfazer os requisitos do alvo de avaliação.

testes independentes especificam o grau com os quais serão testadas as funcionalidades por terceiros.

3.2.2.7 Garantias de Avaliação de Vulnerabilidades (AVA)

Nos componentes da classe de garantias de avaliação de vulnerabilidades, temos a definição dos requisitos direcionados à identificação de vulnerabilidades exploráveis, em especial as introduzidas no processo de construção, na operação, mau uso ou configuração incorreta do alvo de avaliação.

Esta classe está dividida em 4 famílias, que são: **família para análises e descobertas de canais de comunicação não documentados** que pode ser explorada para violar o alvo de avaliação, assim como temos a **família para analisar as possibilidades de mau uso** tornando fácil saber quando não foram seguidas as recomendações dos manuais do administrador e do usuário. Por fim, temos a **família de análise da força das funções de segurança**, as quais são testadas de maneira probabilística ou através de mecanismos de permutação, e a **família de análises de vulnerabilidades**, a qual tem por objetivo a identificação de falhas inseridas em diferentes passos durante o refinamento do desenvolvimento.

3.2.3 Níveis de Avaliação de Garantia

Os Critérios Comuns possuem um conjunto de 7 Níveis de Avaliação de Garantia, os quais foram construídos usando os componentes das famílias de garantia apresentados na seção 3.2.2

A existência dos níveis de garantia é devida à necessidade de compatibilidade com critérios anteriores, americanos e europeus, respectivamente o TCSEC [33] e o ITSEC [34].

Os níveis de garantia foram criados para servir de critério para a avaliação de perfis de proteção e dos alvos de segurança. Estes níveis de garantia provêem uma

crescente e uniforme escala, a qual leva em conta o nível obtido com os custos operacionais para tal.

O aumento dos níveis de garantia é obtidos através do uso ou substituição de componentes por outros componentes das mesmas famílias como no caso do aumento do rigor, escopo e/ou profundidade ou componentes de outras famílias com níveis mais altos de garantia como no caso da adição de novos requisitos.

Os níveis de avaliação de garantia começam com o EAL1 e vão ordenadamente até o EAL7. Pode-se facilmente alcançar o nível EAL4 de forma direta com qualquer produto de mercado, bastando para tal prover os componentes necessários de garantia para o mesmo, o que não envolve o reprojeto.

Acima do EAL4, os Alvos de Avaliação (TOE) devem ter sido especificamente projetados para alcançar determinado nível, visto que os componentes de garantia requerem a semi-formalização como requisito mínimo e em alguns casos a formalização total. No topo dos níveis de avaliação temos o EAL7, o qual nem sempre é fácil de ser atingido, pois sua aplicação prática normamente envolve grandes impactos em prazos e custos de desenvolvimento dos produtos. Em alguns casos, determinados produtos podem ser demasiadamente complexos para poderem ser projetados ou avaliados segundo este nível.

Os níveis de garantia possuem um formato não muito fixo, visto que possuem a idéia de aumento, a qual permite que qualquer EAL seja ampliado em suas características, sempre utilizando componentes hierarquicamente superiores. O mesmo não é válido para o decremento, visto que a norma não considera o decréscimo de nenhuma característica considerada necessária para um determinado nível.

A Tabela 3.5 dá uma visão geral das garantias obtidas em cada nível de avaliação de garantia.

Tabela 3.5: Requisitos dos níveis de avaliação de garantia.

Nível:	**Garantia Obtida:**
EAL1	Funcionalmente testado
EAL2	Estruturalmente testado
EAL3	Metodicamente testado e verificado
EAL4	Metodicamente projetado, testado e verificado
EAL5	Semi-formalmente projetado e testado
EAL6	Semi-formalmente verificado, projetado e testado
EAL7	Formalmente verificado, projetado e testado

3.2.3.1 EAL 1

O primeiro nível de avaliação de garantia provê uma análise básica das funções de segurança do alvo da avaliação, usando basicamente documentação para guiar o usuário para configurações mais seguras e especificações funcionais e de interface conforme a Tabela 3.6, provendo testes funcionais sobre o mesmo.

Tabela 3.6: Componentes de Garantia para EAL1

Componente:	**Descrição:**
ACM_CAP.1	Números de Versão
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.1	Especificação Funcional Informal
ADV_RCR.1	Demonstração Informal de Correspondência
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ATE_IND.1	Testes Independentes de Conformidade

Este nível é aplicável quando se necessita de garantias mínimas de corretude individual das funções de segurança do alvo de avaliação. Não são testadas em momento algum as interações entre as funções de segurança, fazendo-se a avaliação unicamente de forma individualizada. Uma vantagem deste nível de avaliação de garantia é a prova de que as funções implementadas pelo alvo de avaliação estão implementadas de acordo com a sua documentação e protegidas contra ameaças conhecidas. Este nível pode ser atingido sem cooperação do desenvolvedor.

Mesmo não contemplando significativas melhoras na segurança do alvo de avaliação, este nível é capaz de prover significante melhora se comparado com um produto não avaliado.

3.2.3.2 EAL 2

O segundo nível de avaliação de garantias provê testes estruturais no alvo de avaliação e requer que exista cooperação do desenvolvedor para a entrega de informações de projeto assim como de resultados de testes, mas não exige esforço adicional, visto que as informações necessárias por parte do desenvolvedor são obtidas somente usando boas práticas de negócio.

Este nível de avaliação é aplicável quando os desenvolvedores ou usuários requerem um moderado grau de garantia de segurança e não têm disponível todo o

projeto para a avaliação, o que é muito comum quando estão sendo implantados mecanismos de segurança em sistema legados, desenvolvidos por terceiras partes que não cedem seus projetos internos.

Como podemos ver na Tabela 3.7, o EAL2 tem basicamente os mesmos requisitos que o EAL1, com a adição principalmente do projeto descritivo de alto nível, o qual serve para dar um melhor entendimento do comportamento do alvo de avaliação.

Tabela 3.7: Componentes de Garantia para EAL2

Componente:	**Descrição:**
ACM_CAP.2	Ítens de Configuração
ADO_DEL.1	Procedimentos de Entrega
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.1	Especificação Funcional Informal
ADV_HLD.1	Projeto Descritivo de Alto Nível
ADV_RCR.1	Demonstração Informal de Correspondência
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ATE_COV.1	Evidências de Cobertura
ATE_FUN.1	Testes Funcionais
ATE_IND.2	Testes Independentes - Amostragem
AVA_SOF.1	Avaliação da Coesão das Funções de Segurança do Alvo de Avaliação
AVA_VLA.1	Análise de Vulnerabilidades do Desenvolvedor

A avaliação é suportada através de testes feitos por partes independentes do desenvolvedor, evidências de testes entregues pelo desenvolvedor, confirmação seletiva dos testes efetuados pelo desenvolvedor, análise da coesão das funções e evidência de resistência a vulnerabilidades de domínio público.

O EAL2 provê uma significativa melhoria em comparação com o EAL1, ao requerer testes por parte do desenvolvedor, testes de vulnerabilidades e testes independentes, baseados em uma especificação mais detalhada do alvo de avaliação.

3.2.3.3 EAL 3

O terceiro nível de avaliação de garantias provê uma estrutura mais consistente ao alvo de avaliação, exigindo do desenvolvedor boas técnicas de engenharia e sem grandes custos impactantes no projeto e sem alterações substanciais dos métodos de desenvolvimento, envolvendo garantias de testes e checagens metódicas.

Este nível é necessário quando os desenvolvedores ou usuários requerem um nível de segurança moderado e garantido de forma independente entre as funções

do alvo de avaliação, sem a possibilidade de impactos no projeto.

O EAL3 provê garantias através de uma análise mais aprofundada das funções de segurança, usando uma especificação funcional, documentação e um projeto focado em questões de segurança, conforme a tabela 3.8

Tabela 3.8: Componentes de Garantia para EAL3

Componente:	**Descrição:**
ACM_CAP.3	Controles de Autorização
ACM_SCP.1	Cobertura do Gerenciamento de Configuração do Alvo de Avaliação
ADO_DEL.1	Procedimentos de Entrega
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.1	Especificação Funcional Informal
ADV_HLD.2	Projeto de Alto Nível para Reforço de Segurança
ADV_RCR.1	Demonstração Informal de Correspondência
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ALC_DVS.1	Identificação das Medidas de Segurança
ATE_COV.2	Análises de Cobertura
ATE_FUN.1	Testes Funcionais
ATE_DPT.1	Testes do Projeto de Alto Nível
ATE_IND.2	Testes Independentes - Amostragem
AVA_MSU.1	Exame das Orientações aos Usuários
AVA_SOF.1	Avaliação de Coesão das Funções de Segurança do Alvo de Avaliação
AVA_VLA.1	Análise de Vulnerabilidades do Desenvolvedor

A análise no EAL3 é essencialmente dirigida como a do EAL2, sendo unicamente mais detalhada e ajustada às questões relativas ao projeto de alto nível do alvo de avaliação.

O EAL3 provê garantias também através do uso de controles no ambiente de desenvolvimento, gerenciamento de configuração e evidência de procedimentos de segurança. Este nível provê um significativo aumento nas garantias providas pelo EAL2, requerendo melhor cobertura dos testes das funções de segurança e garantias de que o alvo de avaliação não vai ser alterado durante o desenvolvimento.

3.2.3.4 EAL 4

Neste nível o desenvolvedor pode explorar o máximo das garantias provindas de boas práticas de desenvolvimento, as quais ainda não requerem um profissional especializado em segurança e é o último nível sem que o produto tenha sido projetado levando em consideração este critério comum. Neste nível são garantidos metódicamente

os projetos, os testes e as revisões.

O EAL4 é aplicável em circunstâncias onde os desenvolvedores já estão preparados para ter gastos com a adequação dos projetos e os usuários requerem um nível de segurança de moderado a alto em seus alvos de avaliação.

Tabela 3.9: Componentes de Garantia para EAL4

Componente:	**Descrição:**
ACM_AUT.1	Automatização Parcial do Gerenciamento de Configuração
ACM_CAP.4	Suporte de Geração e Procedimentos de Aceitação
ACM_SCP.2	Cobertura do Gerenciamento de Configuração para Seguir Problemas
ADO_DEL.2	Detecção de Modificação
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.2	Interfaces Externas Completamente Definidas
ADV_HLD.2	Projeto de Alto Nível para Reforço de Segurança
ADV_IMP.1	Subconjunto da Implementação das Funções de Segurança do Alvo de Avaliação
ADV_LLD.1	Projeto Descritivo de Baixo Nível
ADV_RCR.1	Demonstração Informal de Correspondência
ADV_SPM.1	Modelo Informal da Política de Segurança do Alvo de Avaliação
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ALC_DVS.1	Identificação das Medidas de Segurança
ALC_LCD.1	Modelo do Ciclo de Vida Definido pelo Desenvolvedor
ALC_TAT.1	Ferramentas de Desenvolvimento bem definidas
ATE_COV.2	Análises de Cobertura
ATE_FUN.1	Testes Funcionais
ATE_DPT.1	Testes do Projeto de Alto Nível
ATE_IND.2	Testes Independentes - Amostragem
AVA_MSU.2	Validação das Análises
AVA_SOF.1	Avaliação de Coesão das Funções de Segurança do Alvo de Avaliação
AVA_VLA.2	Análise de Vulnerabilidades Independentes

Conforme a Tabela 3.9, o EAL4 provê garantias pela análise das funções de segurança usando uma especificação funcional completa, uma boa documentação, projetos de alto e baixo nível e um sub-conjunto da implementação para avaliar o comportamento do alvo de avaliação nos quesitos de segurança. Uma nova garantia é obtida pela modelagem informal da política de segurança do alvo de avaliação.

A análise é ratificada por testes independentes, por validação dos testes do desenvolvedor com base na especificação e no projeto de alto nível, por evidências de análise de vulnerabilidades por parte do desenvolvedor e por uma análise independente que demonstra resistência a ataques de penetração com baixos potenciais. Os ataques são definidos a partir das ameaças descritas no documento do alvo de segurança.

O EAL4 também provê garantias através do ambiente de desenvolvi-

mento controlado e pela adição de um gerenciamento de configuração incluindo automatização de evidências de procedimentos seguros.

Este nível demonstra significativos ganhos de garantias se comparado com o EAL3, requerendo mais descrições de projetos, um sub-conjunto da implementação e melhores mecanismos e procedimentos para garantir que o alvo de avaliação não será violado durante o desenvolvimento ou entrega.

3.2.3.5 EAL 5

O nível de avaliação de garantias EAL5 exige que desenvolvedor explorar ao máximo as técnicas de desenvolvimento comerciais, suportando em conjunto modestas técnicas especializadas em engenharia de segurança. Para a obtenção deste nível de garantia, um alvo de avaliação deverá ser projetado e desenvolvido com o foco na obtenção no nível de avaliação, visto que existem necessidades de projeto e testes semi-formalmente projetados.

O EAL5 é aplicável em situações onde os desenvolvedores ou usuários requerem um alto nível de segurança independente garantido por um desenvolvimento planejado e com mecanismos rigorosos de desenvolvimento.

O EAL5, conforme a Tabela 3.10, provê garantias pela análise de funções usando uma especificação funcional e de interfaces completa, com projetos de alto e baixo nível e com toda a implementação para que a compreensão do alvo de avaliação seja total. Garantias adicionais são providas através de um modelo formal da política de segurança, uma semi-formalização da especificação funcional e do projeto de alto nível, em conjunto com uma demonstração de correspondência entre estes dois, sendo todo o alvo de avaliação obrigatoriamente construído de forma modular.

A análise do EAL5 é suportada por testes independentes nas funções de segurança, evidências de testes realizados por parte do desenvolvedor, projeto de alto e baixo níveis, análise dos testes do desenvolvedor, evidências de testes contra vulnerabilidades com potencial moderado de ataque e também a validação das análises do desenvolvedor. O EAL5 também provê garantias através do uso de controles no ambiente de desenvolvimentos e a automatização do gerenciamento de configuração.

Este nível traz grandes benefícios se comparado com o EAL4, visto que requer descrições semi-formais dos projetos, em conjunto com a completa implementa-

Tabela 3.10: Componentes de Garantia para EAL5

Componente:	Descrição:
ACM_AUT.1	Automatização Parcial do Gerenciamento de Configuração
ACM_CAP.4	Suporte de Geração e Procedimentos de Aceitação
ACM_SCP.3	Cobertura do Gerenciamento de Configuração para Ferramentas de Desenvolvimento
ADO_DEL.2	Detecção de Modificação
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.3	Especificação Funcional Semi-Formal
ADV_HLD.3	Projeto de Alto Nível Semi-Formal
ADV_IMP.2	Implementação das Funções de Segurança do Alvo de Avaliação
ADV_INT.1	Modularidade
ADV_LLD.1	Projeto Descritivo de Baixo Nível
ADV_RCR.2	Demonstração Semi-Formal de Correspondência
ADV_SPM.3	Modelo Formal da Política de Segurança do Alvo de Avaliação
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ALC_DVS.1	Identificação das Medidas de Segurança
ALC_LCD.2	Modelo do Ciclo de Vida Padronizado
ALC_TAT.2	Implementação Compatível com Padrões
ATE_COV.2	Análises de Cobertura
ATE_FUN.1	Testes Funcionais
ATE_DPT.2	Testes do Projeto de Baixo Nível
ATE_IND.2	Testes Independentes - Amostragem
AVA_CCA.1	Análises de Segurança de Canais
AVA_MSU.2	Validação das Análises
AVA_SOF.1	Avaliação de Coesão das Funções de Segurança do Alvo de Avaliação
AVA_VLA.3	Resistência Moderada

ção, e uma arquitetura mais estruturada, com mecanismos que garantam que o alvo de avaliação não seja violado durante a implementação.

3.2.3.6 EAL 6

Este EAL exige dos desenvolvedores melhores procedimentos, através da aplicação de técnicas de engenharia seguras, dentro de um rigoroso ambiente de desenvolvimento, para obter um produto de qualidade que proteja seus recursos contra riscos significativos, usando mecanismos de verificação de projeto e de testes especificados de maneira semi-formal.

O EAL6 é aplicável para o desenvolvimento de produtos que visem aplicações de alto risco, onde os valores adicionais dos procedimentos incluídos neste nível sejam justificados pelos recursos que eles protegem, visto que, para alcançar este nível de avaliação, os alvos de avaliação devem ser projetados desde o início visando este

nível.

Tabela 3.11: Componentes de Garantia para EAL6

Componente:	**Descrição:**
ACM_AUT.2	Automatização Total do Gerenciamento de Configuração
ACM_CAP.5	Suporte Avançado
ACM_SCP.3	Cobertura do Gerenciamento de Configuração para Ferramentas de Desenvolvimento
ADO_DEL.2	Detecção de Modificação
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.3	Especificação Funcional Semi-Formal
ADV_HLD.4	Explanação de Alto Nível Semi-Formal
ADV_IMP.3	Implementação Estruturada das Funções de Segurança do Alvo de Avaliação
ADV_INT.2	Redução da Complexidade
ADV_LLD.2	Projeto Semi-Formal de Baixo Nível
ADV_RCR.2	Demonstração Semi-Formal de Correspondência
ADV_SPM.3	Modelo Formal da Política de Segurança do Alvo de Avaliação
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ALC_DVS.2	Suficiência das Medidas de Segurança
ALC_LCD.2	Modelo do Ciclo de Vida Padronizado
ALC_TAT.3	Implementação Compatível com Padrões - Todas as Partes
ATE_COV.3	Análises de Cobertura Rigorosa
ATE_FUN.2	Testes Funcionais Ordenados
ATE_DPT.2	Testes do Projeto de Baixo Nível
ATE_IND.2	Testes Independentes - Amostragem
AVA_CCA.2	Análises de Segurança de Canais Sistemática
AVA_MSU.3	Análise e Testes por Estados Inseguros
AVA_SOF.1	Avaliação de Coesão das Funções de Segurança do Alvo de Avaliação
AVA_VLA.4	Altamente Resistente

Conforme a Tabela 3.11, o EAL6 provê garantias através da análise das funções de segurança usando uma especificação funcional completa, os projetos de alto e baixo nível, e uma apresentação estruturada da implementação para o entendimento do comportamento de segurança do alvo de avaliação. Mais garantias são adicionadas através de uma representação formal da política de segurança, de projetos de alto e baixo nível com um representação semi-formal de correspondência entre eles, com um projeto modularizado e em camadas.

A análise é suportada por testes independentes nas funções de segurança do módulo, evidências de testes por parte do desenvolvedor com base na especificação funcional, projeto de alto e baixo nível, confirmação seletiva dos testes do desenvolvedor, evidências de testes de vulnerabilidades por parte do desenvolvedor, e análise de vulnerabilidades independentes demonstrando resistência a ataques de penetração de alto

potencial, incluindo também a validação da análise sistemática do desenvolvedor.

O EAL6 também provê garantias através do processo de desenvolvimento estruturado, de controle do ambiente de desenvolvimento, e da completa automatização da gerência de configuração, evidenciando procedimentos seguros de entrega.

Este nível representa significativos avanços em comparação ao EAL5, requerendo uma análise mais compreensiva, uma representação estruturada da implementação, mais análises independentes de vulnerabilidades e melhores gerenciamentos de configurações e controles do ambiente de desenvolvimento.

3.2.3.7 EAL 7

Neste nível de avaliação de garantia, os alvos de avaliação são considerados de extrema importância e devem prover a segurança para ambientes de alto risco e com muito valor agregado, o suficiente para justificar os custos de projeto e desenvolvimento. As implementações práticas deste EAL são normalmente limitadas a alvos de avaliação que são muito dirigidos a uma funcionalidade de segurança e que são passíveis de uma extensiva análise formal.

O EAL7, conforme a tabela 3.12, provê garantias através da análise de funções de segurança, através de uma especificação funcional e de interface completas, dos projetos de alto e baixo nível, e da apresentação estruturada da implementação para um melhor entendimento do comportamento de segurança do alvo de avaliação. Melhoras são asseguradas através da formalização da política de segurança, da formalização da especificação funcional e do projeto de alto nível, e com uma apresentação semi-formal do projeto de baixo-nível, finalizando com demonstrações formais e semi-formais da interação entre eles, usando um projeto simples, modular e em camadas.

A análise é suportada por testes independentes das funções de segurança, evidências de testes por parte do desenvolvedor baseados na especificação funcional, projeto de alto nível, projeto de baixo nível, representação da implementação e através da confirmação completa dos resultados obtidos pelo desenvolvedor, evidências de buscas de vulnerabilidades por parte do desenvolvedor e uma análise independente de vulnerabilidades, demonstrando resistência a potenciais ataques com alto risco.

O EAL7 também provê garantias através do uso de um processo de desenvolvimento estruturado, de controles no ambiente de desenvolvimento, completa auto-

Tabela 3.12: Componentes de Garantia para EAL7

Componente:	Descrição:
ACM_AUT.2	Automatização Total do Gerenciamento de Configuração
ACM_CAP.5	Suporte Avançado
ACM_SCP.3	Cobertura do Gerenciamento de Configuração para Ferramentas de Desenvolvimento
ADO_DEL.3	Prevenção de Modificação
ADO_IGS.1	Procedimentos para Instalação, Geração e Inicialização
ADV_FSP.4	Especificação Funcional Formal
ADV_HLD.5	Explanação de Alto Nível Formal
ADV_IMP.3	Implementação Estruturada das Funções de Segurança do Alvo de Avaliação
ADV_INT.3	Minimização da Complexidade
ADV_LLD.2	Projeto Semi-Formal de Baixo Nível
ADV_RCR.3	Demonstração Formal de Correspondência
ADV_SPM.3	Modelo Formal da Política de Segurança do Alvo de Avaliação
AGD_ADM.1	Guia do Administrador
AGD_USR.1	Guia do Usuário
ALC_DVS.2	Suficiência das Medidas de Segurança
ALC_LCD.3	Modelo do Ciclo de Vida Mensurável
ALC_TAT.3	Implementação Compatível com Padrões - Todas as Partes
ATE_COV.3	Análises de Cobertura Rigorosa
ATE_FUN.2	Testes Funcionais Ordenados
ATE_DPT.3	Testes da Representação da Implementação
ATE_IND.3	Testes Independentes - Completos
AVA_CCA.2	Análises de Segurança de Canais Sistemática
AVA_MSU.3	Análise e Testes por Estados Inseguros
AVA_SOF.1	Avaliação da Coesão da Funções de Segurança do Alvo de Avaliação
AVA_VLA.4	Altamente Resistente

matização na gerência de configuração e evidência de procedimentos de entrega seguros.

Este nível representa o topo da estrutura especificada pelos Critérios Comuns e requer a análise compreensiva, a representação formal, a formalização das correspondências e testes compreensivos no alvo de avaliação.

3.2.4 Perfis de Proteção

Os perfis de proteção - PP descrevem conjuntos independentes de implementação de requisitos de segurança para alvos de avaliação categorizados em grupos de funções ou objetivos afins, e contém uma definição do problema de segurança que um produto em conformidade com o perfil de proteção deve resolver.

Eles especificam requisitos de garantia e de funcionalidades a partir dos componentes dos critérios comuns, e provém uma lógica para a seleção de componentes de segurança funcional e de garantias.

Um perfil de proteção é estruturado de forma a prover os objetivos de segurança que os alvos de avaliação devem cumprir, provendo contextos para a avaliação, os requisitos funcionais e de garantia para o cumprimento dos objetivos de segurança, a segurança do ambiente onde o alvo de avaliação está inserido, e por fim as conclusões lógicas de que os objetivos foram totalmente alcançados e de que os requisitos estão de acordo com os objetivos de segurança.

Os equipamentos de proteção criptográficas, também conhecidos como HSMs, hoje existentes não aderem diretamente ao CC, pelo fato de não possuírem um PP específico para a sua classe de funcionalidades. Para a implementação do PSC proposto neste trabalho, vamos assumir o uso do PP para dispositivos seguros de criação de assinaturas [27–29], visto que este PP trata da segurança no gerenciamento de chaves e de mecanismos de proteção para os processos de assinatura digital, foco do uso de um PSC nas ICPs. Este PP é classificado para a obtenção de um nível de avaliação EAL4, com os requisitos adicionais de avaliação mais rigorosa da coesão das funções de segurança.

3.2.5 Alvos de Segurança

Os Alvos de Segurança - ST são os acordos entre desenvolvedores, consumidores, avaliadores e autoridades de valiação que determina os requisitos de segurança que um alvo de avaliação deve oferecer e seu escopo. Eles são documentos genéricos e podem incluir gerenciamento, publicidade, compra, instalação, operação e uso.

Um alvo de avaliação é estruturado de forma a conter uma introdução, onde é apresentada uma visão geral e são descritos os potenciais usos, uma declaração dos objetivos de segurança, uma descrição dos alvos de avaliação, onde é provido o contexto para a avaliação, os requisitos de segurança, um sumário de especificação, a descrição do ambiente operacional, e por fim são relatadas os perfis de proteção necessários ao alvo de segurança.

