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

Jeandré Monteiro Sutil

GESTÃO SEGURA DE MÚLTIPLAS INSTÂNCIAS DE UMA MESMA CHAVE DE ASSINATURA EM AUTORIDADES CERTIFICADORAS

Florianópolis
2011

Jeandré Monteiro Sutil

GESTÃO SEGURA DE MÚLTIPLAS INSTÂNCIAS DE UMA MESMA CHAVE DE ASSINATURA EM AUTORIDADES CERTIFICADORAS

Dissertação submetida ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina para a obtenção do grau de Mestre em Ciência da Computação.

Ricardo Felipe Custódio
Orientador

Florianópolis
2011

Catalogação na fonte pela Biblioteca Universitária
da
Universidade Federal de Santa Catarina

S966 Sutil, Jeandré Monteiro
Gestão segura de múltiplas instâncias de uma mesma chave de assinatura em autoridades certificadoras [dissertação] / Jeandré Monteiro Sutil ; orientador, Ricardo Felipe Custódio. - Florianópolis, SC, 2010.
94 p.: il., tabs.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação.

Inclui referências

1. Ciência da computação. 2. Criptografia de dados (Computação). 3. Certificação digital. 4. Infraestrutura de Chaves Públicas. I. Custodio, Ricardo Felipe. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU 681

Jeandré Monteiro Sutil

GESTÃO SEGURA DE MÚLTIPLAS INSTÂNCIAS DE UMA MESMA CHAVE DE ASSINATURA EM AUTORIDADES CERTIFICADORAS

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina.

Florianópolis, 02 de março de 2011

Mário Antônio Ribeiro Dantas, Dr.
Coordenador do Curso

Banca Examinadora:

Ricardo Felipe Custódio
Orientador
Universidade Federal de Santa Catarina

Prof. Antonio Alfredo Ferreira Loureiro, Dr.

Prof. Mehran Misaghi, Dr.

Prof^{a}. Michelle Wangham, Dra.

Prof. Mário Antônio Ribeiro Dantas, Dr.

O mais importante não é ter inteligência, mas aquilo a que se há-de aplicar.
(Vergílio Ferreira)

À minha família, amigos, mestres e sobretudo à minha noiva Carla, pelo apoio, que contribuíram para tornar o trabalho muito mais "leve" .

AGRADECIMENTOS

Engana-se quem pensa que um trabalho de mestrado é redigido a duas ou quatro mãos. Este documento foi escrito a muitas mãos. Deve-se considerar as mãos do mestrando e do orientador, ao redigir cada palavra, mas há também que se valorizar a mão dos pais, da companheira, dos amigos, mestres, como tantos outros colaboradores.

Agradeço a cada uma dessas valiosas contribuições. Aos pais e irmãos, de quem sempre tive o apoio, incentivo e o investimento necessários à minha formação profissional e pessoal.

Ao Prof. Ricardo Felipe Custódio, o meu agradecimento não só como orientado, mas também pelos ensinamentos que levo para a vida, tal como o valor do trabalho sério, ético e o respeito ao próximo.

A minha noiva Carla, agradeço pela paciência, suporte (que muitas vezes significou suportar a ansiedade e mau humor do noivo mesmo), mas sobretudo pelo amor e dedicação nos momentos mais delicados ao longo dessa etapa.

Aos amigos do LabSEC, por comporem o ambiente agradável e fértil, onde o conhecimento nasce e é reproduzido sem que sequer se perceba, regado a muito suor e boas risadas. Seria injusto citar apenas parte dos amigos, mas são muitos para citar a todos. Assim, aos que estão longe ou próximos, o meu muito obrigado.