Os alvos de segurança são documentos mais genéricos e não tão detalhados que são usados para determinar os perfis de avaliação de um produto segundo os níveis de avaliação de garantias dos critérios comuns.

O ST recomendado para o PSC não é alvo direto deste trabalho, visto que as generalidades do documento excedem o escopo da dissertação.

3.2.6 Conclusões

Este capítulo mostrou brevemente as normas e preocupações internacionais no desenvolvimento de mecanismos de proteção de processos e chaves criptográficas. Também foram vistos os métodos e técnicas usados para garantir a segurança de produtos de tecnologia da informação em geral.

Estas normas servem como uma base tanto para avaliação de produtos quanto para a construção de um PSC que possa efetivamente ser aceito pela comunidade internacional e usuários. Este levantamento de normas também deixou claro que no Brasil não existe preocupação e normatizar quesitos relativos a segurança, pois não há participação efetiva da comunidade brasileira na determinação de tais normas.

Por fim as normas aqui apresentadas serão o alicerce para o projeto e implementação do PSC proposto e o tornarão um sistema produto desta dissertacão coeso e aceito pela comunidade de segurança da informação.

Capítulo 4

Módulos Criptográficos Comerciais

Encontram-se no mercado vários módulos criptográficos aprovados pelo padrão FIPS PUB 140. É muito interessante e significativo sumarizar e caracterizar os módulos hoje existentes, para aproveitá-los como base na adequação de idéias e projetos de provedores criptográficos embarcados. A lista dos módulos aprovados pode ser facilmente obtida através no sítio http://www.nist.gov/cmvp.

Percebe-se que os módulos criptográficos aprovados pela FIPS PUB 140, em sua maioria, não levam em consideração como parte dos requisitos de projeto do módulo, os requisitos inerentes das aplicações de ICP conforme descrito no capítulo 2. Estes equipamentos aprovados são em sua maioria aceleradores criptográficos que protegem de forma genéricas suas chaves, sendo o controle de acesso e o controle do ciclo de vida das chaves privadas um simples acessório.

Seguindo as diretivas de disponibilidade e uso dos equipamentos, partimos para a avaliação dos módulos criptográficos hoje suportados nativamente pela biblioteca OpenSSL, que é uma ferramenta livre para a implementação de rotinas criptográficas. A escolha do OpenSSL como parâmetro de avaliação dos módulos comerciais se deve à disponibilidade do seu código e, em função disso, da facilidade futura de integração do PSC fruto desta tese com ele.

O OpenSSL foi criado com o intuito de implementar um conjunto de ferramentas para o protocolo SSL. Mas, para atingir tal objetivo, durante o desenvolvimento foram sendo criadas várias outras estruturas, dentre elas, foi implementada de uma forma eficiente e concisa uma biblioteca de funções criptográficas. Uma característica muito importante do OpenSSL é a capacidade que ele tem de se comunicar com módulos

de hardware seguros, como os citados anteriormente.

O projeto OpenSSL é também único em sua implementação, pois é uma das poucas bibliotecas livres para criptografia disponíveis na Internet e que não sofre regulamentação de exportação de nenhum país. Ele também é o único projeto completo de biblioteca para as linguagens C e C++, que são as bases hoje utilizadas na construção de sistemas operacionais abertos. As rotinas criptográficas do OpenSSL foram submetidas aos testes de conformidade com as padronização da norma FIPS PUB 140-2. Os testes foram realizados nos seguintes algoritmos:

- Advanced Encryption Standard (AES): Certificado número 146 [3, 35].
- Data Encryption Standard (DES): Certificado número 258 [2, 36].
- Triple Data Encryption Algorithm (TDEA,"Triple DES"): Certificado número 256 [37].
- Digital Signature Algorithm (DSA): Certificado número 108 [38].
- Secure Hash Algorithm (SHS): Certificado número 235 [7, 39].

O OpenSSL como pacote completo não possui a certificação FIPS PUS 140. Existe um esforço da comunidade de software livre para que ele atinja esta certificação, mas devido as constantes atualizações e desenvolvimentos, os custos de re-certificação inviabilizam tal obtenção.

Os módulos de hardware criptográficos (HSMs) hoje suportados por padrão pela biblioteca OpenSSL, e devidamente suportados pelos seus desenvolvedores, são os apresentados na Tabela 4.1 [22]:

Tabela 4.1: Relação de fabricantes de módulos criptográficos suportados pelo OpenSSL

Fabricante:	**Maiores Informações:**
CRYPTOSWIFT	http://www.safenet-inc.com
NCIPHER	http://www.ncipher.com
ATALLA/HP	http://www.atalla.com
AEP/SUREWARE	http://www.aepsystems.com
IBM 4758	http://www-3.ibm.com/security/cryptocards/

Como quesitos de avaliação dos módulos deste fabricantes, serão somente considerados os aprovados pelas normas FIPS PUB 140-1 e FIPS PUB 140-2, nos

níveis 3 e 4. A avaliação de módulos aprovados para níveis inferiores não será realizada pois os mesmos não são adequados para a construção de uma ICP.

Serão sempre considerados os dados constantes no certificado de homologação do NIST. O certificado de homologação do NIST é o documento que atesta a conformidade de uma versão específica do equipamento à norma vigente. Alguns módulos possuem mais de um certificado, pois tratam-se de versões diferentes do mesmo produto.

O certificado leva em conta:

- Versão da FIPS PUB 140,
- Projeto do módulo criptográfico,
- Construção do mecanismo de autenticação baseado em papéis e serviços,
- Segurança física,
- Emissão eletromagnética,
- Gerenciamento de chaves,
- Interfaces de comunicação,
- Máquina finita de estados,
- Segurança do software,
- Auto testes,
- Algoritmos criptográficos implementados, tantos os concordantes com padrões FIPS como os não concordantes.

Estes critérios referentes a cada módulo avaliado estão expressos nas Tabelas 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.7, 4.8 e 4.9.

Um quesito de avaliação importante serão os custos, tantos de aquisição do equipamento, quanto de manutenção e de continuidade de negócio em caso de desastre.

4.1 AEP - ACCE SureWare Keyper Professional

Figura 4.1: AEP - ACCE SureWare Keyper Professional

Este módulo é o utilizado pela ICP-Brasil [11] e conta com características bastantes interessantes, dentre elas a capacidade de realizar 160 assinaturas RSA de 1024 bits por segundo, e a facilidade de ser totalmente independente de qualquer outro hardware para operar. Ele também é capaz de fazer cópias de segurança da chave privada, autenticar o operador através de tokens, e conta com todos os mecanismos de autenticação integrados, tais como leitor de smart cards, teclado para entrada de senhas, e uma interface RS232 para retirada de dados para auditoria do sistema [40]. Este equipamento é mostrado na Figura 4.1, e as suas características podem ser vistas de acordo com a Tabela 4.2.

Tabela 4.2: Características do AEP Professional

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	112 146 235
Norma FIPS PUB	140-1
Nível Final	Nível 4
Projeto do Módulo	Nível 4
Mecanismo de Autenticação	Nível 4
Segurança Física	Nível 4
Emissão Eletromagnética	Nível 4
Gerenciamento de Chaves	Nível 4
Interfaces de Comunicação	Nível 4
Máquina Finita de Estados	Nível 4
Segurança do Software	Nível 4
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	DES MAC Triple DES Triple DES MAC SHA-1 DSA RSA PKCS#1
Algoritmos Criptográficos Não FIPS	MD5 Diffie-Hellman RSA cifragem e decifragem RSA X.509
Preço Aproximado (FOB)	U$17.500

4.2 AEP - ACCE SureWare Keyper PCI

Este conta com características mais básicas se comparado com seu modelo superior, o ACCE SureWare Keyper Professional. A conectividade é via uma interface PCI, podendo ser ligado em praticamente qualquer computador que disponha de uma interface PCI, com a possibilidade de conexão de um mecanismo externo para comunicação com o módulo, seja para autenticação, seja para manutenção. Ele é também capaz de fazer cópias de segurança da chave privada, autenticar o operador por tokens, e conta com todos os mecanismos de autenticação plugáveis via uma interface externa, tais como leitora de smart cards, e um teclado para entrada de senhas [40]. O equipamento é o que aparece na Figura 4.2, e as características deste equipamento podem ser vistas de acordo com a Tabela 4.3.

Figura 4.2: AEP - ACCE SureWare Keyper PCI

Tabela 4.3: Características do AEP PCI

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	155
Norma FIPS PUB	140-1
Nível Final	Nível 3
Projeto do Módulo	Nível 3
Mecanismo de Autenticação	Nível 3
Segurança Física	Nível 3 + EFT
Emissão Eletromagnética	Nível 3
Gerenciamento de Chaves	Nível 3
Interfaces de Comunicação	Nível 3
Máquina Finita de Estados	Nível 3
Segurança do Software	Nível 4
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	DES MAC Triple DES Triple DES MAC SHA-1 DSA
Algoritmos Criptográficos Não FIPS	MD5 Diffie-Hellman RSA cifragem e decifragem RSA X.509
Preço Aproximado (FOB)	U$8.500

4.3 Atalla/HP - ACE NSP

Os equipamentos da série NSP são todos certificados sob o mesmo certificado FIPS PUB 140. A diferença entre os equipamentos da linha é quanto à capacidade de transações RSA por segundo, tendo assim uma boa escalabilidade. Como características importantes, os equipamentos da Atalla/HP possuem interface de conexão Ethernet, com TCP/IP, e um console administrativo, sem a necessidade de nenhum equipamento adicional para o gerenciamento do módulo. Ele possui também proteção contra atenuações no ambiente, contando com sensores de temperatura, tensão e corrente da fonte de alimentação, e toda a manutenção e atualização do equipamento pode ser feita através de um CD-ROM interno ao equipamento, o qual possui a devida proteção de acesso físico. [41, 42]. Um ponto importante deste módulo é que sua validação de conformidade com a norma FIPS PUB 140 é bastante recente e de acordo com a revisão FIPS PUB 140-2. Os equipamentos desta série aparecem na Figura 4.3, e as características destes equipamentos podem ser vistas de acordo com a Tabela 4.4.

Figura 4.3: Atalla/HP - ACE NSP

Tabela 4.4: Características do Atalla/HP

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	296
Norma FIPS PUB	140-2
Nível Final	Nível 3
Projeto do Módulo	Nível 3
Mecanismo de Autenticação	Nível 3
Segurança Física	Nível 3 + EFP/EFT
Emissão Eletromagnética	Nível 3
Gerenciamento de Chaves	Nível 3
Interfaces de Comunicação	Nível 3
Máquina Finita de Estados	Nível 3
Garantia de Desenvolvimento	Nível 3
Auto Testes	Nível 3
Algoritmos Criptográficos FIPS	Triple DES Triple DES MAC SHA-1
Algoritmos Criptográficos Não FIPS	MD5 RIPEMD RSA PKCS#1 versão 2
Preço Aproximado (FOB)	Não informado

4.4 Rainbow - CryptoSwift HSM

Este módulo conta com características importantes, dentre elas a conectividade via uma interface PCI e a possibilidade de conexão de um mecanismo externo para a cópia de segurança da chave usando um token USB. Ele também é capaz de fazer cópias de segurança da chave privada, autenticar o operador por tokens, e conta com todos os mecanismos de autenticação plugáveis via uma interface externa. Também conta com uma interface RS232 para execução de auditorias no módulo [43]. As características deste equipamento podem ser vistas de acordo com a Tabela 4.5.

Tabela 4.5: Características do Rainbow CryptoSwift HSM

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	162
Norma FIPS PUB	140-1
Nível Final	Nível 3
Projeto do Módulo	Nível 3
Mecanismo de Autenticação	Nível 3
Segurança Física	Nível 3
Emissão Eletromagnética	Nível 3
Gerenciamento de Chaves	Nível 3
Interfaces de Comunicação	Nível 3
Máquina Finita de Estados	Nível 3
Segurança do Software	Nível 3
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	Triple DES Triple DES MAC DES SHA-1 PKCS#1
Algoritmos Criptográficos Não FIPS	MD5 HMAC MD5 HMAC SHA1 RC4 RSA cifragem DSA
Preço Aproximado (FOB)	U$17.350

4.5 IBM 4758-002 PCI

Figura 4.4: IBM 4758-002 PCI

As principais características deste módulo são sua interface de conexão PCI, e uma interface serial RS232 para a auditoria de registros de eventos. Uma característica também importante é que, ao se desrespeitar as condições ambientais e ser detectada uma invasão, o módulo se torna inutilizável, sendo necessária a sua substituição por um novo equipamento. Este é um requisito desejável em aplicações militares [15, 44]. Os módulos da Família 4758 da IBM estavam entre os primeiros a serem validados pela norma FIPS PUB 140-1.

Este é o modulo que aparece na Figura 4.4 e as características deste equipamento estão resumidas na Tabela 4.6.

Tabela 4.6: Características do IBM 4758-002 PCI

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	116
Norma FIPS PUB	140-1
Nível Final	Nível 4
Projeto do Módulo	Nível 4
Mecanismo de Autenticação	Nível 4
Segurança Física	Nível 4
Emissão Eletromagnética	Nível 4
Gerenciamento de Chaves	Nível 4
Interfaces de Comunicação	Nível 4
Máquina Finita de Estados	Nível 4
Segurança do Software	Nível 4
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	DES DES MAC Triple DES DSA SHA-1
Algoritmos Criptográficos Não FIPS	RSA
Preço Aproximado (FOB)	U$12.000

4.6 IBM 4758-023 PCI

As principais características deste módulo são sua interface de conexão PCI, e uma interface serial RS-232 para a auditoria de registros de eventos. Uma característica também importante é que ao se desrespeitar as condições ambientais e de detecção de invasão o módulo se torna inutilizável. Este modelo é menos restritivo às condições ambientais, pois o mesmo está em conformidade com o nível 3 da Norma FIPS PUB 140-1 [15, 44]. As características deste equipamento podem ser vistas de acordo com a Tabela 4.7.

Tabela 4.7: Características do IBM 4758-023 PCI

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	117
Norma FIPS PUB	140-1
Nível Final	Nível 3
Projeto do Módulo	Nível 4
Mecanismo de Autenticação	Nível 4
Segurança Física	Nível 3 + EFP/EFT
Emissão Eletromagnética	Nível 4
Gerenciamento de Chaves	Nível 4
Interfaces de Comunicação	Nível 4
Máquina Finita de Estados	Nível 4
Segurança do Software	Nível 4
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	DES DES MAC Triple DES DSA SHA-1
Algoritmos Criptográficos Não FIPS	RSA
Preço Aproximado	U$7.500

4.7 Ncipher - nShield F3

Figura 4.5: Ncipher - nShield F3

Os módulos de segurança da série nShield, contam com várias interfaces de conexão, dentre elas SCSI e PCI, o que os torna virtualmente conectáveis a qualquer computador ou sistema. Outras características importantes são a escalabilidade, operando de 150 a 400 transações RSA por segundo, além de ser um módulo em constante avaliação pelo NIST, visto pelo número de certificados que ele detém. Um ponto deve ser ressaltado: os módulos desta série são os que contam com a maior gama de algoritmos criptográficos aprovados e não aprovados por normas FIPS PUB, o que o caracteriza como uma solução extremamente maleável e adaptável às necessidades de funções criptográficas da aplicação de ICP [17, 45]. O módulo é o que aparece na Figura 4.5, e as características podem ser vistas de acordo com a Tabela 4.8

Tabela 4.8: Características do Ncipher nShield F3

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	129, 150, 174, 180, 221, 222, 297 , 300
Norma FIPS PUB	140-2
Nível Final	Nível 3
Projeto do Módulo	Nível 3
Mecanismo de Autenticação	Nível 3
Segurança Física	Nível 3
Emissão Eletromagnética	Nível 3
Gerenciamento de Chaves	Nível 3
Interfaces de Comunicação	Nível 3
Máquina Finita de Estados	Nível 3
Garantia de Desenvolvimento	Nível 3
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	Triple DES , Triple DES MAC
	DES,DES MAC, AES,
	RSA PKCS#1, DSA/SHA1
Algoritmos Criptográficos Não FIPS	ARC FOUR
	CAST5, CAST6
	HMAC MD2, MD5, SHA-256, SHA-384, SHA-512, RIPEMD160
	MD2, MD5, RIPEMD160
	SHA-256, SHA-384, SHA-512
	El-Gamal, Diffie-Hellman
	Blowfish ,Twofish
	Serpent, KCDSA, HSA 160
Preço Aproximado	U$22.000

4.8 Ncipher - nForce 800/1600 PCI

Figura 4.6: nForce PCI

Os módulos de segurança da série nForce contam com a interface de conexão PCI. Outras características importantes destes módulos são a escalabilidade, operando de 800 a 1600 transações RSA de 1024 bits por segundo. Também é um módulo com um projeto novo, diferindo do nShield, que vem evoluindo no tempo. Ele difere basicamente da série nShield no quesito invólucro, mas mesmo assim conta com o mesmo nível de certificação FIPS PUB 140 [17, 45]. O módulo é o que aparece na Figura 4.6, e as características podem ser vistas de acordo com a Tabela 4.9.

Tabela 4.9: Características do Ncipher nForce 800/1600 PCI

Requisito:	**Valor Obtido/Referência:**
Números dos Certificado FIPS	294
Norma FIPS PUB	140-2
Nível Final	Nível 3
Projeto do Módulo	Nível 3
Mecanismo de Autenticação	Nível 3
Segurança Física	Nível 3
Emissão Eletromagnética	Nível 3
Gerenciamento de Chaves	Nível 3
Interfaces de Comunicação	Nível 3
Máquina Finita de Estados	Nível 3
Garantia de Desenvolvimento	Nível 3
Auto Testes	Nível 4
Algoritmos Criptográficos FIPS	Triple DES , Triple DES MAC DES , DES MAC, AES RSA PKCS#1, DSA/SHA1
Algoritmos Criptográficos Não FIPS	ARC FOUR CAST5, CAST6, HMAC MD2, MD5, SHA-256, SHA-384, SHA-512, RIPEMD160 MD2, MD5, RIPEMD160 SHA-256, SHA-384, SHA-512 El-Gamal, Diffie-Hellman Blowfish ,Twofish Serpent, KCDSA, HSA 160
Preço Aproximado	U$14.000

4.9 Comparativos entre Equipamentos

Os equipamentos estudados se assemelham bastante quanto as qualidades de proteção de chaves, visto que todos são certificados pelo NIST, usando como base a norma FIPS-PUB 140. Nem todos possuem preocupação com os problemas inerentes de uma ICP, mas de forma geral são capazes de proteger suas chaves de ataques conhecidos.

Um quesito importante na decisão de aquisição de um equipamento por parte de uma empresa, é normalmente a razão custo/benefício, e uma forma de demonstrar quantitativamente esta razão é evidenciarmos a razão RSA/segundo/Dólar investido na aquisição do equipamento. A Tabela 4.10 nos da esta perspectiva.

Tabela 4.10: Comparativo RSA/Segundo/Dólar Investido

Equipamento:	**Relação RSA/Segundo/Dólar:**
ACCE SureWare Keyper Professional	U$109,37/RSA/Segundo
ACCE SureWare Keyper PCI	U$53,12/RSA/Segundo
CryptoSwift HSM	U$86,75/RSA/Segundo
IBM 4758-002 PCI	U$68,57/RSA/Segundo
IBM 4758-023 PCI	U$42,85/RSA/Segundo
nShield F3	U$55,00/RSA/Segundo
nForce 1600 PCI	U$8,75/RSA/Segundo

Vale lembrar que esta é uma forma muito utilizada pelos gerentes de TI, mas não recomendada para ambientes de segurança da informação, pois devemos levar em conta vários outros fatores, além do custo de aquisição, dentre eles os custos de operacionalização, depreciação tecnológica e valor da informação a ser protegida.

4.10 Conclusões

Este capítulo demonstra a grande gama de produtos existentes no mercado, focando explicitamente os compatíveis com sistemas abertos e com uso propício para ambiente de ICP.

Em geral, os equipamentos existentes no mercado se diferenciam principalmente por foco de atuação, onde vemos os equipamentos da IBM mais voltados para o sistema financeiro, e portanto com um preço mais acessível através de subsídios e de uma economia de escala. Também vemos equipamentos voltados para o mercado de aceleração criptográfica, como os os da série nForce e os NSP, os quais possuem altos índices

de transações RSA por segundo.

Vemos alguns equipamentos que começam a se voltar para ambientes de ICP, tais como os da série nShield e os equipamentos da AEP, mas eles não levam em conta muitas das peculiaridades das ICPs, tais como controle de acesso as chaves e mecanismos de auditoria e rastreabilidade, sendo também muito vendidos para ambientes com exigência de aceleração criptográfica.

Por fim vemos de forma sumarizada um comparativo abrangendo as funcionalidades e os custos relativos de cada equipamento, dando assim uma melhor visão de como o mercado seleciona este equipamentos.

Capítulo 5

Projeto do PSC

Como parte do trabalho, propomos o projeto de um PSC completo, incluindo hardware, software e sistemas de comunicação. Devido a limitações do escopo do trabalho, só é detalhado e implementado como um protótipo o mecanismo de software que faz o gerenciamento de chaves. A escolha da implementação deste mecanismo unicamente deve-se a dificuldades orçamentária e à falta de "expertise"para a construção do hardware. As questões relativas a implementação de mecanismos de hardware podem ser sugeridas como trabalhos futuros e seguintes a este.

O projeto do PSC consiste basicamente na especificação de 3 sub-projetos:

- O Projeto Físico, que se destina à implementação do hardware e das funções criptográficas implementadas nele, definindo o meio físico sobre o qual irá operar o PSC;
- O Projeto Lógico, que consiste no projeto e avaliação dos mecanismos de software que serão executados pelo módulo, incluindo firmware, sistema operacional, o sistema de gerenciamento de chaves e rotinas criptográficas implementadas em software;
- O terceiro e último sub-projeto destina-se ao teste e às homologações necessárias ao módulo para que ele se adeque aos principais padrões internacionais que regem hoje a fabricação deste tipo de equipamento em âmbito mundial.

Adicionalmente, neste capítulo vamos ver os requisitos funcionais e não

funcionais do PSC, especificando as características desejadas para um PSC para o provimento de serviços para ICP.

No presente capítulo temos na seção 5.1 a especificação dos requisitos funcionais e na seção 5.2 a especificação dos requisitos não funcionais.

Na seção 5.3 veremos o detalhamento da especificação do hardware necessário para que o PSC atenda aos requisitos para obtenção do nível 3 da FIPS PUB 140-2, dando ênfase para as questões estruturais e de ligação dos componentes internos para garantir as barreiras criptográficas entre os componentes confiáveis e não confiáveis no sistema.

Na seção 5.4 detalhamos os componentes de software necessários para o PSC, dentre eles o sistema operacional embarcado e o sistema gestor de chaves implementado de acordo com o especificado no capítulo 6

Por fim, a seção 5.5 mostraremos os mecanismos de teste que vão garantir a conformidade com as normas e procedimentos internacionais para a construção de dispositivos criptográficos.

5.1 Requisitos Funcionais do PSC

Os requisitos funcionais são declarações que expressam quais funcionalidades um sistema deve possuir, como ele deve reagir e como ele deve se comportar em variadas situações. Eles também podem explicitamente declarar o que o sistema não deve fazer [46]. Os requisitos aqui apresentados levam em consideração as exigências das normas as quais o PSC deve aderir. São aqui detalhados unicamente os requisitos relevantes aos ambientes de ICP onde o PSC estará inserido. No caso do PSC, os requisitos funcionais são:

- Gerar chaves criptográficas;
- Gerenciar as chaves geradas, no seu uso, armazenamento e destruição;
- Proteger as chaves criptográficas da exposição ao ambiente externo ao equipamento;
- Possuir mecanismos de evidência e resposta à intrusão ou utilização indevida;

- Possuir resistência física a ataques em seu revestimento;
- Comunicar-se de forma cifrada com o meio externo;
- Autenticar os usuários com pelo menos duas técnicas de autenticação;
- Possuir um sistema de registros para auditoria;
- Trabalhar com conjuntos de usuários;
- Relacionar usuários com chave criptográficas;
- Possuir uma política de uso de chaves criptográficas;
- Ter uma rotina para cópias de segurança;
- Impossibilitar a execução de instâncias paralelas do PSC;
- Trabalhar hierarquicamente com outros PSCs;

5.2 Requisitos Não Funcionais do PSC

Os requisitos não funcionais são restrições sobre os serviços ou funções oferecidos pelo sistema, focando principalmente nas restrições de tempo, processo, padrões entre outros [46].

- Propiciar a geração de chaves criptográficas em gerador próprio;
- Estar em conformidade com a norma FIPS PUB 140-2 nível 3;
- Ter capacidade de processamento adequada para a operação de autoridades certificadoras;
- Usar software livre para o desenvolvimento;
- Usar bibliotecas livres para o desenvolvimento;
- Ser escrito na linguagem C ou C++;
- Deve gerar uma chave RSA de 4096 bits em menos de 30 minutos;
- Deve efetuar uma assinatura digital em menos de 5 segundos;

- Deve suportar mais de um algoritmo simétrico de criptografia;
- Deve se comunicar externamente usando TCP/IP;
- A interface de comunicação deve ser Ethernet ou USB;

5.3 Projeto Físico

No projeto físico do módulo de hardware criptográfico, nos cabe definir as principais necessidades do software que o hardware deve suprir assim como um prévio esboço das principais entidades de hardware necessárias para o alcance destes objetivos.

Serão levados em consideração os requisitos mínimos e a adequação do equipamento à norma FIPS PUB 140-2 nível 3.

5.3.1 Requisitos Funcionais do Hardware

Definição: DPCPC - O Dispositivo Próprio com Capacidade de Processamento Criptográfico, é um sistema autônomo com capacidade de processamento criptográfico e armazenamento seguro de chaves, tal como um smart-card ou um token USB.

Como requisitos funcionais impostos pela normas e sugeridos por vários fabricantes hoje no mercado, assim como em artigos publicados na área [47], o módulo deve:

- detectar invasões físicas ao dispositivo por meio de sensores que atuem na monitoração do ambiente externo e interno do módulo criptográfico;
- o invólucro do módulo deve ser resistente e acoplado aos componentes internos, para que sua remoção, resulte sem a menor dúvida, na inutilização do módulo;
- responder a variações anormais no fluxo de corrente e tensão assim como de temperatura, e emissão eletromagnética;
- possuir um circuito zerador dos parâmetros críticos de segurança - PCS;
- ter uma interface de comunicação com o meio externo que possibilite a separação dos dados de entrada e saída através de distintos canais físicos ou lógicos;

- prover um mecanismo de cópia de segurança das chaves criptográficas;
- prover meios de autenticação dos operadores através de dispositivos físicos, tais como DPCPCs;
- possuir um circuito dedicado para a geração de números aleatórios e pseudo-aleatórios;
- possuir mecanismos para que todos os processos sejam devidamente finalizados e as proteções acionadas, mesmo sob condições de degradação da fonte de alimentação ou do ambiente operacional do módulo;
- possuir um sistema de registros de tentativas de invasão, fisicamente inviolável, e apagável somente por processo físico-químico;
- possuir memória não volátil para o armazenamento dos dados;
- possuir fonte de alimentação própria interna ou externa;
- possuir um sub-sistema de indicadores visuais para o acompanhamento do estado interno ou operação, tal como um LED , ou mostradores digitais;
- possuir uma barreira física que isole os processos criptográficos dos processos de comunicação, operação e verificação;
- possuir memórias separadas para processos criptográficos e processos de comunicação, operação e verificação;

5.3.2 Requisitos Não Funcionais do Hardware

Nesta seção são apresentados os requisitos não funcionais relevantes a construção do hardware criptográfico para o PSC.Como requisitos não funcionais o PSC deve:

- possuir um invólucro opaco e selado;
- aderir a norma FIPS PUB 140-2 nível 3 em todos os requisitos;
- aderir aos Critérios Comuns e ao Perfil de Proteção de dispositivos de assinatura digital, ambos concordantes com o Nível de avaliação EAL 4;

- possuir um processador de arquitetura compatível com sistemas operacionais livres e de código aberto, e passível da introdução de algoritmos criptográficos em hardware;

- ser construído com peças e componentes disponíveis amplamente no mercado e implementados por mais de um fabricante.

5.3.3 Projeto Lógico de Hardware

Seguindo as características apresentadas, sugere-se a implementação inicialmente prototipada de um equipamento usando os módulos de hardware da arquitetura PC-104/PLUS ou placas integradas específicas para a construção de protótipos de equipamentos embarcados, os quais atendem o requisito não funcional de ampla disponibilidade no mercado e permitem a ligação de sensores mais simples. A especificação final do hardware do módulo criptográfico deverá ser implementada em placa específica, desenvolvida para o fim exclusivo de uso em um módulo criptográfico. Para os requisitos de invólucro, as características de dureza e resistência à invasão, necessários ao atendimento das normas, tais como a fusão do invólucro com componentes que armazenam a chave privada do módulo, serão esclarecidos no decorrer da implementação do projeto.

Para atender aos demais requisitos, sugere-se uma arquitetura semelhante à ilustrada na Figura 5.1.

No diagrama da Figura 5.1, os componentes foram agrupados, para uma melhor representação e entendimento, ficando assim separados em 5 grupos de componentes:

- Contorno Tracejado: Partes internas da barreira criptográfica;

- Contorno Tracejado-Pontilhado: Mecanismos de detecção de intrusão física e monitoração do ambiente externo;

- Preenchimento Preto: Mecanismos de alimentação;

- Sem Preenchimento: Mecanismos para o sistema operacional do equipamento;

- Contorno Pontilhado: Interfaces de entrada e saída de dados.

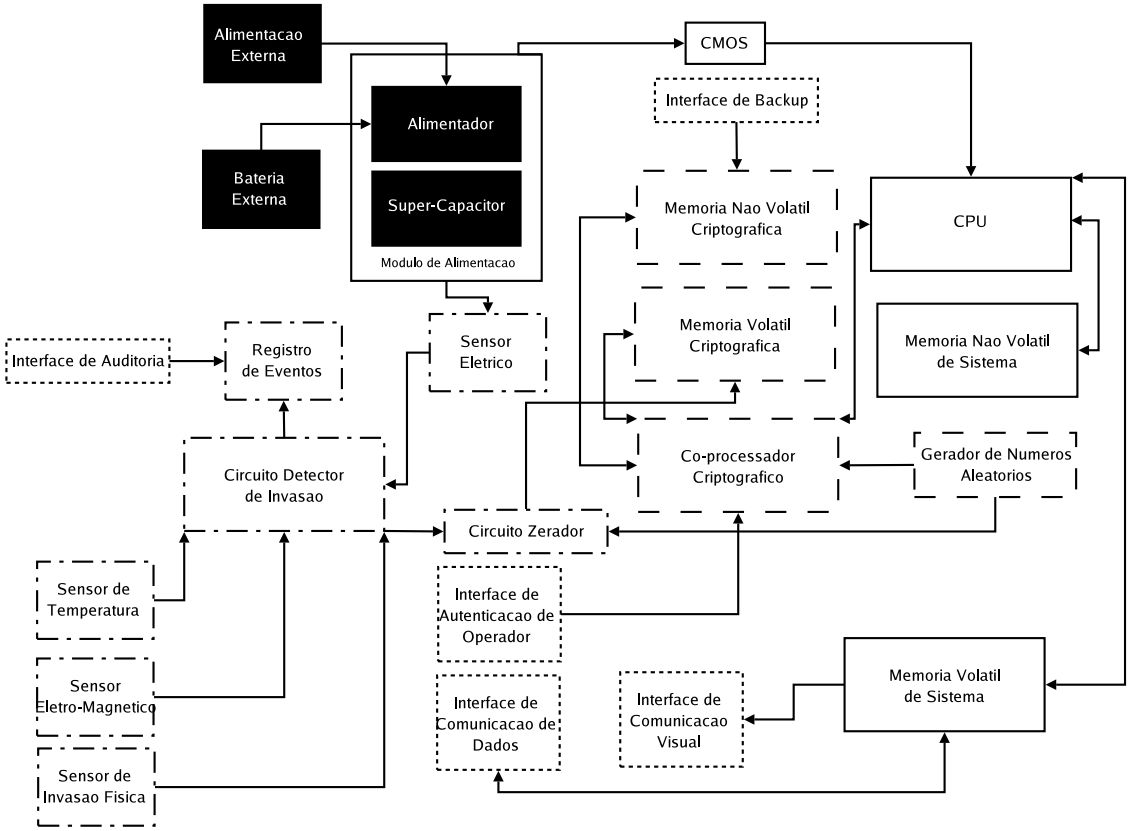


Figura 5.1: Diagrama de componentes de hardware

5.3.4 Detalhamento dos principais componentes de hardware

Os principais componentes do módulo de hardware criptográfico são os incluídos nas interfaces, no sensoreamento, no processador funcional e no meio interno da barreira criptográfica.

A barreira criptográfica deve contar com um co-processador criptográfico, não necessariamente de arquitetura Intel 8086, que seja capaz de implementar as rotinas necessárias aos algoritmos criptográficos, assim como os mecanismos para a gerência do ciclo de vida das chaves criptográficas armazenadas dentro do domínio da barreira criptográfica. Este co-processador deve possuir internamente as implementações dos algoritmos criptográficos, os quais forem mais eficientes em hardware, e ser suficientemente veloz para executar as operações criptográficas requeridas pelo módulo.

A memória interna à barreira criptográfica deve possuir comunicação direta e exclusiva com o co-processador criptográfico, o qual deve se comunicar com os processos exclusivamente a partir do processador principal do módulo, excluindo-se qualquer comunicação com as partes internas da barreira criptográfica por parte de outros

mecanismos, excetuando-se o mecanismo de cópia de segurança das chaves criptográficas e do estado interno do PSC.

O acesso através da interface de cópia de segurança das chaves criptográficas e do estado interno do PSC deve ser dado exclusivamente à memória não volátil criptográfica e à memória não volátil do sistema, onde não existem parâmetros críticos de segurança - PCS, pois nela todos os PCS estarão gravados na forma cifrada.

Definição: PCS - Parâmetro Crítico de Segurança é qualquer dado relacionado com as chaves criptográficas protegidas pelo PSC que pode ser acessado diretamente na forma não cifrada.

O gerador de números aleatórios deve alimentar o co-processador criptográfico e o circuito zerador com números aleatórios.

O circuito zerador é responsável pelo apagamento efetivo dos PCS das memórias voláteis e é um componente externo à barreira criptográfica, sendo ele a única comunicação do gerador de números aleatórios com o meio externo a esta barreira.

As interfaces de autenticação do operador devem permitir o uso de DPCPCs, os quais devem conter as chaves privadas dos usuários e seus respectivos certificados digitais.

A interface de comunicação de dados deve ser capaz de criar canais lógicos para a separação dos dados de controle, tais como comando, dos dados de entrada, tais como resumos para a assinatura, e dos de saída, tais como resumos assinados. Atendendo ao requisito de uso de uma interface USB ou Ethernet, a interface ainda é amplamente disponível nos equipamentos hoje disponíveis no mercado

A interface de comunicação visual, deve ter capacidade de informar aos usuários o estado interno do sistema, assim como deve prover os mecanismos para entradas de dados tais como teclados para digitação das senhas de autenticação dos usuários, ou para a configuração, inicialização e finalização do módulo criptográfico. Os visores devem informar unicamente se o sistema está ativo e qual processo ele está executando no momento, para dar transparência ao processo executado pelo PSC. As interfaces de entrada de dados podem ser substituidas por canais de comunicação com a máquina hospedeira, fazendo assim com que as entradas de senhas e parâmetros de configuracão possam ser obtidos a partir deles.

A interface de auditoria tem que ser capaz de acessar unicamente o conteúdo da memória não apagável do sistema de registro de eventos para extração de registros de auditoria. Isto deve ser possível mesmo se o PSC estiver inoperante. No caso de utilização total da memória de registros, o PSC deve se tornar inoperante, sendo necessária a imediata extração dos registros para que ele possa voltar a funcionar.

Os sensores do sistema de detecção de intrusão e monitoração do ambiente devem estar diretamente ligados ao circuito de detecção de invasão. Os sensores serão detalhados na seção 5.3.5.

O circuito de detecção de invasão deve sempre acionar o circuito zerador, e inserir no sistema de registro de eventos qualquer anormalidade.

O circuito zerador, ao ser acionado pelo circuito de detecção de invasão, deve escrever padrões aleatórios na memória volátil criptográfica, de forma a tornar impossível a recuperação dos parâmetros críticos de segurança que lá estavam armazenados. Os parâmetros não críticos, ou seja, os que estiverem na forma cifrada, não precisam ser apagados. Podem existir dois limiares no sensor, um que apaga os PCS e outro, que na detecção de persistência do ataque, apaga os parâmetros cifrados.

5.3.5 Sensores

Os sensores são utilizados para detectar toda e qualquer tentativa de violação física do PSC. Os valores básicos para a detecção podem ser determinados pelo usuário do PSC. Os valores para o limiares sugeridos nesta seção estão seguindo as recomendações dos atuais fabricantes de equipamentos do gênero e não possuem uma justificativa científica. Eles estão presentes somente para servirem como medida base. Os sensores são:

- Sensor de Remoção da Tampa: Este sensor deve detectar a abertura da tampa que dá acesso aos componentes internos, principalmente àqueles internos à barreira criptográfica;

- Sensor de Luminosidade Interna: Este sensor deve servir como uma contingência ao sensor de Remoção da Tampa, pois qualquer abertura no invólucro deixará penetrar luz. O sensor deverá ser capaz de detectar variações de luminosidade de 0,01 lux;

- Sensor de Temperatura: Este sensor deve ser capaz de detectar variações de temperatura e deve ser especificado para o ambiente de operação ao qual o módulo se destina, nunca deixando a temperatura ser superior ao suportado pelo componentes. O sensor deve ter precisão de 0,1 graus Celsius;

- Sensor de Variação Elétrica: Este sensor deve ser capaz de detectar variações de corrente e tensão da rede elétrica na qual o módulo está ligado, assim como monitorar as mesmas características na rede lógica, fazendo com que variações superiores a 5% nos valores nominais de operação sejam suficientes para acioná-lo;

- Sensor de Vibração: Este sensor deve ser capaz de detectar vibrações na superfície do módulo a fim de detectar perfuração ou vibração que possa comprometer algum componente interno;

- Sensor de Pressão Atmosférica: Este sensor deve ser capaz de detectar variações na pressão externa ao módulo, mantendo seus limiares baseados nos ambientes habitáveis da superfície do planeta;

- Sensor de Emissão Eletromagnética: Este sensor deve ser capaz de detectar as variações de emissão eletromagnética provindas do ambiente externo, tolerando variações não superiores a 20% da emissão gerada pelo módulo no seu funcionamento.

5.4 Projeto Lógico do Software

Cabe ao projeto Lógico do Software a definição dos mecanismos de software que serão executados pelo hardware, assim como os componentes de software que executarão internamente nos componentes físicos do sistema, tais como os algoritmos internos do co-processador criptográfico.

Esta seção está dividida na sub-seção 5.4.1, a qual trata dos componentes básicos de software para que a aplicação gestora de chaves possa executar no equipamento desenvolvido para o PSC. Na sub-seção 5.4.2, temos o detalhamento das funções criptográficas que o PSC vai executar, ressaltando que as mesmas serão implementadas por mecanismos de software, deixando para a execução em hardware somente a geração de números aleatórios. A subseção 5.4.3 mostrará os níveis de execução para a opera-

cionalização do PSC, e a subseção 5.4.4 mostrará os requisitos para o gerenciamento de chaves no PSC.

5.4.1 Software Básico

Todos os softwares básicos do PSC serão desenvolvidos, a partir de especificações e normatizações internacionais. A grande ênfase será no desenvolvimento dos sistemas embarcados no PSC, tais como um sistema gestor do ciclo de vida de chaves criptográficas, uma vez que o restante dos softwares ficarão a cargo do sistema operacional OpenBSD, o qual rodará uma versão personalizada para a aplicação de gerência de chaves do PSC, conforme especificação posterior na seção 5.4.4.

O OpenBSD é um sistema operacional multi-plataforma, baseado nas definições POSIX de interoperabilidade de sistemas operacionais, e um variante do Unix. É um sistema operacional bastante conhecido ter como característica de projeto a preocupação pró-ativa com segurança, fazendo para tal, esforços que vão desde a implementação de algoritmos de criptografia integrados no sistema operacional até a auditoria incessante de todo o código usado no projeto [48].

As preocupações existentes no projeto OpenBSD não se restringem unicamente à implementação de um sistema operacional seguro, mas também à disponibilidade deste sistema para todos. Por causa deste propósito, o projeto hoje é sediado no Canadá e é desenvolvido em vários países, principalmente naqueles que não impõem restrições à exportação de criptografia.

Serão necessárias adaptações no sistema operacional, para que o suporte dado ao co-processador criptográfico seja diretamente integrado ao sistema.

O projeto OpenBSD se encaixa muito bem em inúmeras aplicações nas quais a segurança pró-ativa é necessária, mas mesmo assim ele deixa a desejar quando queremos mais garantias durante a execução de código, principalmente garantias de que o código não foi modificado por pessoas não autorizadas. Pensado nisso, Mike Schiffman criou um conjunto de correções para serem aplicadas no OpenBSD para a criação de um ambiente de execução de código assinado digitalmente, chamado Stephanie [49].

O software básico também contará com a implementação de um motor de ligação para o OpenSSL, o qual coordenará a comunicação do módulo com o sistema de operação da Autoridade Certificadora que o estiver utilizando. Este motor de ligação,

também conhecido como "engine", fará com que qualquer aplicação já compatível com a API da biblioteca OpenSSL possa facilmente se comunicar com o PSC desenvolvido neste trabalho.

5.4.2 Funções Criptográficas

A seleção das operações criptográficas dependerá das aplicações que usarão o PSC. No entanto existe um conjunto de operações que são básicas e que são utilizadas pela maioria das aplicações. Em particular, neste trabalho estamos interessados nas operações utilizadas pelas ACs e ARs de uma ICP.

Para melhor classificar, vamos dividir os serviços criptográficos prestados em 4 grupos:

- as funções resumo-criptográficas, listadas na Tabela 5.1,
- os algoritmos de autenticação, listados na Tabela 5.2,
- os algoritmos criptográficos simétricos, listados na Tabela 5.3,
- os algoritmos criptográficos assimétricos, listados na Tabela 5.4.

Tabela 5.1: Funções resumo-criptográficas

Nome:	**Referência:**
MD2	RFC 1319 [50]
MD5	RFC 1321 [51]
SHA-256	FIPS PUB 180-2 [7]
SHA-384	FIPS PUB 180-2 [7]
SHA-512	FIPS PUB 180-2 [7]
RIPMED160	Paper: RIPEMD-160, a strengthened version of RIPEMD [52]

5.4.3 Níveis de Execução

No PSC proposto contaremos com um gerenciamento de chaves o qual é realizado pelo seguinte conjunto de entidades:

- Um conjunto de Administradores, responsáveis por tarefas administrativas, excetuando-se as atividades de auditoria;

Tabela 5.2: Algoritmos de autenticação

Nome:	**Referência:**
DES MAC	FIPS PUB 81 [4]
Triple DES MAC	ISO 8372 [53]
HMAC MD2	FIPS PUB 198 [54]
HMAC MD5	FIPS PUB 198 [54]
HMAC SHA-256	FIPS PUB 198 [54]
HMAC SHA-384	FIPS PUB 198 [54]
HMAC SHA-512	FIPS PUB 198 [54]
HMAC RIPMED160	FIPS PUB 198 [54]

Tabela 5.3: Algoritmos criptográficos simétricos

Nome:	**Referência:**
DES	FIPS PUB 46 [2]
Triple DES	FIPS PUB 46 [55]
AES	FIPS PUB 197 [3]
CAST-128	RFC 2144 [56]
CAST-256	RFC 2612 [57]
Blowfish	Paper: Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish) [58]
Twofish	Paper: Twofish: A 128-Bit Block Cipher [59]
Serpent	Paper: Serpent: A Proposal for the Advanced Encryption Standard [60]

- Um ou mais conjuntos de Operadores, responsáveis por chaves;
- Um conjunto de auditores, responsáveis pela auditoria dos eventos ocorridos no provedor.

Devido às diferentes políticas e permissões associadas às entidades, é adequado organizar o acesso às funcionalidades em níveis de execução. Cinco níveis são suficientes para tal, sendo designado um deles para procedimento de inicialização e manutenção do PSC e quatro para a operação do PSC.

Tabela 5.4: Algoritmos criptográficos assimétricos

Nome:	**Referência:**
DSA	FIPS PUB 186 [61]
RSA	Paper: A Method for Obtainig Digital Signatures and Public-Key Cryptosystems [9]
El-Gamal	Paper: A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms [62]
Diffie-Hellman	Paper: New Directions in Cryptography [8]

5.4.3.1 Nível 1 - Nível de Testes

No primeiro nível, o de inicialização e testes, deve-se verificar se todos os componentes de hardware e de software são validos, garantindo, através de assinaturas digitais, que nada está comprometido.

Os procedimentos de teste do funcionamento do hardware devem garantir a qualidade e operacionalidade dos algoritmos criptográficos, assim como a entropia do gerador de números pseudo-aleatórios, além obviamente de todo o restante do equipamento, inclusive sensores de detecção de invasão.

Já os procedimentos de software devem garantir que o PSC valide todos os códigos a serem executados. Através de um certificado embarcado em componentes não passíveis de escrita, tais como a memórias ROM, o módulo verifica se todas as assinaturas dos softwares presentes são válidas.

A confiabilidade neste certificado pode ser obtida através da publicação dele em um meio de imprensa físico, tal como um jornal, atestando que ele está relacionado com o número de série do PSC.

Uma vez validado o PSC, incluindo software e hardware, ele estará apto a passar para o nível dois. O primeiro nível de inicialização é executado a cada reinício do PSC

5.4.3.2 Nível 2 - Nível Administrativo

O segundo nível de inicialização trata da geração do conjunto de administradores, sendo neste momento gerado um par de chaves de autenticação pelo módulo, gerando um certificado, que é auto-assinado.

Cada administrador que já não possua um certificado reconhecido pelo PSC, gera um par de chaves, e submete sua chave pública para assinatura. Aqueles que possuem um certificado simplesmente o informam ao PSC.

A chave privada do PSC é cifrada através de um algoritmo simétrico de ciframento com uma chave aleatória, a qual é passada por um gerador de parcelas de segredo compartilhado. Cada parcela é cifrada com a chave pública do certificado de um administrador. As parcelas cifradas são armazenadas.

O uso do gerador de parcelas de segredo compartilhado faz com que a reconstrução da chave privada do PSC possa se dar por um sub-conjunto do conjunto de

administradores, sendo detalhado este procedimento na seção 6.2.

Da mesma forma, durante este nível de execução deverão ser criados os auditores do PSC, os quais são responsáveis pela verificação das atividades realizadas pelo PSC. Os auditores são criados da mesma forma que os administradores, mas possuem um par de chaves de controle de auditoria com o qual são cifrados todos os eventos realizados pelo PSC, de acordo com a seção 6.6. Não existe subordinação entre os auditores e os administradores.

Uma vez completado o segundo nível de segurança de inicialização do PSC, ele estará apto a garantir toda a execução de código, assim como autenticar os administradores e registrar os seus eventos. Este nível de segurança só é executado quando da inicialização primária do módulo, ou no caso do comprometimento de algum administrador, sendo aí adotados os mecanismos propostos na seção 6.7.

5.4.3.3 Nível 3 - Nível Criação de Chaves/Operadores

No terceiro nível de segurança serão geradas as chaves de aplicação protegidas pelo PSC. Antes da geração das chaves de aplicação devem ser gerados os conjuntos de operadores que utilizarão estas chaves.

A geração de um conjunto de operadores implica na geração de uma chave simétrica para cifrar as chaves de aplicação gerenciadas pelo conjunto. Esta chave simétrica é dividida em partes por um gerador de parcelas de segredo compartilhado.

Cada operador que não possua já um certificado reconhecido pelo PSC, gera um par de chaves, e submete sua chave pública para assinatura. Aqueles que possuem um certificado simplesmente o informam ao PSC. Cada parcela é cifrada com a chave pública do certificado de um administrador. As parcelas cifradas são armazenadas.

O uso do gerador de parcelas de segredo compartilhado faz com que a reconstrução da chave simétrica do conjunto de operadores possa se dar por um subconjunto do conjunto de operadores, sendo detalhado este procedimento na seção 6.5.

O processo de geração de chaves é uma atividade administrativa e, portanto, é necessário que os administradores cifrem a chave de aplicação para uso dos operadores. No momento de criação dos operadores, como descrito na seção 6.3, é criado um mecanismo de compartilhamento de sua chave simétrica com os administradores.

Os administradores então geram um par de chaves e cifram a chave

privada com a chave simétrica do conjunto de operadores que podem gerenciar esta chave, conforme a seção 6.4.

Completados os procedimentos deste nível, o PSC passa ao nível 4 operando sobre uma plataforma verificada e validada e com os mecanismos de proteção ao acesso da chave privada das aplicações garantidos por mecanismos de controle de acesso baseados em conjuntos de operadores e administradores, sendo todo o processo registrado para posterior auditoria.

5.4.3.4 Nível 4 - Nível de Operação

O quarto nível de segurança do PSC é atingido quando da operação, e é destinado ao provimento de serviços pelo PSC ao ambiente externo. Este nível não é um nível de inicialização e só pode ser alcançado após pelo menos uma execução de cada um dos níveis anteriores.

Os serviços providos neste nível são assinaturas externas ao módulo, as quais consistem em liberar o acesso à chave privada de alguma autoridade certificadora, a qual só pode ser obtida pelo consentimento de um subconjunto do conjunto de operadores.