(Jeandré Monteiro Sutil)

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

AC Autoridade Certificadora

AES Advanced Encryption Standard

AR Autoridade de Registro

DES Data Encryption Standard

ICP Infraestrutura de Chaves Públicas

IETF Internet Engineering Task Force

ITU-T International Telecommunication Union

LCR Lista de Certificado Revogado

MSC Módulo de Segurança Criptográfico

NIST National Institute of Standards and Technology

OCSP Online Certificate Status Protocol

OUP Originator Usage Period

PCI Peripheral Component Interconnect

PUO Período de Uso pelo Originador

PUD Período de Uso pelo Destinatário

RFC Request for Comment

RUP Recipient Usage Period

X.509 Padrão de certificação digital

RESUMO

Chaves criptográficas são hoje parte fundamental na proteção de dados e informações que trafegam por canais inseguros de forma análoga às chaves físicas na proteção de patrimônios, pessoas, entre outros valores. Uma chave, seja física ou lógica, impede ou dificulta o acesso não autorizado a um bem valioso, por intermédio de um segredo ou combinação de difícil previsão ou composição.

Nesses materiais, o procedimento de replicação é de suma importância para garantir que, mesmo que uma das cópias de determinada chave torne-se indisponível, seja por perda, roubo ou excesso de demanda, o acesso às informações por ela protegidas não seja interrompido.

Entretanto, mesmo em *hardwares* dedicados exclusivamente à gestão do ciclo de vida de chaves, é comum que a segurança seja por vezes preterida em nome da garantia de disponibilidade. Isso se dá em virtude dos modelos de gestão existentes, que unicamente levam em consideração a existência de uma instância da chave ou que trata as diferentes instancias separadamente.

O presente trabalho propõe dois novos modelos de gestão voltados a chaves assimétricas de autoridades certificadoras, que levem em consideração as múltiplas cópias possíveis de uma mesma chave criptográfica, de forma que seu custodiante ou grupo de custodiantes jamais percam o controle sobre nenhuma de suas instâncias. Com isso, busca-se possibilitar a rastreabilidade das cópias, agregando as propriedades alta disponibilidade e tolerância a faltas à gestão de chaves, sem contudo comprometer sua segurança.

Com os modelos propostos, foi possível identificar univocamente cada nova instância de uma mesma chave privada, através de rastros que liguem a chave pai à sua filha. Com isso, a disponibilidade foi beneficiada, sem contudo prejudicar o controle sobre a utilização das múltiplas instâncias, por parte dos custodiantes.

Palavras chave: Gestão de Chaves Criptográficas, Criptografia, Certificação Digital, Infraestrutura de Chaves Públicas, Autoridade Certificadora, Gestão do Ciclo de Vida de Chaves, Criptografia.

ABSTRACT

Cryptographic keys consist on an essential component to provide data security over the electronic environment. Associated with an algorithm, they are responsible for protecting transactions, information and other electronic assets providing authenticity, confidentiality and integrity.

To assure the availability of the protected assets, however, the keys must be strictly controled. One of the aspects in key lifecycle management is the secure replication of keys, in order to avoid the loss of so valuable resources, in case of unavailability of the key.

This work shows that even in hardwares dedicated to securely manage keys lifecycle, namely HSM, security is sometimes underestimated in name of the availability.

This work proposes two new models for key management, allowing the replication of key material, without compromise the control over the managed keys. The main goal of the methods is to keep a track among all derivates of the original key, balancing availability assurances with the control of the multiple instances by the key owners.

Keywords: Digital Certification, Public Key Infrastructure, Digital Certificate, Certification Authority, Key Lifecycle Management, Cryptography

1 INTRODUÇÃO

Chaves criptográficas são hoje parte fundamental na proteção de dados e informações que trafegam por canais inseguros, de forma análoga às chaves físicas na proteção de patrimônios, pessoas, entre outros valores. Uma chave, seja física ou lógica, impede ou dificulta o acesso não autorizado a um bem valioso, por intermédio de um segredo ou combinação de difícil previsão ou composição.

Para que uma chave física tenha utilidade, seu segredo deve estar atrelado a uma ou mais fechaduras que, mediante a um procedimento bem definido e de conhecimento geral (como inserir e girar a chave, por exemplo), controlarão o acesso a estes recursos.

É igualmente comum a utilização de ferramentas específicas para transportar o segredo, contido em uma chave, para outra, procedimento este conhecido como cópia da chave. O procedimento de replicação é de suma importância para garantir que, mesmo se uma das cópias de determinada chave vier a ficar indisponível, seja por perda, roubo ou degradação de seu material, o acesso ao valor protegido não será interrompido.