5.4.3.5 Nível 5 - Nível de Auditoria

No quinto nível de segurança são executadas as atividades de auditoria, onde os auditores podem extrair registros de execuções anteriores e exportá-los para o exterior do PSC. Neste nível também podem ser apagados registros de auditoria pelo conjunto de auditores.

O alcance deste nível consiste em tornar o PSC inoperante, até que o mesmo retorne ao nível 4, visto que neste nível os processos de auditoria inviabilizam o provimento de serviços criptográficos.

5.4.4 Gerenciamento de Chaves

Os mecanismos de gerenciamento de chaves criptográficas de um PSC devem possuir uma estrutura em camadas de forma que os processos e as chaves sejam única e exclusivamente utilizados através de autorização e autenticação nos momentos adequados.

A especificação dos mecanismos de gerenciamento de chaves, as entidades e os processos relativos são dados no capítulo 6.

5.5 Projeto de Testes

O projeto de testes consiste na execução de métodos para a validação do total funcionamento do equipamento, para que seja atestado que o mesmo é capaz de ser submetido com êxito à normatização por um órgão competente. No projeto de testes estão incluídos testes de integração com um software que fará o gerenciamento de ICP. Os testes incluem, a geração de chaves, assinatura de certificados, publicação de listas de certificados revogados, entre outras operações necessárias ao bom funcionamento de uma ICP. Também será testado o OpenSSL, para que o mesmo seja usado como um mecanismo independente de ambiente.

Os testes devem incluir todo e qualquer ataque efetivo e conhecido que possa vir a comprometer o mecanismo nas suas funções definidas como requisitos funcionais.

Os testes deverão ser realizados de acordo com as especificações dos critérios comuns, levando em consideração a obtenção do nível de garantia EAL4. Serão alvos dos testes os componentes individuais na sua forma mais atômica possível, a integração dos componentes, tratando os quesitos exigidos pelo CC, e também testes de produção, levando em conta todo o ambiente operacional onde o PSC estará inserido.

Os testes devem ser realizados em cada componente de forma independente, e por sua natureza, não devem ser realizados pelo mesmo grupo que se propôs ao desenvolvimento do componente testado.

Se for encontrado um problema, o mesmo deverá ser mensurado e classificado de acordo com a sua urgência e comprometimento dos requisitos funcionais. Os testes sempre serão realizados em grupos ou baterias, as quais podem testar inúmeros fatores de cada componente. Ao término de uma bateria de testes, os resultados devem ser submetidos ao grupo de desenvolvimento do respectivo módulo para análise e correção.

Os procedimentos de testes devem sempre ser realizados por no mínimo duas equipes independentes, as quais, ao término da bateria de testes, devem emitir o seu parecer, qualificando o atendimento aos requisitos e à segurança do componente. A

especificação do critérios de testes será dada tanto pelo desenvolvedor quanto pela equipe responsável pelo teste, sempre de comum acordo quanto à relevância dos mesmos.

Uma vez testados todos os componentes, serão feitos os testes de integração dos componentes. Os testes serão dirigidos a comunicação e iteração entre os componentes. Sendo detectados problemas, os mesmos devem ser classificados e submetidos para todos os grupos envolvidos no desenvolvimento dos componentes, para que sejam apuradas as competências na resolução do problema detectado.

Uma vez concluídos os teste de integração, o PSC deve ser submetido a um teste de produção, onde deve ser simulado o ambiente de produção no qual o PSC deverá ser incluído.

Nos testes de produção, o PSC deve operar com todas as funcionalidades necessárias para o cumprimento das suas funções, e no caso da detecção de um problema, o mesmo deve ser encaminhado para a coordenação do projeto, a qual reunirá os grupos para a determinação do foco do problema, as possíveis soluções, as mudanças de projeto necessárias, as execuções, os testes, e por fim a liberação do equipamento para a continuidade dos teste de produção.

No fim do processo de testes, os resultados obtidos, incluindo as falhas detectadas durante todos os procedimentos, devem ser publicadas para o conhecimento da comunidade acadêmica, com o fim de garantir a idoneidade do processo e obter sugestões de ataques possivelmente não testados.

5.6 Conclusões

O presente capítulo tratou dos detalhes inerentes ao projeto de implementação do PSC, dando clara ênfase nos métodos de execução para o projeto.

A divisão do projeto em 3 partes visa facilitar o entendimento e delimitar o que efetivamente o trabalho cobriu. Os requisitos do PSC foram definidos levando em conta as necessidades de uma ICP. Foram detalhados os requisitos de hardware para garantir as devidas proteções físicas ao PSC.

O foco do trabalho é a construção do software de gerenciamento do ciclo de vida das chave criptográficas, o qual é detalhado no capítulo 6, e neste capítulo de projeto vimos o detalhamento do projeto de software incluindo o sistema operacio-

nal embarcado, as funções criptográficas necessárias, e os níveis de operacionalização necessários para o funcionamento do PSC.

Finalizando o capítulo é definido um projeto de testes, o qual tem por objetivo a garantia da idoneidade do processo de construção do PSC.

Capítulo 6

Gerenciamento de Chaves Criptográficas no PSC

A construção de provedores de serviços criptográficos, PSC, sejam eles embarcados ou não, envolve a determinação de mecanismos para o gerenciamento do ciclo de vida das chaves criptográficas. Este gerenciamento é dado através de mecanismos que geram, protegem e destroem as chaves de forma segura.

Os PSCs embarcados, por sua vez, possuem peculiaridades na forma como eles gerenciam as chaves, e como permitem operá-las. Estes provedores são usados normalmente em ambientes com necessidades de segurança elevadas.

A seção 6.1 descreve os procedimentos para o gerenciamento do ciclo de vida de chaves criptográficas e estabelece os requisitos para a construção de um mecanismo eficiente para tal. A seção 6.2 descreve o procedimento de configuração inicial deste mecanismo, consistindo da criação das chaves do PSC e da criação do conjunto de administradores. A seção seguinte (6.3) descreve o processo de criação do conjunto de administradores e as informações necessárias para a geração das chaves a eles associadas. Na seção 6.4 completamos o procedimento inicial para a operacionalização do PSC, que é a geração de chaves assimétricas para uso em aplicações.

No decorrer da seção 6.5, vemos os procedimentos necessários para o uso das chaves assimétricas protegidas pelo PSC, e na seção 6.6, vemos os procedimentos de criação de auditores, os quais atuam de forma a monitorar todos os processos executados pelo PSC.

As seções 6.7 e 6.8 descrevem os processos administrativos de troca

do conjunto de administradores e troca de propriedade de chaves entre os conjuntos de operadores do PSC. Por fim, na seção 6.9, é descrito o procedimento de criação de cópias de segurança e na seção seguinte (6.10) é descrito o processo de recuperação das cópias de segurança no caso de um desastre. A última seção, 6.11, descreve as funcionalidades de subordinação do PSC numa estrutura de ICP já existente através da importação de certificados confiáveis ao provedor.

6.1 Gerenciamento do Ciclo de Vida de Chaves Criptográficas

O ciclo de vida de chaves criptográficas compreende desde a criação de uma chave, o seu uso e a sua correta destruição. Na análise dos ciclos de vida de chaves criptográficas devemos levar em consideração os seguintes requisitos:

- A geração de chaves - Garantindo que as chaves geradas terão as qualidades necessárias para não comprometer os algoritmos nas quais estarão sendo usadas. Isto normalmente compreende, além de gerar uma chave, testar se a mesma não é considerada fraca para o algoritmo criptográfico no qual estará sendo utilizada.

- O uso de chaves - Deve ser garantido o correto uso das chaves criptográficas, garantindo que elas estarão sempre disponíveis aos seus detentores, mesmo no caso de desastres, e além disso, deve ser garantido que o acesso será controlado e restrito a somente pessoas autorizadas.

- A auditoria do uso de chaves - Deve ser garantido a criação de um histórico completo e rastreável de tudo o que foi feito com uma chave, desde o momento de sua criação até o momento de sua destruição.

- A destruição de chaves - Deve ser garantida a correta destruição de chaves criptográficas para que as mesmas não possam ser utilizadas fora do ambiente de controle do seu ciclo de vida.

Os processos responsáveis pelo gerenciamento do ciclo de vida das chaves devem ser adequadamente projetados de forma segura e ser executados em um ambiente seguro. Para tal, na presente proposta colocamos a gerência de chaves como um

mecanismo protegido por outros mecanismos de autenticação e autorização. Estes procedimentos visam melhorar a operacionalização de ambientes de ICPs, visto que os atuais provedores atuam de forma genérica e não são focados neste tema.

Os mecanismos de gerenciamento de chaves criptográficas de um PSC devem possuir uma estrutura em camadas, de forma que os processos e as chaves sejam única e exclusivamente utilizados através de autorização e autenticação nos momentos adequados.

Ao projetarmos o mecanismo de gerenciamento do ciclo de vida de chaves criptográficas proposto neste trabalho, levamos em consideração unicamente a proteção de chaves criptográficas assimétricas. O intuito é de proteger pares de chaves assimétricas utilizados em ambientes de alto valor agregado, tais como ICPs. A estas chaves daremos o nome de chaves de aplicação, visto que as mesmas podem representar quaisquer aplicação que queira fazer uso da proteção de suas chaves assimétricas.

Ao estipularmos o gerenciamento de chaves do PSC proposto, com base nos requisitos acima detalhados, definimos 3 conjuntos de atores/entidades para a iteração com o sistema. O gerenciamento de chaves do PSC é realizado pelo seguinte conjunto de entidades:

- Um conjunto de Administradores, os quais são responsáveis por todas as tarefas administrativas relativas ao provedor, excetuando-se as atividades de auditoria;
- Um ou mais conjuntos de Operadores, os quais são os responsáveis por uma chave;
- Um conjunto de auditores, os quais são responsáveis pela auditoria dos eventos ocorridos no provedor.

Os processos associados e as responsabilidades relativas às atividades de que cada um dos grupos deve efetuar durante a sua iteração com o provedor de serviços criptográficos serão descritos nas seções que seguem.

6.2 Criação do Conjunto de Administradores

O PSC deve criar um conjunto de administradores sempre que for inicializado pela primeira vez, realizando o processo descrito na Figura 6.1.

Para a criação do conjunto de administradores, é necessário fornecer:

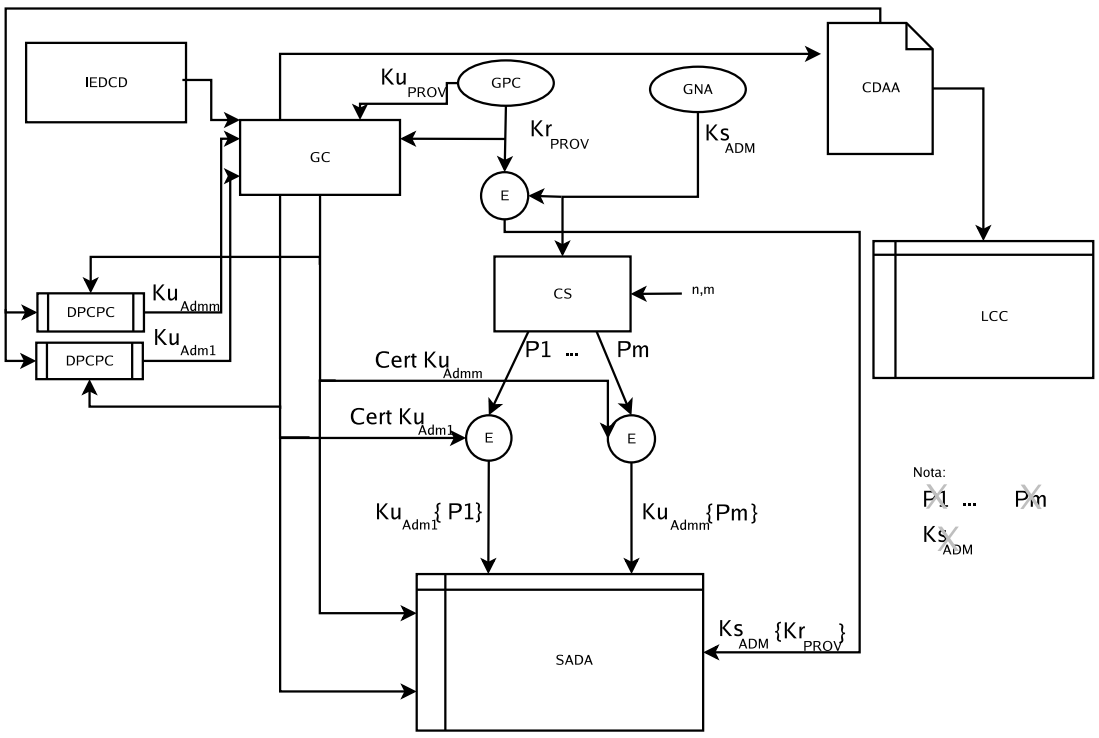


Figura 6.1: Mecanismo para Criação do Conjunto de Administradores.

- O número m de administradores do conjunto, ou seja, o tamanho do conjunto;
- O número n de administradores, sendo $1 \leq n \leq m$, mínimo necessário para realizar as atividades administrativas do PSC;
- Para os m DPCPC que deverão ser apresentados ao PSC durante a criação do conjunto de administradores, os que ainda não possuírem certificados confiáveis ao PSC, as informações necessárias à criação dos certificados digitais de cada administrador deverão ser fornecidas.

Definição: GPC - Gerador de Par de Chaves, é o componente responsável pela geração das chaves criptográficas assimétricas.

Definição: GNA - Gerador de Números Aleatórios, é o responsável pela geração de números ou cadeias de números escolhidos ao acaso e de forma imprevisível.

Definição: GC - Gerador de Certificados, é o responsável pela emissão de certificados digitais.

Definição: E - Algoritmo Simétrico de Criptografia operando no modo de ciframento, tendo como entrada um texto e uma chave, e como saída um valor cifrado.

Definição: D - Algoritmo Simétrico de Criptografia operando no modo de deciframento, tendo como entrada um valor cifrado e uma chave, e como saída um texto.

Definição: Kr_{adm_i} - Chave privada de cada administrador i armazenada no seu DPCPC.

Definição: Ku_{adm_i} - Chave pública de cada administrador i relacionada com a sua chave privada.

Definição: IEDCD - Interface de Entrada de Dados para Certificado Digital.

Definição: Kr_{PROV} - Chave privada que identifica o PSC e é protegida pelo conjunto de administradores.

Definição: Ku_{PROV} - Chave pública do PSC relacionada com a sua chave privada.

Definição: CDAA - Certificado Digital Auto-Assinado.

Definição: LCC - Lista de Certificados Confiáveis.

Definição: $CertKU_{adm_i}$ - Certificado emitido para a chave pública de cada administrador i, relacionando-a ao seu detentor. Este certificado é assinado usando Kr_{PROV}.

Definição: Ks_{ADM} - Chave simétrica que representa o conjunto de administradores e protege a chave Kr_{PROV}.

Definição: CS - Compartilhamento de segredo é a técnica de dividir um segredo em um conjunto de m partes e recompô-lo com um sub-conjunto de tamanho n tal que $1 \leq n \leq m$, usando para tal técnicas computacionais avançadas.

O processo de criação de administradores de um PSC deve seguir os seguintes passos:

1. Cada administrador i, utilizando um DPCPC próprio, deve gerar um par de chaves criptográficas: uma chave privada e uma pública. A chave privada Kr_{adm_i} deve permanecer armazenada sem possibilidade de cópia, em um DPCPC;

2. A chave pública Ku_{adm_i} deve ser disponibilizada ao PSC. Caso o administrador i já possua um certificado emitido por uma AC confiável ao PSC, ele deve informar o seu certificado;

3. Caso não o possua, seu certificado será gerado internamente ao PSC, sendo necessário então fornecer as informações que constarão no certificado do administrador através da IEDCD.Vale lembrar que ambos os tipos de certificados, sejam emitidos no PSC ou externamente, devem conter as corretas extensões para a validação dos administradores de PSC;

4. O PSC também deve gerar seu par de chaves Ku_{PROV} e Kr_{PROV}, e o respectivo CDAA. O certificado possui extensões que o definem como um PSC;

5. Este certificado será armazenado numa LCC interna ao provedor. A LCC nos possibilita, adicionalmente, a inclusão de outras autoridades certificadoras como confiáveis ao provedor, o que é plenamente justificável no caso de uma estrutura hierárquica de ICP, fazendo assim com que o provedor aceite certificados de administradores emitidos por ACs externas;

6. Caso os administradores não informem um certificado emitido por uma AC confiável ao PSC, utilizando os dados informados pelos administradores através da IEDCD, o provedor emite um certificado para Ku_{adm_i}. Os $CertKU_{adm_i}$ e o CDAA devem ficar em um subsistema interno do provedor chamado de Sistema de Armazenamento de Dados de Administradores - SADA, para fins posteriores de autenticação através de desafio;

7. Após a definição dos certificados dos administradores e do PSC conforme ilustra a Figura 6.1, deveremos gerar uma chave de sessão Ks_{ADM}, a qual será utilizada internamente para representar todo o conjunto de administradores e para cifrar a chave Kr_{PROV};

8. Uma vez cifrada, Kr_{PROV} deverá ser apagada, objetivando-se mantê-la sempre na forma cifrada dentro do PSC. $Ks_{ADM}\{Kr_{PROV}\}$ deve ser armazenada no SADA visando a recuperação de Kr_{PROV};

9. A chave Ks_{ADM} será protegida por mecanismos de compartilhamento de segredos - CS [31], onde serão sempre definidos os m administradores, dos quais no mínimo n farão a recomposição deste segredo;

10. As partes de P_1 até P_m serão cifradas de forma independente entre os administradores através de suas chaves públicas contidas nos seus certificados;

11. Uma vez cifradas as partes de P_1 até P_m, elas serão armazenadas no SADA na forma cifrada, e serão apagadas em sua forma não-cifrada junto com Ks_{ADM}. O armazenamento das partes de P_1 até P_m se dará unicamente na forma cifrada.

O uso de Ks_{ADM} para proteger Kr_{PROV} deve-se ao fato que, caso um ataque ao provedor utilizando um DPCPC mal intencionado ou comprometido tenha sucesso, o acesso será sempre a partes de Ks_{ADM} ou a $Ks_{ADM}\{Kr_{PROV}\}$ e nunca a Kr_{PROV}. A chave Kr_{PROV} nunca sai da parte interna da barreira criptográfica do provedor, tornando assim o sistema recuperável a este ataque, simplesmente trocando o conjunto de administradores e conseqüentemente a Ks_{ADM} afetada.

Os administradores serão responsáveis por processos administrativos internos ao provedor, os quais consistem em:

- Inicialização e Operacionalização do provedor.
- Configuração do provedor e determinação de todos os seus parâmetros de operação, entre os quais estão os tamanhos de chaves assimétrica, endereços de acesso, etc.
- Criação de Operadores, os quais serão os detentores e utilizadores das chaves assimétricas.
- Criação de Chaves assimétricas para uso em aplicações externas ao provedor.
- Gerência das Chaves, tais como a sua delegação a um conjunto de operadores, a sua destruição ou inutilização.
- Participação de forma ativa nos procedimentos de cópias de segurança.
- Criação do Conjunto de Auditores que vão gerenciar os processos de auditoria de registros e a sua legitimidade.

A execução destas tarefas necessitará da identificação de um subconjunto de tamanho mínimo n do conjunto de administradores.

Com os administradores efetivamente criados e aptos a configurar o provedor, temos a necessidade da execução de um novo procedimento para dar andamento à operacionalização do provedor, que é a criação do conjunto de operadores, conforme será descrito na seção 6.3.

6.3 Criação do Conjunto de Operadores

Uma tarefa dos administradores do PSC é a geração de conjuntos de operadores para que os mesmos possam fazer uso de chaves criptográficas em suas aplicações. Isto é feito realizando o processo descrito na Figura 6.2.

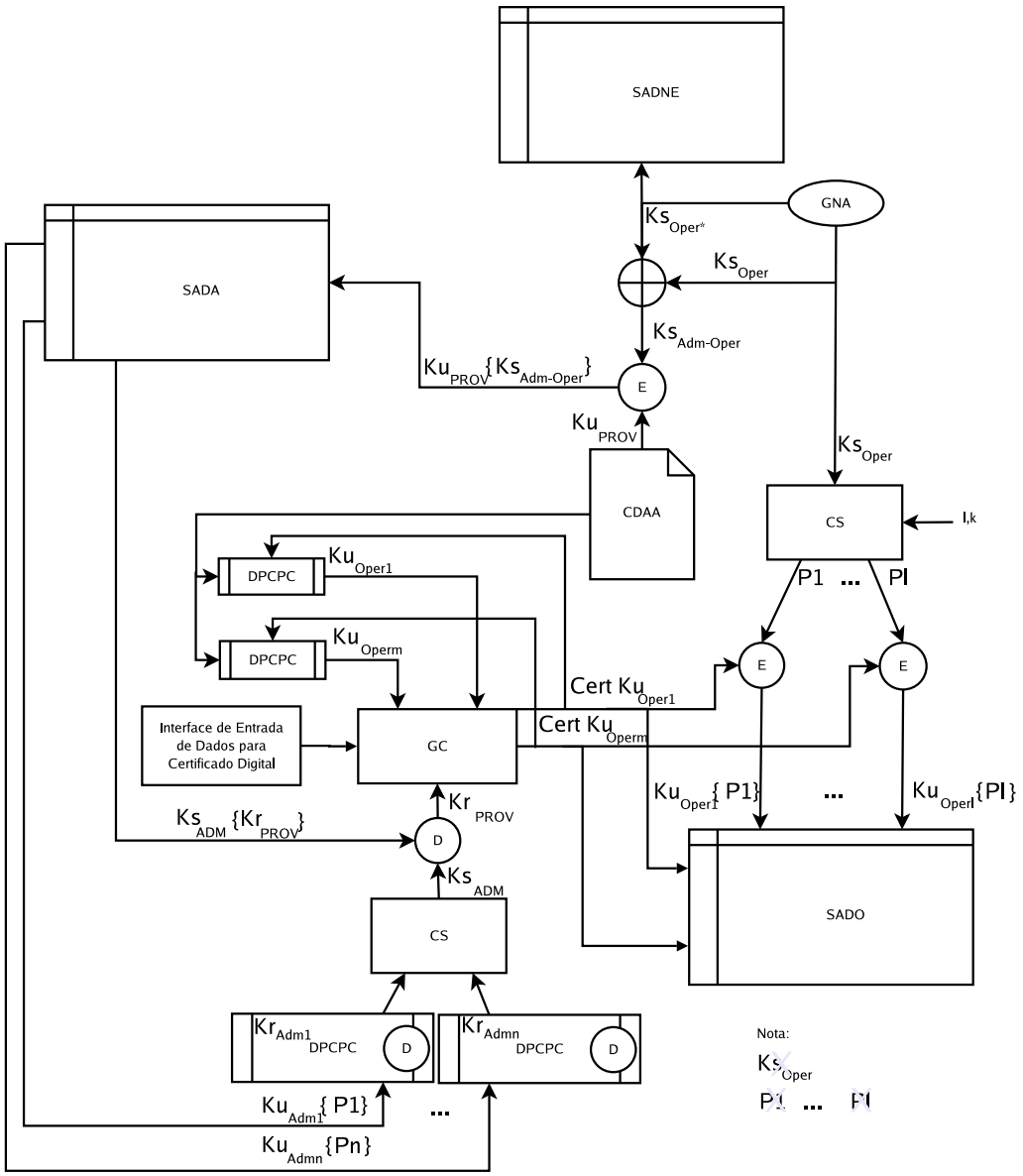


Figura 6.2: Mecanismo para Criação do Conjunto de Operadores.

Para a criação do conjunto de operadores, é necessário fornecer:

- O número k de operadores do conjunto, ou seja, o tamanho do conjunto;
- O número l de operadores, sendo $1 \leq l \leq k$, mínimo necessário para realizar as atividades relativas ao conjunto de operadores.;

- Para os k DPCPCs que deverão ser apresentados ao PSC durante a criação do conjunto de operadores, os quais ainda não possuírem certificados confiáveis ao PSC, as informações necessárias à criação dos certificados digitais de cada operador deverão ser fornecidas.

Definição: Kr_{oper_i} - Chave privada de cada operador i armazenada no seu DPCPC.

Definição: Ku_{oper_i} - Chave pública de cada operador i relacionada com a sua chave privada.

Definição: $CertKu_{oper_i}$ - Certificado emitido para a chave pública de cada operador i, relacionando-a ao seu detentor.

Definição: SADO - Sistema de Armazenamento de Dados de Operadores

Definição: Ks_{oper} - Chave simétrica de sessão que representa o conjunto de operadores e protege as chaves de aplicação associadas ao conjunto.

Definição: Kr_{aplic} - Chave assimétrica de aplicação que é protegida pelo PSC e pertence ao conjunto de operadores.

Definição: $Ks_{adm-oper}$ - Chave simétrica de sessão compartilhada entre o conjunto de administradores e um conjunto de operadores para fins administrativos.

Definição: Ks_{oper*} - Chave simétrica para ativação dos processos administrativos sob um conjunto de operadores armazenada num sistema de armazenamento de dados não exportáveis.

Definição: SADNE - Sistema de Armazenamento de Dados Não Exportável e não copiável.

O processo de criação de operadores de um PSC deve seguir os seguintes passos:

1. Cada operador i, utilizando um DPCPC próprio, deve gerar neste DPCPC um par de chaves criptográficas: uma chave privada e uma pública. A chave privada Kr_{oper_i} deve permanecer armazenada sem possibilidade de cópia, em um DPCPC;

2. A chave pública Ku_{oper_i} deve ser disponibilizada ao PSC. Caso o operador i já possua um certificado emitido por uma AC confiável ao PSC, ele deve informar o seu certificado;

3. Caso os operadores não informem um certificado emitido por uma AC confiável ao PSC utilizando os dados informados pelos operadores através da IEDCD, o provedor emite um certificado para Ku_{oper_i};

4. Os $CertKu_{oper_i}$ e o CDAA devem ficar em um subsistema interno do provedor chamado de SADO para fins posteriores de autenticação através de desafio;

5. Após a definição dos certificados dos operadores e do PSC, conforme ilustra a Figura 6.2, deveremos gerar uma chave de sessão Ks_{oper}, a qual será utilizada internamente para representar todo o conjunto de operadores e para cifrar as chaves de aplicação pertencentes ao conjunto de operadores;

6. A chave Ks_{oper} será protegida por mecanismos de CS [31], onde serão sempre definidos os k operadores, dos quais no mínimo l farão a recomposição deste segredo;

7. As partes de P_1 até P_k serão cifradas de forma independente entre os operadores através das chaves públicas contidas nos seus certificados;

8. Uma vez cifradas as partes de cada operador as mesmas serão armazenadas no SADO, e serão apagadas junto com Ks_{oper}. Seu armazenamento se dará unicamente na forma cifrada;

9. Devemos gerar uma chave de sessão Ks_{oper*}, a qual é armazenada num SADNE e tem por objetivo garantir a rastreabilidade da chave Ks_{oper}. Ela não é exportável, mas pode ser facilmente reconstruída através de Ks_{oper} e $Ks_{adm-oper}$;

10. Para prover as funcionalidades de reconstrução dos componentes, fazemos um ou-exclusivo bit a bit , $\oplus$, entre as chaves simétricas, gerando assim uma nova componente, denominada $Ks_{adm-oper}$, a qual nada mais e que $Ks_{oper} \oplus Ks_{oper*}$, e será armazenada no SADA;

11. $Ks_{adm-oper}$ será cifrada com a chave pública Ku_{PROV}, extraída do CDAA, e será armazenada no SADA. Sempre após seu uso a mesma deve ser prontamente apagada;

Caso não um operador não possua um certificado emitido externamente ao PSC por uma AC confiável, seu certificado será gerado internamente ao PSC, sendo necessário então fornecer as informações que constarão no certificado do operador através da IEDCD. Vale lembrar que ambos os tipos de certificados, sejam emitidos no PSC ou externamente, devem conter as corretas extensões para a validação dos operadores de PSC. Estas extensões serão declaradas críticas ao certificado e serão definidas em tempo de implementação usando como base os Identificadores de Objetos(OID) da instituição que estiver implementando este mecanismo.

Para o cumprimento das tarefas administrativas, os administradores devem ser capazes de gerar pares de chaves criptográficas assimétricas de aplicação e delegá-las aos operadores. No cumprimento desta tarefa é necessário que, após a geração, os administradores possam cifrar a chave Kr_{aplic} com a chave simétrica Ks_{oper} que representa o conjunto de operadores para o qual a chave será delegada.

A chave Ks_{oper} representa o conjunto de operadores, portanto, não deve ser compartilhada diretamente com os administradores. Por tal motivo, optou-se pela criação de uma chave compartilhada entre o grupo de administradores e cada grupo de operadores, denominada $Ks_{adm-oper}$.O objetivo é prover os administradores com as informações administrativas necessárias e não abrir Ks_{oper}.

Para acessar Ks_{oper}, teremos que recompor as P_k partes decifradas pelos operadores, ou através da liberação de $Ks_{adm-oper}$ pelos administradores, usando as propriedades do ou-exclusivo para descontar de $Ks_{adm-oper}$, o valor de Ks_{oper*}, obtendo assim Ks_{oper}. No caso de ser recuperada uma cópia de segurança do PSC num novo PSC, os dados constates no SADNE não existirão prontamente. Para recria-los, é necessários que os administradores liberem $Ks_{adm-oper}$, e os operadores liberem sua respectiva Ks_{oper}, e em uma operação de ou-exclusivo seja recomposta a chave Ks_{oper*}, habilitando assim os administradores em totalidade de suas funções administrativas sobre o grupo de operadores.

Com um ou mais conjuntos de operadores criados, poderemos dar continuidade às tarefas administrativas, sendo agora necessário que sejam geradas as chaves e delegas para um conjunto de operadores, conforme será descrito na seção 6.4.

6.4 Geração de Pares de Chaves Assimétricas de Aplicação

A geração de pares de chaves assimétricas é uma tarefa administrativa, necessitando da interação de um sub-conjunto do conjunto de administradores, de acordo com a figura 6.3. As chaves aqui geradas são o foco de proteção do PSC, e normalmente serão utilizadas pelas aplicações externas que querem obter as vantagens da proteção física e controle de acesso propiciadas pelo PSC.

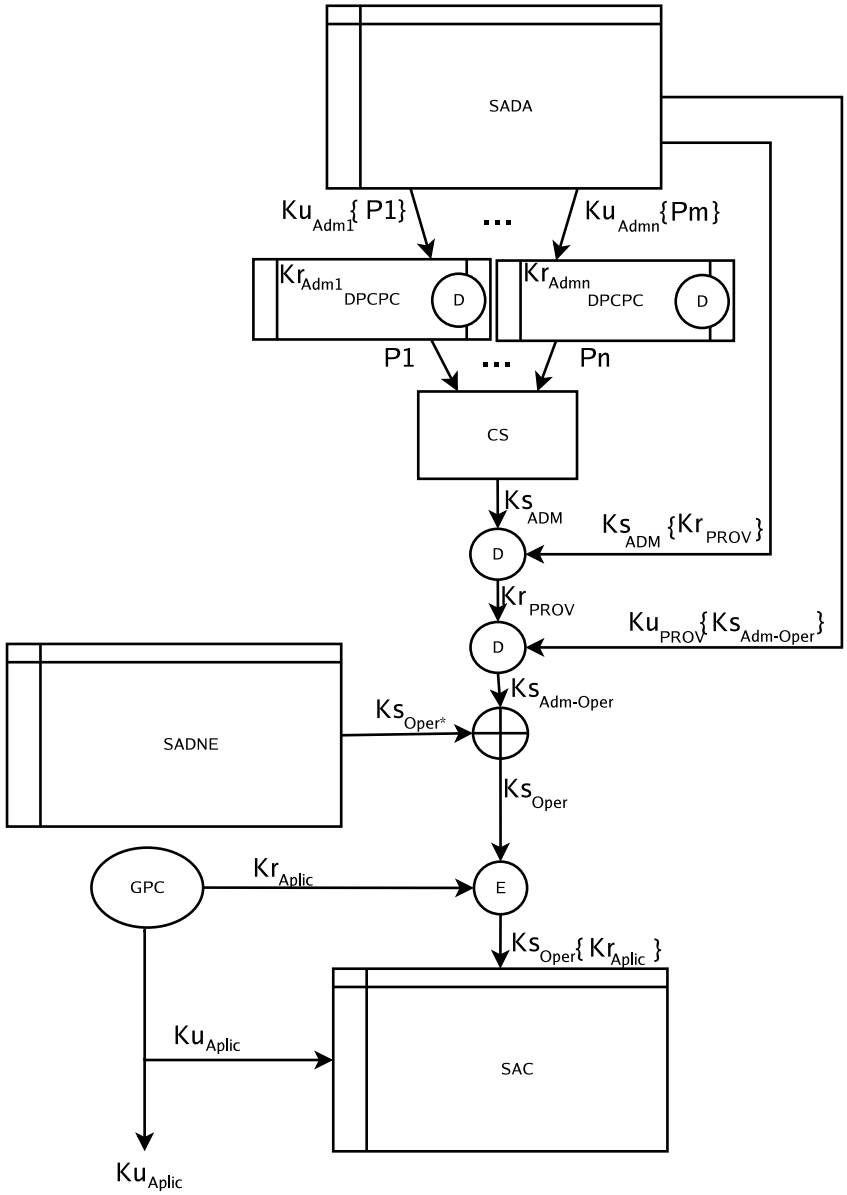


Figura 6.3: Mecanismo para Geração de Par de Chaves Assimétricas.

Para a geração de pares de chaves assimétricas de aplicação é necessário

fornecer:

- Os parâmetros de criação da chave, tais como o algoritmos no qual a chave será usada, o tamanho, e a necessidade de verificações especiais durante a sua geração;
- A política de uso da chave, a qual determina como a chave ficará aberta para seu uso, podendo esta política ser de uso por tempo, ou por número de iterações;
- Serão necessários i DPCPCs de Administradores, onde o valor de i deve ser maior ou igual o mínimo necessário para se efetuar qualquer atividade administrativa dentro do PSC.

Definição: Ku_{aplic} - Chave pública de aplicação relacionada com a chave privada Kr_{aplic}.

Definição: Kr_{aplic} - Chave privada de aplicação relacionada com a chave pública Ku_{aplic}.

Definição: SAC - Sistema de Armazenamento de Chaves.

O processo de geração de chaves de assimétricas de aplicação deve seguir os seguintes passos:

1. Os administradores devem definir os parâmetros de criação da chaves, os quais incluem o seu tamanho e as políticas de uso da mesmas;
2. Uma vez determinados os parâmetros e políticas das chaves, um GPC ligado a um GNA confiável fará a geração de um par de chaves, onde a chave pública Ku_{aplic}, será exportada para o ambiente externo ao provedor e Kr_{aplic}, será cifrada com Ks_{oper} de um conjunto de operadores, o qual será o detentor da chave privada;
3. O conjunto de administradores terá que liberar Ks_{adm}, através da decifragem de cada parte individual $Ku_{adm_i}\{P_i\}$, e a respectiva remontagem do CS;
4. De posse de Ks_{adm}, vamos então recuperar do SADA o valor de $Ks_{adm}\{Kr_{PROV}\}$ e decifrá-lo, obtendo assim Kr_{PROV};
5. Com o objetivo de recompor a chave Ks_{oper}, vamos recuperar o valor de $KU_{PROV}\{Ks_{adm-oper}\}$ e vamos decifra-lo usando a chave Kr_{PROV};

6. Estando habilitados para fazer a delegação em nome de um determinado conjunto de operadores, teremos acesso a Ks_{oper*}, o qual também sofrerá um ou-exclusivo com $Ks_{adm-oper}$, resultando assim no valor de Ks_{oper};

7. De posse de Ks_{oper}, podemos então cifrar a Kr_{aplic} que foi gerada para este grupo de operadores e então guardar $Ks_{oper}\{Kr_{aplic}\}$ no SAC.

As políticas são expressas para determinação de como serão utilizadas as chaves e como elas serão administradas. Isso pode incluir desde mecanismos de liberação por uso ou tempo, até se a chave pode ser de propriedade de mais de um conjunto de operadores.

Com as chaves assimétricas geradas e delegadas aos devidos operadores, o próximo passo é a utilização das chaves criptográficas armazenadas no provedor de serviços criptográficos para fins de exercer sua função numa aplicação externa, o que ocorrerá conforme o descrito na seção 6.5.

6.5 Utilização de Pares de Chaves Assimétricas de Aplicação

A utilização das chaves criptográficas armazenadas num PSC é uma tarefa operacional e não necessita de nenhuma intervenção de administradores para tal.

A Figura 6.4 descreve o processo de liberação de uma chave assimétrica privada de aplicação:

Para a liberação do uso de uma chave privada de aplicação, serão necessários:

- Um mínimo de k operadores, os quais farão a recomposição de Ks_{oper}.

O processo de liberação de uma chave de assimétrica privada de aplicação deve seguir os seguintes passos:

1. Vamos recuperar do SADO as partes cifradas para cada operador;

2. Cada $Ku_{oper_l}\{P_l\}$ parte será submetida ao operador detentor de Kr_{oper_l}, para o deciframento;

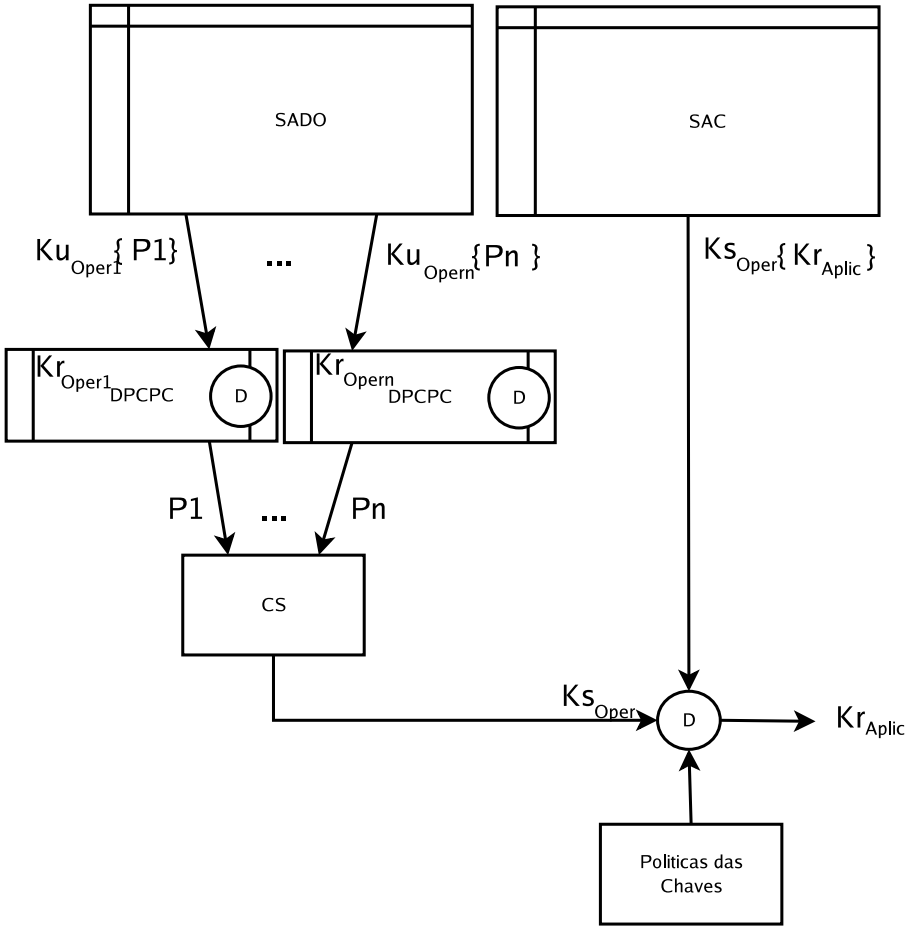


Figura 6.4: Mecanismo para Uso de Chave Privada de Aplicação.

3. De posse das P_l partes, será remontado Ks_{oper} através de CS;

4. Deverá então ser recuperado do SAC a $Ks_{oper}\{Kr_{aplic}\}$ solicitada para uso e através de um algoritmo de decifragem simétrica, liberamos Kr_{aplic} para uso. A partir deste momento, devemos eliminar da memória a presença de Ks_{oper}.

O uso de Kr_{aplic} se dará de acordo com as políticas estabelecidas pelos administradores no momento de sua criação e poderão determinar que a mesma seja liberada para n operações criptográficas, ou por um período de tempo que pode variar de 1 segundo a infinito. Após a expiração da política de uso, a chave deverá ser eliminada da memória na sua forma decifrada, sendo necessário que seja refeito todo o procedimento para um novo uso. Pro se tratar de um PCS, esta chave na formas decifrada sob qualquer ameaça detectada pelo PSC deve ser devidamente removida da memória, mesmo que sua política não tenha expirado ainda.

6.6 Criação do Conjunto de Auditores

Os auditores são responsáveis pela monitoração das operações de todos os indivíduos que operam com o PSC, sendo garantido a eles o acesso aos registros gerados em virtude do uso. A existência de auditores é obrigatória para o funcionamento do PSC, sendo a sua criação necessária após a criação do conjunto de administradores.

A criação do conjunto de auditores é um caso específico da criação do conjunto de operadores. Para a criação do conjunto de auditores é necessário fornecer:

- O número k de auditores do conjunto, ou seja, o tamanho do conjunto;
- O número l de auditores, sendo $1 \leq l \leq k$, mínimo necessário para realizar as atividades relativas ao conjunto.;
- Para os k DPCPC que deverão ser apresentados ao PSC durante a criação do conjunto de auditores, os quais ainda não possuírem certificados confiáveis ao PSC, as informações necessárias à criação dos certificados digitais de cada auditor deverão ser fornecidas.

Definição: Kr_{audit_i} - Chave privada de cada auditor i armazenada no seu DPCPC.

Definição: Ku_{audit_i} - Chave pública de cada auditor i relacionada com a sua chave privada.

Definição: Ks_{audit} - Chave simétrica de sessão que identifica o conjunto de auditores e protege a chave de registro e auditoria.

Definição: $Ks_{adm-audit}$ - Chave simétrica de sessão compartilhada entre o conjunto de administradores e um conjunto de auditores para fins administrativos.

Definição: Ks_{audit*} - Chave simétrica para ativação dos processos administrativos sob um conjunto de auditores.

Definição: Kr_{AUDIT} - Chave privada de recuperação de auditoria no PSC.

Definição: Ku_{AUDIT} - Chave pública para conferência dos registros de auditoria.

Definição: SRDA - Sistema de Registro de Dados de Auditoria.

O processo de criação de um conjunto de auditores para o PSC deve seguir os seguintes passos:

1. Cada auditor i, utilizando um DPCPC próprio, deve gerar um par de chaves criptográficas: uma chave privada e uma pública. A chave privada Kr_{audit_i} deve permanecer armazenada sem possibilidade de cópia, em um DPCPC;

2. A chave pública Ku_{audit_i} deve ser disponibilizada ao PSC. Caso o auditor i já possua um certificado emitido por uma AC confiável ao PSC, ele deve informar o seu certificado;

3. Caso os auditores não informem um certificado emitido por uma AC confiável ao PSC, seu certificado será gerado internamente ao PSC, sendo necessário então fornecer as informações que constarão no certificado do auditor através da IEDCD;

4. Os $CertKU_{audit_i}$ e o CDAA devem ficar em um subsistema interno do provedor chamado de SADO, para fins posteriores de autenticação através de desafio. Até aqui o processo é igual ao de criação do conjunto de operadores, e para tal não necessita de um subsistema próprio de armazenamento.

5. Será então gerada uma chave Ks_{audit} e dividida por CS para os k auditores;

6. Após a divisão de Ks_{audit} em P_k partes, ciframos cada parte com as Ku_{audit_i} chaves públicas extraídas dos certificados de cada auditor e armazenamos cada parte $Ku_{audit_i}\{P_i\}$ no SADO;

7. Devemos fazer as operação complementares que vão gerar $Ku_{PROV}\{Ks_{adm-audit}\}$ e Ks_{audit*}. Armazenamos no sistema de armazenamento de dados de administradores o valor de $Ku_{PROV}\{Ks_{adm-audit}\}$, descartando o valor de Ks_{audit*};

8. $Ks_{adm-audit}$ será cifrada com a chave pública Ku_{PROV} extraida do CDAA e será armazenada no SADA. Sempre após seu uso a mesma deve ser prontamente apagada;

9. Deve-se gerar de um par de chaves assimétricas de auditoria, Ku_{AUDIT} e Kr_{AUDIT};

10. A chave Kr_{AUDIT} é cifrada usando um algoritmo simétrico com a chave Ks_{audit} e armazenando-a no SAC;

11. Para Ku_{AUDIT}, vamos emitir um certificado digital X509v3 com extensões que identifiquem o seu uso para fins de sigilo e integridade de registros de provedores de serviços criptográficos, assinando este certificado com Kr_{PROV};

12. Todos os registros de atividades exercidas no PSC devem ser armazenados num SRDA;

13. Este sistema de dados deve possuir um tamanho pré-determinado e será acessado unicamente pelos auditores, os quais podem a qualquer momento retirar os registros e limpar o SRDA;

14. No processo de retirada todos os registros retirados do PSC deve ser assinados usando Kr_{AUDIT}, sendo necessário para tal a liberação da chave através da presença do numero mínimo de auditores do conjunto.

Caso os auditores não informem um certificado emitido por uma AC confiável ao PSC utilizando os dados informados pelos auditor através da IEDCD, o provedor emite um certificado para Ku_{audit_i}. Vale lembrar que ambos os tipos de certificados, sejam emitidos no PSC ou externamente, devem conter as corretas extensões para a validação dos auditor de PSC.

O descarte de Ks_{audit*} é necessário por que em momento algum queremos que os administradores possam subverter o sistema e trocar o conjunto de auditores sem a expressa autorização dos mesmos. No caso do comprometimento de um conjunto de auditores podemos a qualquer momento gerar um novo conjunto de auditores, que atuará de forma concorrente com os conjuntos que já existirem.

A chave Ks_{audit} representa o conjunto de auditores, portanto, não deve ser compartilhada com os administradores. Por tal motivo optou-se pela criação de uma chave compartilhada entre o conjunto de administradores e cada conjunto de auditores, denominada $Ks_{adm-audit}$.

A chave assimétrica chamada Ks_{audit*}, descartada ao seu primeiro uso, tem por objetivo garantir que os administradores somente operarão sobre a chave de registros com autorização dos auditores. Ela é apagada, mas pode ser facilmente reconstruída através de Ks_{audit} e $Ks_{adm-audit}$.

Para a troca do conjunto de auditores, deve-se proceder de forma análoga à da troca de operadores, mas primeiro devemos reconstruir o valor de Ks_{audit*},

autorizando assim os administradores para tal. Na delegação do novo conjunto de auditores, descartamos sempre o valor de Ks_{audit*} para o novo conjunto que for delegado. O mesmo ocorre no caso da criação de um segundo conjunto de auditores.

O preenchimento total do SRDA implica em inoperância do provedor para quaisquer atividades até que os auditores retirem os registros do PSC.

6.7 Troca de Administradores para um Provedor de Serviços Criptográficos

A troca de administradores de um PSC é uma tarefa realizada por n administradores. Esta tarefa deve ocorrer sempre que exista o comprometimento de um sub-conjunto do conjunto de administradores,cujo número não seja superior a $m-n$. Um comprometimento maior que o estipulado significa um comprometimento definitivo dos dados do PSC

A figura 6.5 mostra como é realizada a operação de troca de administradores.