As propriedades supracitadas podem também ser verificadas em meio digital, onde busca-se garantir, com o emprego de técnicas de criptografia, propriedades como sigilo, integridade e autenticidade a informações valiosas, mantidas e transmitidas em meio eletrônico.

Nesse contexto, as chaves passaram a ser representadas por um conjunto de bits, representando também um segredo, utilizado em conjunto com funções criptográficas (fechadura), comumente chamadas de algoritmo criptográfico ou cifrador. É importante notar que um algoritmo criptográfico em geral é composto por duas funções, a de ciframento e a de deciframento, sendo a segunda utilizada para a reversão da primeira.

Até meados da década de 70, conhecia-se somente a criptografia de chave única, ou *simétrica* (NIST, 1977). Nesse tipo de criptografia, uma mesma chave é utilizada para cifrar ou decifrar informações, tal como ocorre com fechaduras convencionais, em sistemas de controle de acesso como portas, cofres, guarda volumes, entre outros.

Em 1976, Diffie e Hellman (1976) propuseram o uso de um par de chaves distintas para cifrar e decifrar informações, ao invés de uma só chave, como era usual à época. A ideia era utilizar uma das chaves para a operação de cifragem e a outra para decifragem. Este tipo de criptografia ficou conhecida como criptografia assimétrica ou de chave pública.

Tratando-se adequadamente esse par de chaves, é possível prover, além do sigilo, a autenticidade das informações, escondendo-se uma das chaves e tornando a outra pública. A chave oculta é conhecida como chave de assinatura ou chave privada.

Em tese, uma informação cifrada com a chave privada só pode ser

decifrada com a chave pública correspondente. Associando-se um par de chaves a uma entidade, é possível reconhecer a origem de uma dada informação. Uma entidade detentora de determinada chave privada pode cifrar informações e distribuí-las a um terceiro, que poderá por sua vez verificar sua autenticidade e autoria, fazendo uso da respectiva chave pública.

Em 1978, foi proposto por Rivest et al. (1978) um método para processamento de assinaturas digitais em meio eletrônico, com base no trabalho de Diffie e Helman, dando origem ao algoritmo RSA. Hoje o algoritmo RSA é um dos algoritmos assimétricos mais utilizados na garantia de autenticidade, integridade e irretratabilidade, propriedades essenciais a contratos, acordos, cartas, entre outros documentos de nosso cotidiano, passíveis de representação em meio digital.

O surgimento do conceito de assinatura digital permitiu um grande avanço na forma como entidades autenticam-se e comunicam-se, umas perante as outras, em meio digital. O problema passou então a ser a forma de se estabelecer a associação entre uma determinada entidade e sua chave, de forma totalmente eletrônica, sem a necessidade de um acordo prévio de chaves em meio real. Uma solução para este problema permitiria maior escalabilidade e aceitação da técnica, dada a globalidade e agilidade, possíveis com o uso da rede mundial de computadores.

Com vista a solucionar o problema de acordo de chaves, duas propostas ganharam destaque e encontram-se amplamente difundidas. A primeira diz respeito às Infraestruturas de Chaves Públicas (ICPs) (ITU-T, 2005), estruturas de natureza hierárquica, marcadas pela existência de terceiras partes, confiáveis a um conjunto de entidades. Tais terceiros confiáveis, conhecidos como Autoridades Certificadoras ou ACs, são responsáveis pelo credenciamento das entidades. Essas ACs emitem certificados digitais, que associam uma entidade a sua chave pública. O padrão X.509 define os procedimentos e políticas necessários à gestão das ICPs, sendo o modelo mais aceito em nível global.