Para efetuarmos a troca, devemos seguis os seguintes passos:

1. Devemos contar com n administradores, os quais individualmente decifrarão $Ku_{adm_i}\{P_i\}$ para a obtenção de P_n partes;

2. As partes serão remontadas em um mecanismo CS para a obtenção de Ks_{adm};

3. Devemos então recuperar a partir do SADA o valor de $Ks_{ADM}\{Kr_{PROV}\}$, e a ele aplicamos um algoritmo simétrico de decifragem usando Ks_{ADM} como chave. Este processo nos dará acesso a Kr_{PROV};

4. Geramos uma nova chave simétrica Ks_{ADM*} usada para cifrar novamente Kr_{PROV};

5. Após o ciframento, armazenamos $Ks_{ADM*}\{Kr_{PROV}\}$ no lugar do $Ks_{ADM}\{Kr_{PROV}\}$ no SADA, garantindo assim que não haverá mais acesso à antiga chave;

6. Devemos criar um novo conjunto de administradores, passando novamente os parâmetros m e n para a criação do mecanismo de CS;

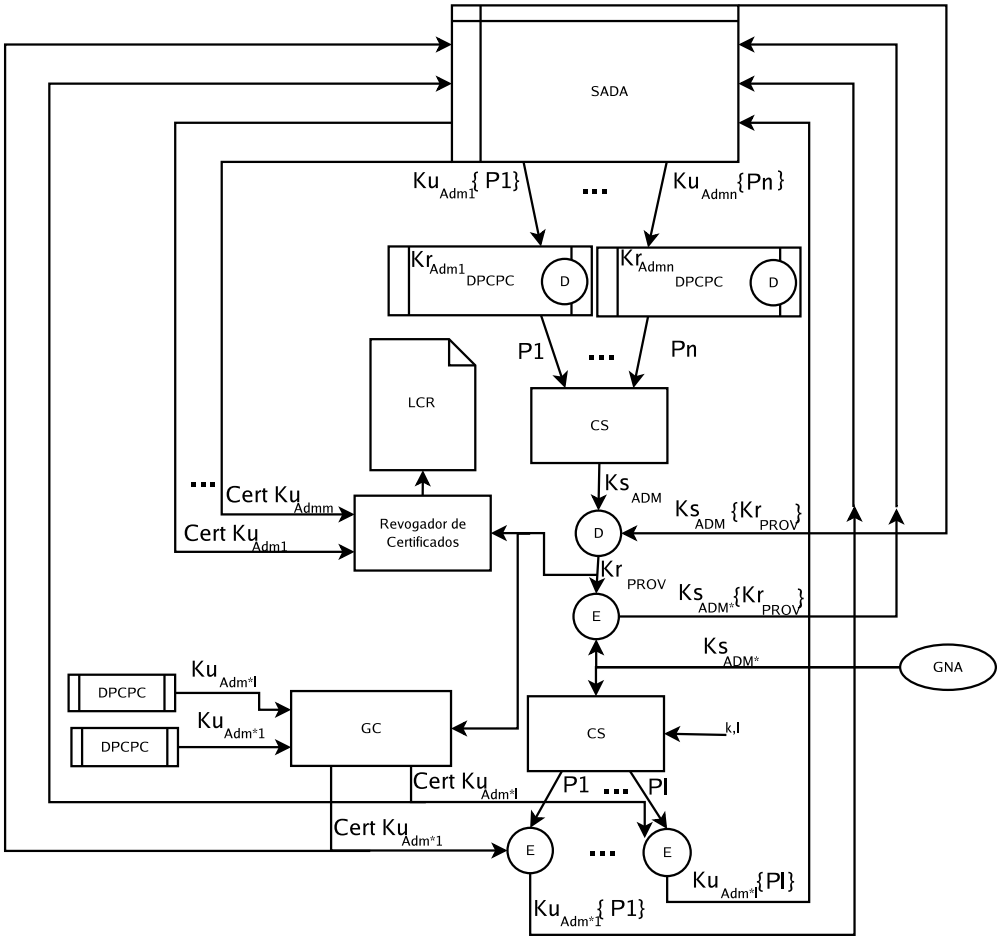


Figura 6.5: Troca do Conjunto de Administradores.

7. Cada novo administrador i, utilizando um DPCPC próprio, deve gerar um par de chaves criptográficas: uma chave privada e uma pública. A chave privada Kr_{adm_i} deve permanecer armazenada sem possibilidade de cópia, em um DPCPC;

8. A chave pública Ku_{adm_i} deve ser disponibilizada ao PSC;

9. Caso o novo administrador i já possua um certificado emitido por uma AC confiável ao PSC, ele deve informar o seu certificado;

10. Caso o novo administrador não o possua, seu certificado será gerado internamente ao PSC, sendo necessário então fornecer as informações que constarão no certificado do administrador através da IEDCD;

11. Com os administradores novamente criados, quebramos a chave Ks_{ADM*} em m pedaços e ciframos cada pedaço usando a chave pública de cada novo administrador;

12. Eliminamos os antigos $Ku_{adm_i}\{Pi\}$ do sistema de armazenamento de dados de administradores e colocamos os recém criados em seus lugares, garantindo assim a impossibilidade de recuperação do antigo Ks_{ADM} e a efetiva recuperação do novo Ks_{ADM*};

13. Com Kr_{PROV}, revogamos todos os m certificados de administradores constantes no SADA, e a partir da revogação emitimos um LCR, a qual será devidamente armazenada no provedor e checada a cada nova intervenção de administradores no sistema.

O certificado de cada administrador dever ser validado neste processo, assim como acontece em todas as atividades administrativas executadas pelo conjunto de administradores. Vale lembrar que ambos os tipos de certificados, sejam emitidos no PSC ou externamente, devem conter as corretas extensões para a validação dos administradores de PSC.

A recuperação de administradores de um PSC é uma tarefa importante e arriscada, pois quaisquer problemas podem inviabilizar a gerência de todas as chaves protegidas pelo provedor. Claro que, independentemente, devemos sempre registrar a ocorrência deste procedimento no sistema de auditoria e registro do PSC.

6.8 Troca de Operadores para uma Chave Assimétrica

Os mecanismos de operação do PSC podem exigir uma tarefa administrativa que consiste na troca do conjunto de operadores responsável por um par de chaves assimétricas de aplicação.

A necessidade desta operação pode ser devido a inúmeros fatores, dentre eles, o comprometimento de algum operador seja, por fraude ou inutilização do seu DPCPC, ou pela simples expiração da posse da chave a um grupo.

A figura 6.6 representa o processo de troca de operadores para uma chave criptográfica.

Para efetuarmos a troca, devemos seguir os seguintes passos:

1. Devemos contar com n administradores, os quais individualmente decifrarão $Ku_{adm_i}\{P_i\}$ para a obtenção de P_n partes;

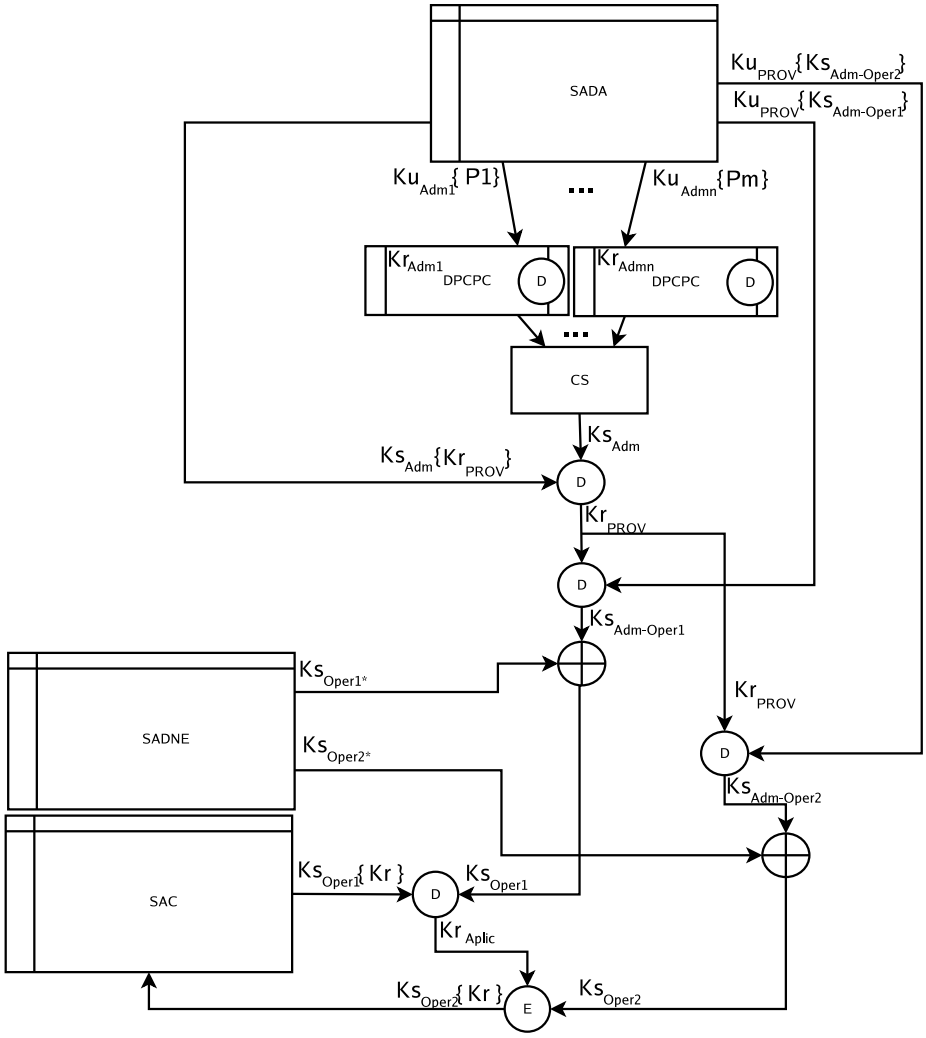


Figura 6.6: Troca do Conjunto de Operadores para um Chave Assimétrica.

2. As partes serão remontadas em um mecanismo CS para a obtenção de Ks_{adm};

3. Buscamos no SADA o valor de $Ks_{adm}\{Kr_{PROV}\}$. Aplicamos um algoritmo simétrico de deciframento em $Ks_{adm}\{Kr_{PROV}\}$ com a chave Ks_{adm} e obtemos Kr_{PROV};

4. Devemos buscar também no SADA os valores de $Ku_{PROV}\{Ks_{adm-oper1}\}$. Em $Ku_{PROV}\{Ks_{adm-oper1}\}$ aplicamos um algoritmo assimétrico de deciframento e obtemos $Ks_{adm-oper1}$;

5. Com $Ks_{adm-oper1}$, e estando em um provedor de conhecimento do conjunto de operadores, temos acesso a Ks_{oper1*},sobre o qual fazemos um ou-exclusivo com $Ks_{adm-oper1}$ para obtermos Ks_{oper1}, o qual pertence ao primeiro conjunto de operadores;

6. Devemos buscar também no SADA os valores de $Ku_{PROV}\{Ks_{adm-oper2}\}$. Em $Ku_{PROV}\{Ks_{adm-oper2}\}$ aplicamos um algoritmo assimétrico de deciframento e obtemos $Ks_{adm-oper2}$;

7. Com $Ks_{adm-oper2}$, e estando em um provedor de conhecimento do conjunto de operadores, temos acesso a Ks_{oper2*},sobre o qual fazemos um ou-exclusivo com $Ks_{adm-oper2}$ para obtermos Ks_{oper2}, o qual pertence ao segundo conjunto de operadores;

8. De posse de Ks_{oper_1} e Ks_{oper_2}, recuperamos do SAC o valor de $Ks_{oper_1}\{Kr\}$, ao qual aplicamos um algoritmo de deciframento assimétrico e obtemos Kr_{aplic};

9. Com Kr_{aplic} disponível, podemos agora cifrá-lo para o novo conjunto de operadores, simplesmente aplicando em Kr_{aplic} um algoritmo simétrico de cifragem com a chave sendo Ks_{oper_2}, obtendo assim $Ks_{oper_2}\{Kr_{aplic}\}$;

10. Devemos eliminar do SAC o valor de $Ks_{oper_1}\{Kr_{aplic}\}$ e colocarmos o valor de $Ks_{oper_2}\{Kr_{aplic}\}$, para garantir que o antigo conjunto de operadores não mais possua acesso as chaves.

Este procedimento nos habilita a trocar de propriedade de uma chave protegida pelo provedor para um novo conjunto de operadores, mesmo sem o consentimento do atual conjunto. Isto é necessário, pois mesmo havendo um comprometimento total do conjunto de operadores, a chave protegida pode ser recuperada para um novo conjunto.

Um fator importante deste processo é que o mesmo só é possível de ser realizado em um provedor no qual exista a chave não exportável $Ksoper*$ correspondente ao conjunto de operadores inicial, garantindo assim a rastreabilidade por parte dos operadores do ocorrido com sua chave, visto que todos os procedimentos administrativos e operacionais do provedor são registrados pelo sistema de auditoria.

6.9 Sistema para Criação de Cópias de Segurança das Chaves

A criação de cópias de segurança das chaves do provedor de serviços criptográficos é uma tarefa administrativa, a qual deverá ser realizada com uma periodicidade estabelecida pelas normas segundo as quais o provedor estará operando. As copias de segurança consistem em manter em lugar seguro e de forma segura os dados do PSC para que, no caso de um desastre, possa ser dada continuidade às suas atividades.

O sistema para criação das cópias de segurança das chaves do PSC, consistem e 2 atividades distintas. Uma executada cada vez que se quer incluir um novo PSC para a recuperação de cópias de segurança, e outra para a geração das copias de segurança efetivamente.

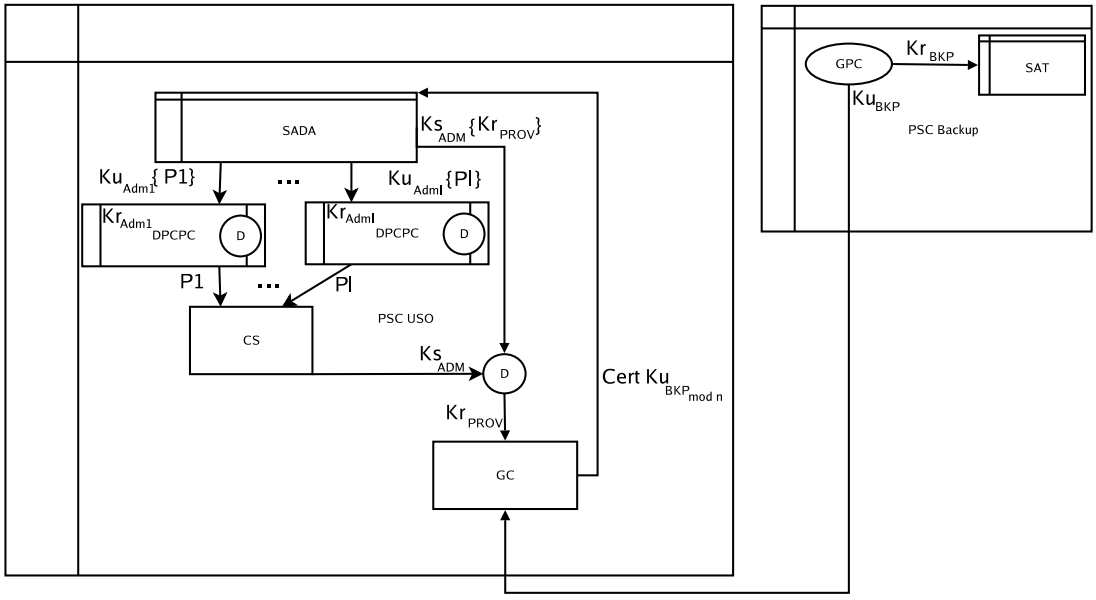


Figura 6.7: Geração e Troca de Chaves Assimétricas para Cópias de Segurança .

Conforme mostra a Figura 6.7, para fazermos cópias de segurança de um PSC, precisaremos:

- De um PSC do qual queremos extrair cópias de segurança, o qual chamaremos de PSC em uso;
- De um ou mais PSCs para os quais queremos importar os arquivos com as cópias de segurança, os quais chamaremos de PSC backup.

- De um subconjunto de tamanho l do conjunto de Administradores, onde l deve respeitar a condição $1 \leq n \leq l \leq m$.

Definição: Kr_{Bkp} - Chave assimétrica privada para o transporte seguro de cópias de segurança.

Definição: Ku_{Bkp} - Chave assimétrica pública para o transporte seguro de cópias de segurança.

Definição: SAT - Sistema de Armazenamento de Dados Temporário

Definição: Ks_{Bkp} - Chave simétrica de sessão gerada aleatoriamente para cifrar os dados exportados pelo sistema de cópias de segurança.

Para a inclusão de um novo PSC para a recuperação de cópias de segurança devemos:

1. Inicializar o PSC backup no modo de geração de chaves de backup. Isso consistem em gerar um par de chaves, Ku_{BKP} e Kr_{BKP}. A chave privada Kr_{BKP} deve ficar armazenada num subsistema SAT sem a possibilidade de exportação;

2. A chave pública Ku_{BKP} deve ser exportada para o exterior do PSC;

3. Os administradores do PSC em uso devem pegar a chave Ku_{BKP} e importá-la através do GC. A importação através do GC implica na emissão de um certificado de chave publica de backup;

4. Devemos contar com l administradores, os quais individualmente decifrarão $Ku_{adm_i}\{P_i\}$ para a obtenção de P_l partes;

5. As partes serão remontadas em um mecanismo CS para a obtenção de Ks_{adm};

6. Buscamos no SADA o valor de $Ks_{adm}\{Kr_{PROV}\}$. Aplicamos um algoritmo simétrico de deciframento em $Ks_{adm}\{Kr_{PROV}\}$ com a chave Ks_{adm} e obtemos Kr_{PROV};

7. De posse de Kr_{PROV}, emitimos um certificado digital de chave pública de backup para a chave importada do PSC backup. Este certificado deve conter as extensões que identifiquem o seu uso como uma chave pública de transporte de cópias de segurança;

8. Caso a entrada de Ku_{BKP}, seja dada já na forma de um certificado digital emitido por uma AC confiável ao PSC, este certificado deve conter as extensões corretas para a sua identificação como um certificado de chave pública de transporte de cópias de segurança.

9. O certificado $CERTKu_{BKP}$, deve ser armazenado no SADA para posterior uso nos processo de criação dos arquivos de cópias de segurança.

Normalmente esta etapa é executada exporádicamente, e pode ser repetida quantas vezes forem necessárias para incluir todos os PSCs backup necessários à política definida para a operação do PSC em uso.

Uma vez existindo pelo menos um certificado $CERTKu_{BKP}$, podemos passar para a atividade de geração das copias de segurança. A figura 6.8 nos mostra o detalhamento do processo de criação de arquivos de cópias de segurança.

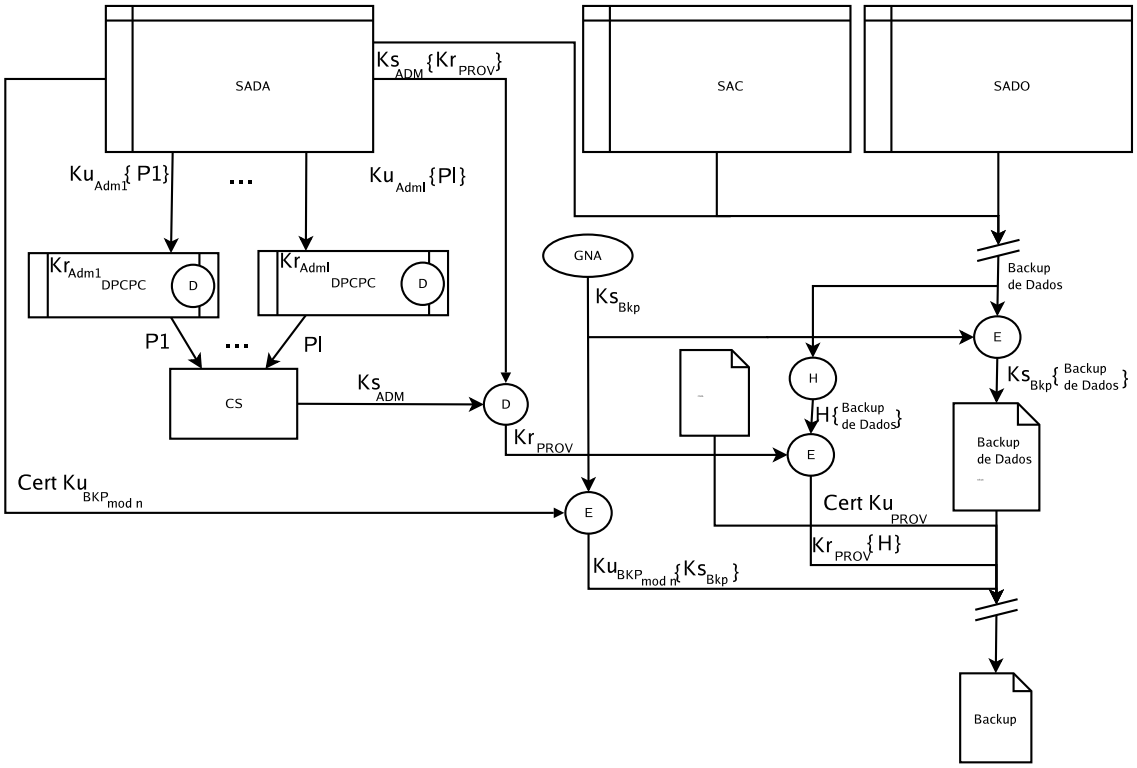


Figura 6.8: Mecanismo de Criação de Cópias de Segurança.

Para a criação de arquivos de cópias de segurança devemos:

1. Contar com l administradores, os quais farão a recomposição de Ks_{ADM}. Através da decifragem individual das l partes cifradas $Ku_{adm_i}\{P_i\}$ e da remontagem dos P_l pedaços por um mecanismo de CS é que obteremos Ks_{ADM};

2. O valor de l deve ser informado, e deve respeitar a condição $1 \leq n \leq l \leq m$;

3. Para fins de cifragem dos dados e das chaves que serão exportadas durante o processos de cópias de segurança, geramos uma nova chave denominada Ks_{Bkp};

4. Devemos então recuperar a partir do SADA, o valor de $Ks_{ADM}\{Kr_{PROV}\}$, e a ele aplicamos um algoritmo simétrico de decifragem usando Ks_{ADM} como chave. Este processo nos dará acesso a Kr_{PROV};

5. Todos os sistemas de armazenamento de dados serão concatenados em um único arquivo, o qual chamaremos de *Backup de Dados*;

6. O *Backup de Dados* por sua vez será cifrado com um algoritmo simétrico usando como chave Ks_{Bkp}, dando origem a $Ks_{Bkp}\{$*Backup de Dados*$\}$;

7. Em *Backup de Dados* também vamos aplicar uma função de resumo criptográfico $H\{\}$ e obteremos $H\{$*Backup de Dados*$\}$, o qual será usado para garantir a integridade dos dados na recuperação das cópias de segurança;

8. Para provermos garantias de autenticidade à operação de cópias de segurança, vamos cifrar $H\{$*Backup de Dados*$\}$ com um algoritmo assimétrico usando como chave Kr_{PROV}, gerando assim uma assinatura digital verificável através do certificado auto-assinado do provedor denominada $Kr_{PROV}\{H\{$*Backup de Dados*$\}\}$;

9. A proteção de KS_{Bkp} para o transporte se dará cifrando-a através de um algoritmo assimétrico, com a chave de entrada cada uma das $Ku_{BKP_{modn}}$ extraídas dos $CertKu_{BKP_{modn}}$, obtendo assim $Ku_{BKP_{modn}}\{KS_{Bkp}\}$;

10. A geração do arquivo final de cópia de segurança se dará concatenando os seguinte componentes obtidos neste processo:

    - $Ks_{Bkp}\{$*Backup de Dados*$\}$
    - $Kr_{PROV}\{H\{$*Backup de Dados*$\}\}$
    - CDAA
    - Os n $Ku_{BKP_{modn}}\{KS_{Bkp}\}$

A exigência de l administradores não é um requisito do mecanismo de CS, mas sim, é forçado internamente ao provedor para propiciar uma maior participação na hora da recuperação das cópias.

Os dados que serão guardados pelo mecanismo de cópias de segurança são os presentes nos seguintes sistemas de armazenamento:

- SADA;
- SADO;
- SAC;

O SADNE não deve entrar na rotina de cópias de segurança, visto que ele é fundamental para garantirmos a rastreabilidade das chaves dos operadores.

Com este processo de geração de cópias de segurança, será possível transportar todo o ambiente operacional já em uso de um provedor para outro de forma segura, garantindo a autenticidade, integridade e segurança de recuperação.

6.10 Recuperação de Cópias de Segurança

O mecanismo de recuperação de cópias de segurança tem a função de recuperar de forma segura do ambiente operacional de um PSC já existente em outro PSC, garantindo os requisitos de segurança neste processo.

Conforme mostra a Figura 6.9, para recuperarmos cópias de segurança de um PSC, precisaremos:

- De um PSC no qual queremos recuperar cópias de segurança, o qual chamaremos de PSC backup, e que já foi devidamente inicializado para tal função;
- Do arquivo de cópia de segurança gerado em um PSC o qual tinha na sua lista de PSCs backup o PSC no qual queremos recuperar tais cópias;
- De um subconjunto de tamanho l do conjunto de Administradores do PSC de origem, onde l deve respeitar a condição $1 \leq n \leq l \leq m$.