Uma abordagem diferente é a do PGP[1] (GARFINKEL, 1994), solução proposta por Phil Zimmermman, baseada em uma rede de confiança em que entidades assinam as chaves públicas umas das outras. Tais chaves e assinaturas são publicadas em repositórios de livre acesso, que podem ser baixadas por qualquer outro usuário. A ideia principal do PGP é a de uma rede de contatos, ou seja, mesmo que Beto não conheça Alice, ele pode confiar com um alto grau de certeza que Alice é quem clama ser, desde que "amigos" de confiança de Beto assegurem essa correspondência. Hoje o PGP é uma solução muito utilizada para a proteção de mensagens de correio eletrônico em pequenos domínios, tal como um laboratório de pesquisa ou o setor de desenvolvimento de uma empresa.

Seja para os usuários de PGP ou para as entidades afiliadas a uma ICP, a proteção da identidade do titular de uma chave está intimamente relacionada à segurança de sua chave privada. Assim, torna-se fundamental que o par de chaves criptográficas seja mantido sob os cuidados deste

[1] Do Inglês *Pretty Good Privacy*.

durante toda sua vida útil.

Uma chave criptográfica transita entre diferentes fases ao longo de sua vida, como a de geração, distribuição, armazenamento, replicação, suspensão, revogação, arquivamento e destruição. Ao conjunto de medidas e mecanismos utilizados para garantir a segurança da chave ao longo das fases citadas, dá-se o nome de Gestão do Ciclo de Vida de Chaves[2].

Para garantir a gestão segura do ciclo de vida de chaves criptográficas, foram desenvolvidos sistemas em *software* e/ou *hardware* dedicados a essa tarefa, comumente chamados de módulos criptográficos.

Um dos principais desafios no tocante à gestão chaves está relacionado ao controle de suas cópias. Apesar de um procedimento sensível, a cópia de uma chave é por vezes necessária para garantir sua disponibilidade, seja em situações de falha dos módulos que a gerenciam, seja para satisfazer a uma alta demanda de assinaturas.

Como boa prática, a chave privada usada para assinatura digital não deveria ter cópias. Caso a chave seja destruída ou comprometida, é de praxe revogar a chave e proceder a geração de um novo par de chaves. Este é o caso que ocorre com usuários finais, aqueles que usam suas chaves exclusivamente para assinar mensagens digitais.

Entretanto, existem situações onde faz-se necessário ter cópias das chaves. Isso ocorre, por exemplo, com as autoridades certificadoras. Dois são os motivos. O primeiro é em situações de alta demanda de geração de assinaturas, tal como a de uma Autoridade Certificadora *online* que pode demandar a emissão de milhares de certificados num curto espaço de tempo. Neste caso, são utilizados mais de um sistema emissor de certificados com um sistema de balanceamento de cargas distribuindo as demandas entre estes sistemas.

O segundo motivo é que a chave privada não pode ser perdida, ou seja, na ocorrência eventual de sua destruição deveria haver uma cópia da mesma para dar prosseguimento ao serviço de assinatura, uma vez que a revogação desta implicaria na revogação de todos os certificados já emitidos.

No entanto, permitindo-se a cópia de chaves privadas, aumenta-se o risco de comprometimento do sistema de assinatura. Para diminuir esse risco, os módulos criptográficos existentes no mercado, os quais são usados por ACs, dispõe de serviços de geração e recuperação de cópia de chaves privadas. Esses serviços não são, todavia, padronizados, e as empresas vinculam a permissão de cópias a produtos da própria companhia. Isso impede, por exemplo, que a chave privada de uma AC gerada no módulo criptográfico da empresa X possa ser transportada para um módulo da empresa Y. Justifica-se esse procedimento para dificultar o vazamento da chave de uma AC e, portanto, comprometer toda uma infraestrutura.

Um problema que surge deste comportamento dos desenvolvedores de módulos criptográficos é que equipamentos têm um tempo de vida útil muito menor que o tempo estimado de uso de uma AC. Comumente,

[2]Do Inglês *Key Lifecycle Management*.

ACs podem ter uma vida de dez, vinte ou até mais anos. Já os módulos de segurança são construídos para uma vida, em geral, de não mais de cinco anos. Há também a possibilidade da empresa fabricante deixar de produzir o equipamento ou mesmo deixar de existir.