Para a recuperação das cópias de segurança do ambiente de um PSC devemos seguir os seguintes passos:

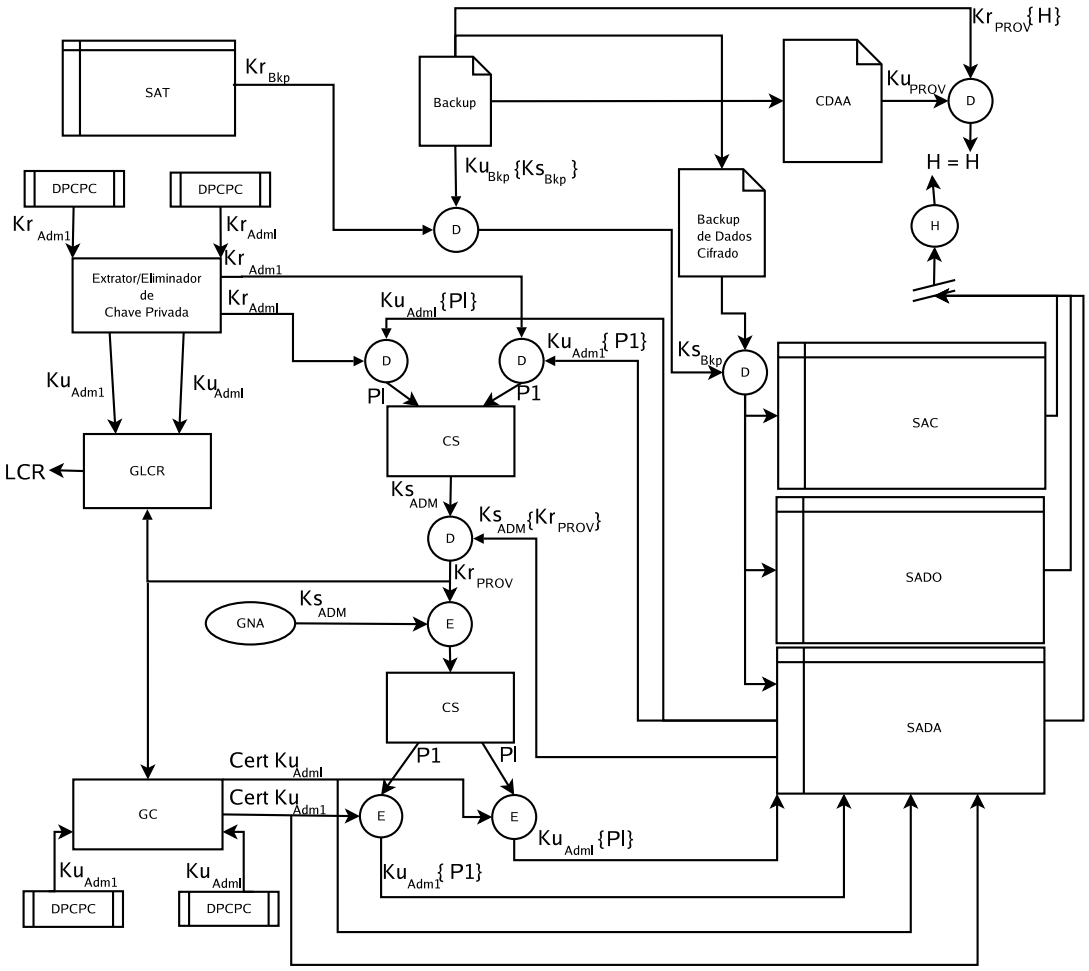


Figura 6.9: Mecanismo de Recuperação de Cópias de Segurança.

1. De posse do arquivo de saída do processo de criação de cópias de segurança, carregá-lo no novo PSC;

2. Após carregado, vamos separar as inúmeras partes que o compõem e dar inicio ao processo de recuperação.

3. Recuperamos Kr_{Bkp} do SAT, e com ela aplicamos um algoritmo assimétrico de deciframento no componente $Ku_{Bkp}\{Ks_{Bkp}\}$ respectivo ao PSC sendo utilizado para a recuperação de cópias de segurança no momento, obtendo assim Ks_{Bkp};

4. Utilizando Ks_{Bkp}, podemos extrair do arquivo de cópia de segurança o valor de $Ks_{Bkp}\{$*Backup de Dados*$\}$, ao qual aplicamos um algoritmo simétrico de decifragem com a chave Ks_{Bkp}, e obtemos o *Backup de Dados*;

5. Com *Backup de Dados*, separamos os arquivos anteriormente concatenados nos

subsistemas de armazenamento do provedor, tendo assim acesso aos dados necessários para a continuidade do processo de recuperação;

6. A garantia de autenticidade e integridade dos dados recuperados de $Ks_{Bkp}\{$*Backup de Dados*$\}$ pode ser dada facilmente calculando um resumo criptográfico do arquivo, chamado $H_{novo}\{$*Backup de Dados*$\}$, e comparando ele com a saída da decifragem assimétrica de $Kr_{PROV}\{H\{$*Backup de Dados*$\}\}$ decifrado com o CDAA, ambos extraídos do arquivo de cópias de segurança e obtendo $H_{antigo}\{$*Backup de Dados*$\}$;

7. Para cada $Ku_{adm_i}\{Pi\}$,devemos decifrá-la com a correspondente Kr_{adm_i}. Uma vez decifrada a referida parte devemos inutilizar Kr_{adm_i} da DPCPC do administrador, inviabilizando assim que ele possa utilizá-la novamente no PSC antigo. Isto pode ser feito através da troca da senha de acesso ao dispositivo de armazenamento, ou através de comandos específicos do dispositivo para apagamento da chave privada nele contida;

8. As partes serão remontadas em um mecanismo CS para a obtenção de Ks_{adm};

9. Devemos então recuperar a partir do SADA o valor de $Ks_{ADM}\{Kr_{PROV}\}$, e a ele aplicamos um algoritmo simétrico de decifragem usando Ks_{ADM} como chave. Este processo nos dará acesso a Kr_{PROV};

10. Devemos informar novos limiares m e n para o mecanismo de segredo compartilhado;

11. Devemos gerar um novo valor para Ks_{adm}, o qual passará a representar o novo conjunto de administradores e através de um algoritmo simétrico de ciframento irá proteger Kr_{PROV}, gerando assim um novo $Ks_{adm}\{Kr_{PROV}\}$;

12. Cada novo administrador i, utilizando um DPCPC próprio, deve gerar um par de chaves criptográficas: uma chave privada e uma pública. A chave privada Kr_{adm_i} deve permanecer armazenada sem possibilidade de cópia, em um DPCPC.;

13. A chave pública Ku_{adm_i} deve ser disponibilizada ao PSC.;

14. Caso o administrador i já possua um certificado emitido por uma AC confiável ao PSC, ele deve informar o seu certificado;

15. Caso não possua um certificado válido, um novo certificado será gerado internamente ao PSC, sendo necessário então fornecer as informações que constarão no certificado do administrador através da Interface de Entrada de Dados para Certificado Digital - IEDCD;

16. Os $CertKU_{adm_i}$ e o CDAA devem ser armazenados no SADA, para fins posteriores de autenticação através de desafio;

17. $Ks_{ADM}\{Kr_{PROV}\}$ deve ser armazenada no SADA visando a recuperação de Kr_{PROV};

18. A chave Ks_{ADM} será protegida por mecanismo de CS [31], onde serão sempre definidos os m administradores, dos quais no mínimo n farão a recomposição deste segredo sendo respeitada sempre a condição $1 \leq n \leq m$;

19. As partes de P_1 até P_m, serão cifradas de forma independente entre os novos administradores através de suas chaves públicas contidas nos seus certificados;

20. Uma vez cifradas as partes, as mesmas serão armazenadas no SADA, e serão apagadas junto com Ks_{ADM}. Seu armazenamento se dará unicamente na forma cifrada;

21. Com o SADA recuperado e verificado quanto a autenticidade e integridade, podemos extrair dele os m certificados dos administradores. Estes certificados devem ser submetidos a um Gerador de Listas de Certificados Revogados - GLCR, garantido assim a sua revogação;

22. Para fins de proteção de Kr_{PROV}, vamos extrair do SADA os valores das m partes cifradas com os certificados já revogados , assim como o antigo valor de $Ks_{adm}\{Kr_{PROV}\}$;

A importância da destruição das l Kr_{adm_i} se deve ao fato de que, pelos requisitos de segurança de um PSC ,não podemos permitir que existam instâncias paralelas de chaves criptográficas operando.

A operação de instâncias paralelas do provedor torna inviável a garantia dos processos de auditoria de uso e a gerência de uma chave criptográfica, tornando possível fraudes indetectáveis. A necessidade do apagamento ou destruição das chaves privadas dos administradores é requisito porque não existe meio de comunicação entre a

nova instância e a antiga, sendo assim uma forma de garantir a não operação da antiga instância.

Deve-se levar em consideração que a cópia de segurança deve existir unicamente para a recuperação de um desastre ocorrido com o provedor que possuía as chaves. A não existência dos administradores deste provedor com problemas não afeta em nada a operação das chaves no novo provedor.

A comparação positiva entre $H_{novo}\{$*Backup de Dados*$\}$ e $H_{antigo}\{$*Backup de Dados*$\}$ indica que os fatores de integridade e autenticidade da recuperação da cópia de segurança foram respeitadas.

A operacionalização deste novo provedor agora depende da geração de um novo conjunto de administradores. Este processo de recuperação das cópias de segurança nos prove as garantias de integridade, autenticidade e da não operacionalização de instâncias paralelas, mas deixou de tratar o problema de rastreabilidade das chaves assimétricas protegidas pelo provedor.

A garantia de rastreabilidade das chaves protegidas pelo provedor é dada automaticamente, pelo simples fato de que os valores constantes no sistema de armazenamento de dados não exportáveis não foram trazidos para o novo provedor, garantindo assim que nem mesmo os administradores podem se apropriar de chaves, e muito menos delegar as mesmas para outros conjuntos de operadores.

Para que as chaves voltem a ser gerenciáveis pelo conjunto de administradores, devemos regerar os valores de todos os Ks_{oper*} existentes no SADNE do antigo provedor. Isso é facilmente alcançável usando-se da interação de Administradores e Operadores em conjunto, garantindo assim que os operadores conheçam a existência de seu novo PSC.

No processo de recuperação dos x segredos Ks_{oper*}, vamos recuperar do SADA, para cada conjunto de operadores, os valores de $Ks_{adm-oper}$ e $Ku_{PROV}\{Ks_{adm-oper}\}$. Com a presença de n administradores, podemos decifrar as n partes cifradas $Ku_{admi}\{Pi\}$ partes, e com estas partes remontarmos Ks_{adm}.

Recuperamos também do SADA o valor de $Ks_{adm}\{Kr_{PROV}\}$ e a ele aplicamos um algoritmo simétrico de deciframento com a chave Ks_{adm} e obtemos Kr_{PROV}. De posse de Kr_{PROV} podemos aplicar em $Ku_{PROV}\{Ks_{adm-oper}\}$ um algoritmo de deciframento assimétrico e obter $Ks_{adm-oper}$.

Com a presença de l operadores podemos recuperar as l partes cifradas $Ku_{oper_i}\{Pi\}$ e decifra-las para a obtenção das l partes necessárias para a reconstrução de Ks_{oper}.

De posse de $Ks_{adm-oper}$ e Ks_{oper}, aplicaremos uma operação de ou-exclusivo sobre Ks_{oper} e $Ks_{adm-oper}$, uma com cada conjunto de operadores, obtendo ao final os valores de Ks_{oper*}, os quais serão devidamente armazenados no SADNE do novo provedor.

Após estes procedimentos temos a operacionalização completa de todo o ambiente do antigo PSC no novo PSC, garantindo também os requisitos de rastreabilidade das chaves por ele gerenciadas.

6.11 Importação de Certificados de ICPs confiáveis ao PSC

A construção de provedores de serviços criptográficos para estruturas de ICP implica na colocação de características desejáveis para melhorar o ambiente operacional do mesmo neste contexto. Para tal, criamos um mecanismo de importação de certificados de ICPs confiáveis ao PSC.

A qualquer momento durante a operação do provedor, os administradores podem efetuar uma relação de confiança com uma ICP, garantindo assim o reconhecimento interno ao provedor de certificados digitais emitidos fora dele.

Este reconhecimento se dará pela inclusão de novos caminhos de certificação às listas de certificados confiáveis constantes no PSC. A inserção de novos caminhos de certificação, quando solicitada ,fará com que seja necessário regerar a LCC e para que existam as garantias de integridade da mesma, ela deve ser sempre assinada digitalmente com Kr_{PROV}.

A importação de caminhos de certificação para o provedor acarreta também na importação das LCR de todas as autoridades certificadoras existentes no caminho de certificação. A não existência de uma LCR válida implica na não confiança de quaisquer certificados emitidos pela AC constante no caminho de certificação.

Com estes mecanismos, podemos representar de forma muito mais elaborada e concisa as relações de confiança que existem entre as ACs de uma ICP, garan-

tindo assim a interoperabilidade entre os membros de vários níveis dentro das hierarquias.

6.12 Conclusões

Para o efetivo funcionamento de um PSC em um ambiente de ICP, precisamos de um mecanismo que nos possibilite total controle do ciclo de vida das chaves protegidas.

O objetivo deste capítulo foi a criação e a explanação deste tipo de mecanismo, tentando sempre cumprir os requisitos impostos pelas características de uso. Nos mecanismos propostos foram evidenciados os problemas e as soluções para vários casos de uso de um PSC.

Os principais detalhes apresentados neste capítulo foram os relativos aos mecanismos de ativação de chaves por parte dos operadores e do mecanismos de troca de chaves de administradores no momento da recuperação de backup, pois estes não são tratados por nenhum equipamento de mercado e são problemas eminentes de ICP.

Por fim este capítulo provê todas as necessidades para a implementação do mecanismo, o qual vai ser detalhado no projeto adiante.

Capítulo 7

Implementação do Protótipo

Este capítulo tem por objetivo mostrar o processo de prototipagem do PSC proposto, detalhando o hardware escolhido para base do protótipo, o mecanismo de geração de números aleatórios e aceleração criptográfica, o sistema operacional embarcado, as bibliotecas escolhidas e a implementação da aplicação gestora.

Para a implementação do protótipo foi escolhida uma plataforma baseada em arquitetura Intel, visando a facilidade de integração com o ambiente de desenvolvimento. O ambiente de desenvolvimento foi totalmente baseado em ferramentas de software livre, desde ferramentas de projeto e modelagem, até as bibliotecas, compilador e ambiente de desenvolvimento.

Neste capítulo, veremos na seção 7.1, uma explanação das características da plataforma de hardware escolhida para o processo de prototipagem, assim como na seção 7.2 veremos a placa aceleradora e de geração de números aleatórios escolhida. Na seção 7.3 veremos o sistema operacional OpenBSD, o qual foi escolhido para executar em conjunto com o hardware e prover as necessidades de nossa aplicação de gerenciamento de chaves.

A seção 7.4 trata das bibliotecas de software livre escolhidas para ajudarem no processo de prototipagem do projeto, sendo mostradas as principais bibliotecas utilizadas, dentre elas o OpenSSL, OpenCT, OpenSC e share secret.

Finalizando, veremos o detalhamento da aplicação gestora de chaves, a qual é uma implementação do sistema proposto para gestão de chaves do capítulo 6.

7.1 Hardware

Para a implementação do protótipo do PSC proposto neste trabalho, foram avaliados inúmeros equipamentos para servirem de plataforma de prototipagem. Os principais requisitos avaliados durante a escolha foram:

- Compatibilidade com sistema operacionais abertos e compatíveis com Unix;
- Ampla disponibilidade de mercado;
- Conectividade Ethernet;
- Conectividade USB;
- Possibilidade de ligação de dispositivos de Smart-cards;
- Armazenamento em memória flash;
- Disponibilidade de alguns sensores embarcados;
- Interface para conexão de sensores externos;
- Possibilidade da ligação de displays externos;
- Disponibilidade de aceleradores criptográficos compatíveis;
- Disponibilidade de geradores de números aleatórios compatíveis;
- Interfaces de comunicação PCI e ou Mini-PCI;
- Custos para a aquisição do equipamento.

Após a avaliação de vários equipamentos disponíveis no mercado, selecionamos, por atender todos os requisitos, a placa Net 4801 da fabricante Soekris.

A placa Net 4801, como podemos ver na Tabela 7.1, atendeu parcialmente nossos requisitos, faltando unicamente alguns dos sensores detalhados na Seção 5.3.5. Seu propósito de desenvolvimento é voltado para roteadores e firewalls de internet. Ela é baseada num processador AMD Geode de 266Mhz e conta com 128Mb de memória RAM.

A adequação da placa ao protótipo do trabalho foi feita, desabilitando 2 de suas interfaces de rede e adquirindo uma memória compact-flash de 32 MB, para o

Tabela 7.1: Características da Plataforma Soekris

Componente:	**Descrição:**
Processador	Chip AMD SC1100 de 266 Mhz
Chip de Controle	Integrado no SC1100
Chip de Multi-IO	NSC PC87366
Memória Principal	128 Mbytes PC133 SDRAM soldada na placa
BIOS	FLASH de 512 Kbyte, soldada na placa rodando comBios.
Conectores de Expansão	1 soquete PCI 3.3V e 1 soquete Mini-PCI tipo IIIA
Conectores Ethernet	3 controladoras ethernet, suportado 10BaseT e 100BaseT
Portas Seriais	2 portas seriais
Armazenamento	1 soquete CompactFlash tipo I/II
Relógio de Tempo Real	Integrado no SC1100. Resguardado por uma bateria com duração mínima de 1 mês
Sensores	Monitor de temperatura integrado no PC87366 e monitor de tensão integrado no PC87366
Fonte de Alimentação	de 6 a 28V DC, com consumo máximo de 15W

armazenamento do sistema operacional embarcado e da aplicação gestora de chaves. Uma vantagem do uso desta placa foi a facilidade de operacionalizar o acesso aos sensores de hardware que monitoram a temperatura e a tensão da placa, visto que o fabricante garante total compatibilidade com o sistema operacional OpenBSD, também escolhido pelo projeto.

Neste equipamento foram feitos testes de desempenho, os quais não foram satisfatórios para a geração de chaves e para procedimentos criptográficos. Para tal foi adquirida um placa aceleradora de rotinas criptográficas.

7.2 Placa Aceleradora

A aquisição de uma placa de aceleração criptográfica deve-se ao fato do processador da placa de sistema ser de baixa velocidade, o que é plenamente justificável pela isenção de mecanismos de dissipação de calor.

Como requisitos para a aquisição da placa aceleradora de rotinas criptográficas foi levando em conta:

- A aceleração de rotinas simétricas;
- A aceleração de rotinas assimétricas;
- Possuir um gerador de números aleatórios baseado em mecanismos físicos;

- Ter interface compatível com a placa base (PCI ou mini-PCI);
- Possuir drivers abertos, auditáveis e implementados para sistemas livres;

As placas consideradas na avaliação final foram a Kryptus K1 de fabricação nacional, e a VPN-1411/1401 da fabricante Soekris. A escolha foi dada pela VPN-1411 tendo em vista a total compatibilidade com o equipamento base, o qual é da mesma fabricante, e a existência dos drivers abertos para os sistemas operacionais livres.

A VPN-1411 é a versão da placa de aceleração criptográfica distribuída pela Soekris que possui interface mini-pci e é baseada no processador criptográfico hi/fn 7955.

Tabela 7.2: Características da VPN-1411

Componente:	**Descrição:**
Chip Acelerador	Hi/fn 7955
Velocidade Nominal de Acesso	Capacidade de até 250 Mbps
Algoritmos de Compressão	LZS e MPPC com velocidade de 420 até 510 Mbps
Algoritmos de Cifragem Simétrica	AES 128/192/256, DES, 3-DES e RC4 com velocidade de 210 até 460 Mbps
Algoritmos de Autenticação	SHA-1 e MD5 com velocidade de 325 até 360 Mbps
Algoritmos de Cifragem Assimétrica	RSA, DSA, SSL, IKE e DH, com velocidade de 24 até 70 operações/segundo usando chaves de 1024 bits
Geração de Números Aleatórios	Feita em hardware próprio
Interface de Comunicação	Conector Mini-PCI tipo III com velocidade de 33/66 Mhz
Consumo de Energia	Máximo de 1.8 Watt
Temperatura de Operação	0-60 °C

O processador criptográfico utilizado na construção da VPN-1411 é focado para o uso em sistemas de VPN, e não conta com mecanismos de proteção das chaves sendo operadas, muito menos com certificação perante os Critérios Comuns e a FIPS PUB 140-2. Este fato não inviabiliza o seu uso, muito menos retira quaisquer méritos do processador, o qual é amplamente utilizado para este fim.

Na construção de um PSC para ambiente de produção real, deve ser considerada a produção de uma placa própria, pois sendo o processo de geração de números aleatórios controlado pela placa, a geração de chaves fracas ou previsíveis é possível.

Uma característica importante da VPN-1411 é seu suporte nativo no OpenBSD, sendo suportada para todos os processos nativos do sistema operacional. Outro fator importante é a capacidade do OpenBSD de uso de múltiplas placas, fazendo

um balanceamento de carga entre ambas, e fazendo assim o protótipo atingir índices de desempenho de PSCs comerciais.

7.3 OpenBSD

O OpenBSD é um sistema operacional multi-plataforma, baseado nas definições POSIX de interoperabilidade de sistemas operacionais, e um variante do Unix. É um sistema operacional bastante conhecido ter como característica de projeto a preocupação pró-ativa com segurança, fazendo para tal, esforços que vão desde a implementação de algoritmos de criptografia integrados no sistema operacional até a auditoria incessante de todo o código usado no projeto [48].

As preocupações existentes no projeto OpenBSD não se restringem unicamente à implementação de um sistema operacional seguro, mas também à disponibilidade deste sistema para todos. Por causa deste propósito, o projeto hoje é sediado no Canadá e é desenvolvido em vários países, principalmente naqueles que não impõem restrições à exportação de criptografia.

O mecanismo de auditoria usado pelo grupo que desenvolve o sistema operacional não serve explicitamente para a caça de problemas de segurança, mas sim para ver se os códigos estão bem escritos, se o seu desempenho é satisfatório, e principalmente se não há erros comuns que possam vir a ser explorados posteriormente por alguém que por ventura possa achar um problema no sistema operacional.

Uma grande preocupação do projeto OpenBSD é que se tenha um sistema operacional seguro por padrão, ou seja, ao se instalar o sistema ele já é configurado com os requisitos mínimos necessários ao funcionamento para garantir a sua segurança. Este fator é muito importante para a questão segurança, mas pode ocasionar alguns problemas de usabilidade na hora de torná-lo um sistema para um usuário final.

Como já foi dito anteriormente, os algoritmos criptográficos do OpenBSD não são desenvolvidos em países que regulam a exportação de criptografia, e no âmbito do projeto os componentes criptográficos que são utilizados atualmente foram escritos na Argentina, Austrália, Canadá, Alemanha, Grécia, Noruega e Suécia.

Um ponto importante do OpenBSD é que o mesmo hoje já é amplamente utilizado em sistemas embarcados, existindo assim todo um mecanismo de desen-

volvimento do mesmo para seu uso em ambientes embarcados.

No OpenBSD temos um pré-processador da configuração de compilação do núcleo do sistema que analisa o código da aplicação que será executada no ambiente embarcado e gera o núcleo de forma a conter única e exclusivamente o código necessário para execução da referida aplicação.

Em geral o sistema operacional OpenBSD conta com diversas características fundamentais para o projeto de um módulo de hardware seguro, já previamente implementadas, tais como os mecanismos criptográficos e o conceito de ambiente seguro de execução de programas, sendo tais funcionalidades, aliadas com a facilidade de torná-lo um sistema embarcado, o que justifica a escolha dele para ser a base da nossa plataforma embarcada de software.

7.4 Bibliotecas

A escolha das bibliotecas para uso no desenvolvimento da aplicação gestora de chaves e do sistema de controle de intrusão do protótipo se devem principalmente pelos requisitos de uso de ferramentas livres e pelo uso da linguagem C/C++ para a implementação.

As bibliotecas utilizadas foram escolhidas dentre uma gama de bibliotecas de código aberto, e foram levados em consideração a auditabilidade e a a garantia de desenvolvimeto e maturidade da comunidade que desenvolve estes projetos.

O uso de software livre nos permite o uso de bibliotecas prontas para tal desenvolvimento, pois a sua auditoria é garantida a qualquer momento, dando assim transparência a todos os processos criptográficos e de controle confiados a estas bibliotecas.

Estaremos usando basicamente quatro bibliotecas. A sub-seção 7.4.1, nos apresenta a biblioteca OpenSSL, uma biblioteca criptográfica de uso geral, que conta com a implementação de rotinas simétricas, assimétricas,de resumo e de gerenciamento de certificado, entre outras. Na sub-seção 7.4.2 temos a biblioteca OpenCT, a qual é utilizada para criarmos o suporte ao uso de leitores de smart-cards, aqui representando os DPCPCs. Ainda neste intuito de suportar um DPCPC, na sub-seção 7.4.3, tratamos do OpenSC, uma biblioteca para conversar com os sistemas operacionais que rodam inter-

namente aos smart-cards, o que nos permite operacionalizar os processos de armazenamento e operação com chaves dentro destes dispositivos. Por fim, a sub-seção 7.4.4 trata da biblioteca share-secret, a qual implementa um mecanismos para dividir segredo e nos possibilita criar o ambiente para o uso dos conjuntos de administradores e operadores descritos no capitulo 6.

7.4.1 OpenSSL

O OpenSSL é um sistema derivado do SSLeay, que foi originalmente escrito por Eric A. Young e Tim J. Hudson, no começo de 1995, com o intuito de criar uma biblioteca para a implementação do SSL, um protocolo seguro para a camada de transporte. Em dezembro de 1998, o projeto SSLeay foi finalizado, dando origem ao OpenSSL, e a partir deste momento tornando o código do mesmo livre e aberto para a comunidade, o que sem dúvida contribui para a transparência do projeto.