Todo o rígido controle necessário sobre as chaves privadas, aliado à necessidade de realização do procedimento de cópia, sem contudo comprometer o controle do custodiante sobre sua chave, tornam o problema da gestão de chaves uma tarefa extremamente complexa e ainda em aberto.

O presente trabalho propõe um modelo de gestão que leve em consideração as múltiplas cópias possíveis de uma mesma chave criptográfica ao longo de sua vida, de forma que seu custodiante ou grupo de custodiantes jamais percam o controle sobre nenhuma de suas instâncias. Busca-se, com este modelo, balancear os conceitos de alta disponibilidade, rastreamento e tolerância a faltas, com a gestão segura do ciclo de vida de chaves criptográficas.

1.1 OBJETIVOS

Vistas as principais preocupações na busca por uma gestão segura de chaves criptográficas, são apresentados os objetivos deste trabalho, bem como as contribuições almejadas.

1.1.1 Geral

Este trabalho tem como objetivo geral tratar o problema da gestão de chaves criptográficas sob uma ótica que se preocupe com o controle das múltiplas instâncias de uma mesma chave, ao longo de sua vida útil, buscando sempre mantê-las sob o controle dos custodiantes. O modelo busca ainda tornar evidente o procedimento de backup ou replicação, por intermédio de um rastro, amarrando a chave original às suas réplicas. O objetivo com isso é permitir um processo de auditoria mais simples e com menor custo, bem como a identificação das diferentes instâncias.

1.1.2 Específicos

Como objetivos específicos do trabalho, pode-se listar:

- Avaliar as técnicas e modelos de gestão de chaves existentes na literatura;
- Verificar a segurança das chaves, sobretudo quanto à segurança do procedimento de replicação, nos modelos de gestão avaliados;
- Propor uma técnica que permita a gestão de múltiplas instâncias de uma mesma chave, de forma simultânea;
- Comparar a abordagem proposta às técnicas existentes.

1.2 MOTIVAÇÃO

Durante muito tempo o desenvolvimento de módulos de segurança criptográfica (MSC) esteve restrito a empresas estrangeiras, que sempre buscaram proteger seus produtos, tornando seus protocolos e modelos de gestão extremamente obscuros. Essa realidade faz com que módulos criptográficos sejam hoje conhecidos como *caixas pretas*, dispondo de protocolos e políticas de segurança voltadas à gestão de chaves criptográficas.

No ano de 2003, iniciou-se no Laboratório de Segurança da Computação (LabSEC), na Universidade Federal de Santa Catarina, o desenvolvimento de um projeto ambicioso, com o objetivo principal de desenvolver um MSC com tecnologia nacional, em ambiente acadêmico.

O projeto, inicialmente intitulado OpenHSM, compunha um de maior abrangência, idealizado e financiado pela Rede Nacional de Ensino e Pesquisa (RNP), o ICPEDU. O projeto ICPEDU, por sua vez, almejava a criação e gestão de uma infraestrutura de chaves públicas educacional, de forma a vincular instituições de ensino e pesquisa federais ao longo de todo o território nacional.

Alguns anos depois, em 2006, o projeto OpenHSM lançava seu primeiro protótipo e, nos anos subsequentes, entrava em fase de produção à medida que o projeto ICPEDU ingressava em sua fase piloto, visando avaliar as soluções produzidas em um ambiente de teste, restrito a um número menor de instituições.

No ano de 2008, o projeto ICPEDU entrou em sua segunda fase, de produção, encontrando-se hoje já difundido entre cerca de 20 instituições federais, provenientes de diferentes partes do Brasil, constituindo-se na maior ICP educacional existente do país. O projeto encontra-se em franca expansão, com o ingresso de um número crescente de instituições.

Por ter se tornado um produto competitivo, o OpenHSM passou a despertar a atenção de outras instituições nacionais como a ICP-Brasil, Infraestrutura de Chaves Públicas Brasileira, que passou a adotar o MSC como parte de sua plataforma criptográfica.