O OpenSSL foi criado com o intuito de implementar um conjunto de ferramentas para o protocolo SSL, mas para atingir tal objetivo, durante o desenvolvimento, foram sendo criadas várias outras estruturas, dentre elas, foram implementados de forma eficiente e concisa, uma biblioteca de funções criptográficas que conta com:

- Algoritmos de criptografia simétrica;
- Algoritmos de criptografia assimétrica;
- Funções de resumo criptográfico;
- Geradores de números aleatórios;
- Suporte ao gerenciamento de certificados digitais;
- Gerenciamento de chaves criptográficas;
- Gerenciamentos de números grandes, etc.

Também foi implementado junto com o OpenSSL um mecanismo para a criação e gerenciamento de certificados digitais, os quais são a base de todo o processo de comunicação SSL. Com estes mecanismos é possível a criação da base de uma infra-estrutura de chaves públicas, com a Implementação de autoridades certificadoras e o gerenciamento das sua respectivas chaves privadas [22].

O OpenSSL conta hoje com uma grande variedade de algoritmos simétricos de criptografia, dentre eles , Blowfish, Cast, Cast5, DES, Triple-DES, IDEA, RC2, RC4, RC5. Todos os algoritmos implementados pelo OpenSSL estão em conformidade com suas respectivas normas. Ele também conta com inúmeras funções de resumo criptográfico [1], dentre elas, MD2, MD5, MDC2, Ripemd-160, SHA e SHA-1 [22].

Os algoritmos assimétricos de criptografia implementados pelo OpenSSL são também os mais comumente utilizados em quaisquer ambientes que façam uso de criptografia de chave pública. Dentre eles temos o RSA, Diffie-Hellman e DSA.

O projeto é também único em sua implementação, pois é uma das poucas bibliotecas livres para criptografia disponíveis na Internet e que não sofre regulamentação de exportação de nenhum pais. Ele também é o único projeto completo de biblioteca para as linguagens C e C++, que são as bases hoje utilizadas na construção de sistemas operacionais abertos.

Do ponto de vista do projeto, o OpenSSL é uma ótima opção, pois é capaz de implementar todos os mecanismos criptográficos de uma forma rápida e já testada, assim como também integrar o uso de módulos de aceleração criptográfica já existentes hoje no mercado.

7.4.2 OpenCT

O OpenCT é uma camada de software intermediário, também conhecido como midleware, escrito por Olaf Kirch para fazer a interface entre terminais de leitura de smart-cards e software de mais alto nível que necessitam se comunicar com os smart-cards.

Ele atualmente suporta um grande número de leitores, os quais podem ser facilmente ligados a qualquer porta serial ou conexão USB para operação. Estes leitores normalmente são o caminho para acessar o sistema operacional interno dos smart-cards. Eles não efetuam nenhuma transformação nos dados que por eles passam, atuando unicamente como meio de transmissão dos dados.

Para uso com o OpenCT, escolhemos o leitor Towitoko ChipDrive 110, o qual possui uma interface USB, e é 100% compatível com o OpenCT. A escolha deste

[1] Função Hash

leitor também levou em consideração a sua interface USB, a qual é compatibilizada pelo hardware base do protótipo.

O OpenCT inicializa um processo no sistema operacional e simplesmente faz a interface de comunicação entre a aplicação e o smart-card.

7.4.3 OpenSC

O OpenSC é uma grande caixa de ferramentas para o desenvolvimento de aplicações baseadas no uso de smart-cards. O seu ponto principal é a biblioteca OpenSC, a qual tem 3 camadas de código entrepostas cada qual com vários drivers de comunicação com dispositivos e outros softwares. Outras bibliotecas presentes no OpenSC são os módulos para comunicação usando PKCS#11, um módulo de autenticação PAM e dois motores de ligação com o OpenSSL, além de inúmeras ferramentas para testar leitores e cartões.

A biblioteca OpenSC, também conhecida com libopensc, é usada por todos os componentes, e ele oferece as funcionalidades básicas tais como comunicar com smart-cards e também funções avançadas como gerar chaves dentro dos smart-cards. As várias camadas de funcionalidades do OpenSC são:

Leitor O OpenSC precisa de um meio para comunicação com os leitores de smart-cards e com os cartões colocados nestes leitores. Vários softwares de midleware são suportados, cada qual com o seu driver. Um deles é o OpenCT.

Cartão A maioria dos smart-cards deveria implementar o padrão ISO 7816 e então aceitar e responder de maneira homogênea a estes comandos. Infelizmente existem cartões que fogem do padrão e esta camada tem por objetivo suavizar esta diferenças.

PKCS#15 Os smart-cards normalmente tem um sistema de arquivos onde são armazenadas ou criadas a chaves, certificados e quaisquer outros dados que se queiram armazenar neste cartões. O PKCS#15 é o padrão de como criar estas estruturas de armazenamento, mas os diferentes fabricantes de cartões implementam o padrão com mecanismos de controle de acesso próprio. Esta camada tem por objetivo viabilizar a comunicação com o sistema de arquivos de diferentes cartões, suavizando as diferença entre os cartões.

Para a implementação dos recursos de smart-cards no protótipo, foram escolhidos os smart-cards da fabricante alemã G&D da série SPK 2.3, os quais são completamente suportados pelo OpenSC. A escolha e fixação de um modelo de leitor e um modelo de cartão devem-se ao fato de que o protótipo é uma implementação de testes, sendo que a implementação de suportes a vários leitores e cartões poderia inviabiliza-lo, mas este suporte pode ser fácilmente implementado.

7.4.4 Share Secret

A biblioteca share secret foi implementada por Stefan Karrmann [63] para fazer o compartilhamento de arquivos para várias pessoas, usando métodos polinômicos e baseados em operadores lógicos.

A implementação do share secret é uma das poucas existentes com código livre e aberta para análise. A incorporação da biblioteca no projeto foi precedida por uma série de modificações na mesma para que ela passasse a operar não só com arquivos mas sim com estruturas de memória, como as chave criptográficas que queremos dividir.

A entropia dos processos de divisão é dada pelos geradores de entropia presentes no sistema operacional, que no nosso caso é o OpenBSD, com uma placa para geração de entropia. A biblioteca usa o método de Aitken-Neville para interpolação de polinômios como base para a desmontagem e remontagem dos segredos. O fato da escolha deste método deve-se ao fato de que, de acordo com a sua implementação o X do polinômio sempre será zero, sendo assim o método que mais rápido converge.

A implementação da biblioteca pode ser muito melhorada quanto aos quesitos de gerenciamento de memória e quanto a otimização das rotinas implementadas, fazendo com que ela seja muito mais dinâmica e funcional para a divisão de quaisquer tipos de segredos.

7.5 Aplicação Gestora de Chaves Criptográficas

A aplicação gestora de chaves é o núcleo do processo de prototipagem, a qual mais tempo consumiu. Ela é a implementação dos conceitos propostos nos capítulos 5 e 6.

A aplicação gestora foi implementada usando-se linguagem C/C++,

usando uma arquitetura cliente-servidor baseada em protocolo TCP/IP. A parte cliente foi implementada para ser utilizada em qualquer plataforma, de forma que possamos monitorar e gerenciar o PSC enviando mensagens e processos para ele através de uma conexão SSL.

A aplicação servidor é a principal parte do protótipo, e é a que roda internamente ao equipamento escolhido para protótipo. A aplicação implementa todos os processos descritos no capítulo 6, demonstrando na prática a viabilidade de implementação e a coesão da idéia lá apresentada.

Esta aplicação gestora é também alvo de um trabalho de conclusão de curso do bacharelado em sistema de informação e contou com 2 alunos de graduação em ciência da computação na sua implementação. Pelos requisitos do trabalho de conclusão de curso, foram adotadas todas as medidas para o gerenciamento do projeto de software, e de mecanismos de desenvolvimento, sempre focando o atendimento ao estabelecido nos critérios comuns. O resultado final do protótipo pode ser visto na Figura 7.1

Figura 7.1: Visão do Protótipo do PSC

7.6 Conclusões

Este capítulo mostrou o processo de implementação do protótipo que incluis os mecanismos descritos nesta dissertação. Foi detalhado o processo de escolha dos

mecanismos de hardware e de aceleração criptográfica necessários ao desenvolvimento.

Também foram vistos os detalhes de um sistema operacional voltado para o embarque de aplicações de segurança, o qual é aberto e completamente passível de auditoria. Foram vistas quatro bibliotecas que nos deram apoio para o desenvolvimento do protótipo, dando ênfase na escolha de soluções de software livre pela sua transparência e qualidade.

Por fim ,foi demonstrado um pouco da arquitetura do projeto da aplicação de gerenciamento de chaves que implementa os conceito proposto nos capítulos anteriores. Esta aplicação também deu origem a um trabalho de conclusão de curso do bacharelado em sistema de informação, levando alunos de graduação ao aprendizado do uso das bibliotecas e dos cuidados necessários ao desenvolvimento de aplicações tão sensíveis quanto esta.

Capítulo 8

Considerações Finais e Trabalhos Futuros

O trabalho apresentado teve como grande objetivo o estudo dos equipamentos existentes no mercado para gerenciamento de chaves criptográficas e a proposta de um mecanismo que fosse diretamente voltado para os requisitos de uso em ICPs.

Inicialmente foram revistos alguns conceitos relativos a ICP e criptografia em geral, foram levantados os estados da arte de projeto nacionais e de pesquisas acadêmicas internacionais da área deste projeto. Deste ponto ficou esclarecido que o mercado hoje procura por produtos que estejam em conformidade com normas internacionais pra tal.

As normas internacionais foram estudadas e delas foram retirados todos os quesitos relevantes, tanto para avaliar quanto para projetar um bom equipamento passível da obtenção de certificados de conformidade com estas normas. As normas mais gerais servem não exclusivamente para a construção de PSCs, mas também para o desenvolvimento de qualquer sistema que tenha a segurança como seu foco.

Posteriormente, foram avaliados vários equipamentos de mercado para confrontá-los com nossas necessidades no ambiente de ICP e como eles se comportariam. Este estudo foi complementado pela aquisição de um equipamento da empresa nCipher, sendo ele a base dos estudos. Isso nos mostrou como realmente um PSC trabalha no mercado e nos esclareceu muito sobre questões relativas a proteção de chaves e desempenho.

Com base nos estudos das normas e dos PSC existentes no mercado, lançamos um projeto para a construção de um PSC que fosse diretamente voltado para

o ambiente de ICP, uma vez que nos estudos preliminares ficou claro que este não era o objetivo dos produtos no mercado. Desta proposta de projeto, surgiram problemas, principalmente quanto ao processo ideal para o gerenciamento de chaves por parte de um PSC.

Da necessidade de um mecanismo de gerenciamento de chaves consistente surge o núcleo deste trabalho, que é a proposta de um mecanismo para gerenciamento interno de um PSC, primando pelas características necessárias e esperadas de um PSC voltado unicamente para ICPs. Com este trabalho proposto, necessitávamos de um protótipo que demonstrasse a viabilidade da implementação deste projeto, assim como da corretude dos mecanismos proposto para tal.

Desta necessidade surgiu o projeto de prototipagem de um PSC usando hardware e aceleradores criptográficos, combinando com software livre para atingir tal objetivo. A implementação deste protótipo nos levou a compreender as dificuldades inerentes aos processos de gerenciamento de chaves, e também do processo da construção de um PSC para um ambiente de produção real.

Por fim, este trabalho abre áreas de pesquisa inexploradas no Brasil, tais como a evolução deste equipamento para um propósito geral de proteção de chaves, para o embarque de aplicações genéricas em equipamentos com proteções físicas a fim de proteger estes processos e chaves.

Uma proposta natural de evolução deste PSC é a criação de um ambiente seguro para execução de código, dando assim abertura para que processos sensíveis não sejam colocados em ambientes aonde não possamos controlar o seu acesso. Outra área aberta por este trabalho de execução segura de código pode ser o estabelecimento de métodos formais para validar o código sendo executado por estes PSCs, dando assim abertura para que além de executarem num ambiente seguro, ele estejam formalmente verificados quanto à sua corretude.

Referências Bibliográficas

[1] DIAS, J. da S. *Confiança no Documento Eletrônico*. Tese (Doutorado) — Universidade Federal de Santa Catarina, Setembro 2004.

[2] NIST. *FIPS PUB 46 - Data Encryption Strandard*. 1977.

[3] NIST. *FIPS PUB 197 - Advanced Encryption Strandard*. 2001.

[4] NIST. *FIPS PUB 81 - DES Modes of Operation*. 1980.

[5] SCHNEIER, B. *Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd Edition*. 2. ed. [S.l.]: John Wiley and Sons, 1995.

[6] HOUSLEY, R.; POLK, T. *Planning for PKI: Best Practices Guide for Deploying Public Key Infrastructure*. 1. ed. [S.l.]: Wiley, 2001.

[7] NIST. *FIPS PUB 180 - Secure Hash Standard*. 1993.

[8] DIFFIE, W.; HELLMAN, M. New directions on cryptographic techniques. *Proceedings of the AFIPS National Computer Conference*, 1976.

[9] RIVEST, R.; SHAMIR, A.; ADELMAN, L. *A Method for Obtainig Digital Signatures and Public-Key Cryptosystems*. [S.l.], 1977. 15 p.

[10] BRASIL. *Medida Provisória 2.200-2*. Agosto 2001. Media Provisória que instituíu a ICP-Brasil.

[11] MARTINI, R. *Anteprojeto - Construção de Plataforma Criptografica Aberta para a ICP-Brasil*. Outubro 2003. Renato.Martini@planalto.gov.br.

[12] PAGLIUSI, P. S. O programa joão de barro - a marinha, a presidência e a sociedade na construção de uma plataforma criptográfica aberta para a icp-brasil.

[13] Rede Nacional de Pesquisa. *GT ICP-EDU II*. Janeiro 2005. Http://www.rnp.br/pd/gts2004-2005/chaves_publicas.html.

[14] GRAAF, J. van de. Icp-edu: unificando as icps no âmbito acadêmico. *Boletim bimestral sobre tecnologia de redes*, v. 7, n. 2 e 3, 2003. Disponível em: <http://www.rnp.br/newsgen/0303/icp.html>.

[15] IBM. Ibm 4758 model 2 and 23 pci cryptographic cooprocessor - ibm coorporation. 2000.

[16] NCIPHER. ncipher security world.

[17] NCIPHER. *nCipher Technical Specifications*. Outubro 2003. Http://www.ncipher.com/nshield/nshield_specs.html.

[18] BOND, M. *Understanding Security APIs*. Tese (Doutorado) — University of Cambridge, 2004.

[19] KOHNFELDER, L. *Towards a pratical public-key cryptosystem*. [S.l.], 1978.

[20] ITU. *"Recommendation X.509 – Information Technology – Open Systems Interconnection – The Directory: Authentication Framework"*. [S.l.], 1988.

[21] ITU. *"Recommendation X.509 (11/93) – Information Technology – Open Systems Interconnection – The Directory: Authentication Framework"*. [S.l.], 1993.

[22] CHANDRA, P.; MESSIER, M.; VIEGA, J. *Network Security with OpenSSL*. [S.l.]: O Reilly, 2002.

[23] NIST. *FIPS PUB 140-2 - Security Requirements for Cryptographic Modules*. December 2002.

[24] International Organization for Standardization. *ISO/IEC 15408-3:1999 Information tecnology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction adn general model*. 1999.

[25] International Organization for Standardization. *ISO/IEC 15408-3:1999 Information tecnology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction adn general model*. 1999.

[26] International Organization for Standardization. *ISO/IEC 15408-3:1999 Information tecnology - Security techniques - Evaluation criteria for IT security - Part 3: Security assurance requirements*. 1999.

[27] KILLMANN, W. et al. *Protection Profile - Secure Signature-Creation Device Type 1*. Julho 2001. Http://www.commoncriteriaportal.org/public/files/ppfiles/pp0004b.pdf.

[28] KILLMANN, W. et al. *Protection Profile - Secure Signature-Creation Device Type 2*. Julho 2001. Http://www.commoncriteriaportal.org/public/files/ppfiles/pp0005b.pdf.

[29] KILLMANN, W. et al. *Protection Profile - Secure Signature-Creation Device Type 3*. Julho 2001. Http://www.commoncriteriaportal.org/public/files/ppfiles/pp0006b.pdf.

[30] NIST. *FIPS PUB 140-1 - Security Requirements for Cryptographic Modules*. January 1994.

[31] SHAMIR, A. How to share a secret. *Communications of the ACM, Volume 22, pages 612-613*, 1979.

[32] NIST. *Programa de validação de módulos criptográficos*. Outubro 2003. Http://www.nist.gov/cmvp. Disponível em: <http://www.nist.gov/cmvp>.

[33] Department of Defense. *Trusted Computer System Evaluation Criteria - TCSEC*. [S.l.], Dedcember 1985.

[34] ITSEC. *Information Technology Security Evaluation Criteria - Harmonised Criteria of France, Germany, the Netherlands, the United Kingdom*. 1. ed. [S.l.]: ITSEC, 1990.

[35] NIST. *Advanced Encryption Standard Algorithm Validation List*. Março 2005. Http://csrc.nist.gov/cryptval/aes/aesval.html.

[36] NIST. *DES Validation Lists*. Março 2005. Http://csrc.nist.gov/cryptval/des/desval.html.

[37] NIST. *Triple DES Validation List*. Março 2005. Http://csrc.nist.gov/cryptval/des/tripledesval.html.

[38] NIST. *DSA Validation List*. Março 2005. Http://csrc.nist.gov/cryptval/dss/dsaval.htm.

[39] NIST. *SHS Validation List*. Março 2005. Http://csrc.nist.gov/cryptval/shs/shaval.htm.

[40] AEP. *AEP/Sureware*. Outubro 2003. Http://www.aepsystems.com.

[41] HP. *HP's Atalla Net Security Products*. Outubro 2003. Http://h20138.www2.hp.com/object/AT8100PD.html.

[42] HP. Raising the bar: Security processing for the new era. 2002.

[43] RAINBOW. *Cryptoswift/ Rainbow Products*. Outubro 2003. Http://www.rainbow.com/products/cryptoswift/HSM.asp.

[44] IBM. *IBM Hardware: IBM PCI Cryptographic Coprocessor*. Outubro 2003. Http://www-3.ibm.com/security/cryptocards/html/overhardware.shtml.

[45] NCIPHER. Hardware security modules, deploying startegies for enterprise security.

[46] SOMMERVILLE, I. *Engenharia de Software*. 6. ed. [S.l.]: Addison Wesley, 2003.

[47] CUSTóDIO, R. F. Padrões de hardware na infra-estrutura de chaves públicas. In: *Fórum do ITI sobre Padrões de Hardware na ICP*. Brasilia: [s.n.], 2003.

[48] OpenBSD Project. *OpenBSD*. Outubro 2003. Http://www.openbsd.org/.

[49] SCHIFFMAN, M. *Stephanie for OpenBSD*. Outubro 2003. Http://www.innu.org/ brian/Stephanie.

[50] KALISKI, B. *RFC 1319 - The MD2 Message-Digest Algorithm*. [S.l.], April 1992.

[51] RIVEST, R. *The MD5 Message Digest Algorithm*. [S.l.], April 1992.

[52] DOBBERTIN, H.; BOSSELAERS, A.; PRENEEL, B. Ripemd-160, a strengthened version of ripemd. In: GOLLMANN, D. (Ed.). *Fast Software Encryption*. [S.l.]: Springer-Verlag, 1996. LNCS, n. 1039, p. 71–82.

[53] International Standards Organization. *ISO 8372 - Modes of operation for a 64-bit block cipher algorithm*. [S.l.], 1987.

[54] NIST. *FIPS PUB 198 - The Keyed-Hash Message Authentication Code (HMAC)*. Março 2002.

[55] NIST. *FIPS PUB 46-3 DATA ENCRYPTION STANDARD (DES)*. Outubro 1999.

[56] ADAMS, C. *The CAST-128 Encryption Algorithm*. [S.l.], May 1997.

[57] ADAMS, C.; GILCHRIST, J. *RFC 2612 - The CAST-256 Encryption Algorithm*. [S.l.], June 1999.

[58] SCHNEIER, B. Description of a new variable-length key, 64-bit block cipher (blowfish). In: SPRINGER-VERLAG (Ed.). *Fast Software Encryption*. [S.l.: s.n.], 1994. Cambridge Security Workshop, p. 191–204.

[59] SCHNEIER, B. et al. Twofish: A 128-bit block cipher. Junho 1998.

[60] ANDERSON, R.; BIHAM, E.; KNUDSEN, L. Serpent: A proposal for the advanced encryption standard.

[61] NIST. *Digital Signature Standard*. [S.l.], May 1994.

[62] ELGAMAL, T. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, IT-31, n. 4, p. 469?472, 1985.

[63] KARRMANN, S. *ShareSecret*. Março 2005. Http://www.mathematik.uni-ulm.de/m5/sk/sharesecret.html.

Apêndice A

Glossário

Algoritmo Assimétrico: algoritmo usado por cifradores que utilizam par de chave: chave pública/privada. Enquanto uma chave é usada para cifrar a outra é usada para decifrar.

Algoritmo Simétrico: algoritmo usado por cifradores que utilizam uma chave secreta para cifrar. São mais rápidos do que os algoritmos assimétricos.

Assinatura Digital: transformação matemática de uma mensagem por meio da utilização de uma função matemática e da criptografia assimétrica do resultado desta com a chave privada da entidade assinante.

Autenticidade: garante a identidade de quem está enviando a mensagem, ou seja, poderemos assegurar a autoria de determinado documento. No documento tradicional demonstra-se essa autoria através da assinatura no documento. No documento eletrônico prova-se sua autenticidade com a assinatura digital.

Autoridade Certificadora: entidade que emite certificados de acordo com as práticas definidas na Declaração de Praticas de Certificação. É comumente conhecida por sua abreviatura - AC.

Autoridade de Registro: entidade de registro. Pode estar fisicamente localizada em uma AC ou ser uma entidade de registro remota. É parte integrante de uma AC.

Cifrador: programa que contém um algoritmo usado para cifrar mensagens ou arquivos, geralmente utilizando chaves pública/privada ou chave secreta.

Certificado Digital: declaração assinada digitalmente por uma AC, contendo nome de uma AC, que emitiu o certificado; nome do assinante para quem o certificado foi emitido; a Chave Pública do assinante; o período de validade operacional do certificado; o número de série do certificado, único dentro da AC; uma assinatura digital da AC que emitiu o certificado com todas estas informações.

Chave Privada: chave de um par de chaves mantida secreta pelo seu dono e usada no sentido de criar assinaturas para cifrar e decifrar mensagens com a Chave Pública correspondente.

Chaves Pública: chave de um par de chaves criptográficas que é divulgada pelo seu dono e usada para verificar a assinatura digital criada com a chave privada correspondente

ou, dependendo do algoritmo criptográfico assimétrico utilizado, para cifrar e decifrar mensagens.

Criptografia: disciplina que trata dos princípios, meios e métodos para a transformação de dados, de forma a proteger a informação contra acesso não autorizado a seu conteúdo.

Criptografia Assimétrica: sistema criptográfico que envolve o uso de um par de chaves matematicamente relacionadas, uma chave pública e outra privada.

Criptografia Simétrica: sistema criptográfico que utiliza a mesma chave para cifrar e decifrar o texto.

Documento Eletrônico: informações manipuladas por computador e armazenadas em programa específico capaz de traduzir uma seqüencia de bits.

Firmware: Componente de software embarcado em um componente de hardware, e tem por função implementar um comportamento específico do equipamento.

Irretratabilidade (não repúdio): garantia de que o emissor da mensagem não irá negar posteriormente a autoria de uma mensagem ou participacão em uma transação, controlada pela existência da assinatura digital que somente ele pode gerar.

Infra-Estrutura de Chaves Públicas: arquitetura, organização, técnicas, práticas e procedimentos que suportam, em conjunto, a implementação e a operação de um sistema de certificação baseado em criptografia de Chaves Públicas.

Integridade: garantia de que o conteúdo da mensagem não foi alterado. No caso dos documentos eletrônicos esta verificação é determinada pela assinatura digital.

Lista de Certificados Revogados: lista dos números seriais dos certificados revogados, que é digitalmente assinada e publicada em um repositório. A lista contém ainda a data da emissão do certificado revogado e outras informações, tais como as razões específicas para a sua revogação.

Resumo Criptográfico: um conjunto de caracteres mapeado de uma mensagem ou arquivo por uma função resumo que único. Se a mensagem ou arquivo sofre alterações, o resumo não será o mesmo. Geralmente usado em assinaturas digitais para garantir a integridade do objeto.

Segredo Compartilhado: Mecanismo pelo qual se divide um segredo em partes e através de um sub-conjunto destas partes podemos remontar o segredo inicialmente dividido. Normalmente é usado em métodos onde queremos dividir as responsabilidades de guarda de chaves.

Sigilo: Condição na qual dados sensíveis são mantidos secretos e divulgados apenas para as partes autorizadas.

Smart card: É um cartão que tem seu próprio microprocessador embutido, completo, com seu próprio sistema operacional, e pode processar e armazenar dados independentemente.

Tempestividade: permite saber se determinado documento foi ou não produzido naquela ocasião.