No ano de 2009, o MSC da ICPEDU passou a ser chamado de ASI-HSM, sendo OpenHSM mantido como o nome do protocolo implementado. Ainda naquele ano, o MSC apresentou sua segunda versão, contemplando inúmeras novidades sobre a versão anterior, como a adição de um canal autorizado de externalização dos estados do módulo, na forma de um um visor, bem como suporte a novos algoritmos criptográficos, para uso no âmbito da ICP-Brasil.

Com a avaliação da realidade das instituições que utilizavam o módulo criptográfico, tanto na ICPEDU quanto na ICP-Brasil, percebeu-se a necessidade de satisfazer a altas demandas, bem como a manutenção de múltiplos ambientes simultâneos, o que acaba por acarretar um procedimento de replicação do material criptográfico e uma dificuldade na auditoria das Autoridades Certificadoras.

A constatação de tal deficiência nos modelos de gestão de chaves

existentes culminaram no desenvolvimento do presente trabalho.

1.3 JUSTIFICATIVA

À medida que documentos, processos e negócios convergem para o meio digital, cresce a preocupação com formas de garantir a autoria, autenticidade, integridade e o sigilo dessas informações. Nesse contexto, a criptografia tem se consolidado como ferramenta imprescindível para o alcance dos objetivos anteriormente citados. Algoritmos criptográficos, por sua vez, baseiam sua segurança na escolha de uma chave secreta imprevisível, como forma de prover ofuscamento das informações protegidas, de forma praticamente impossível de ser burlada.

Tendo ciência da dificuldade em atacar um algoritmo criptográfico consolidado, a atenção de um atacante volta-se às chaves criptográficas, tendo em vista que um único ataque bem sucedido pode abrir caminho para uma série de fraudes e danos. Daí a importância de se proteger tais materiais ao longo de toda a sua vida útil. Apesar dos requisitos de proteção necessários, a disponibilidade das chaves criptográficas é também uma questão importante, que deve ser balanceada com os requisitos de segurança. Essa não é uma tarefa simples, mas que necessita ser melhor estudada, sendo este o objetivo do presente trabalho.

Para resolver o problema de disponibilidade sem contudo relaxar a segurança e controle das chaves, espera-se identificar unicamente as múltiplas instâncias possíveis de uma mesma chave privada, com vista a permitir aos custodiantes seu rastreamento.

1.4 METODOLOGIA

Para o desenvolvimento deste trabalho, foram avaliados os principais trabalhos de pesquisa, normas e padrões que regem a gestão de chaves criptográficas, bem como o projeto e desenvolvimento de módulos de segurança criptográficos.

A partir do conhecimento adquirido, pode-se elencar as principais propriedades necessárias à gestão segura destes artefatos, principalmente na de chaves assimétricas.

Com o estudo de caso acerca da segurança provida por um módulo de segurança criptográfico de mercado, foi possível avaliar a segurança dos procedimentos de cópias de segurança existentes.

Baseado na análise crítica dos modelos existentes, bem como dos problemas ainda remanescentes, foram propostos dois novos modelos de gestão, contemplando as garantias de disponibilidade das chaves, sem que seja comprometido o controle sobre o material, por parte de seus custodiantes.

Por fim, através da realização de uma análise comparativa com os modelos existentes, tornou-se possível a avaliação da proposta.

1.5 LIMITAÇÕES

Este trabalho restringe-se aos requisitos lógicos relacionados à gestão de chaves criptográficas, no contexto dos módulos de segurança criptográficos, apesar de apresentar de modo geral os requisitos físicos relacionados à construção do hardware criptográfico. Maiores detalhes acerca da construção de dispositivos criptográficos podem ser encontrados em (MARTINA, 2005; NIST, 2002).

1.6 TRABALHOS CORRELATOS

Como trabalho correlato a este pode-se citar o proposto por Chaum e Heyst (1991), que propõe o uso de assinaturas de grupo, do inglês *Group Signatures*.

Com base no trabalho de Chaum, outras propostas surgiram para tratar do problema da assinatura em grupo. Ateniese e Tsudik (1999) faz um apanhado geral das propostas, listando as seguintes propriedades aos sistemas:

Não Forjabilidade: Apenas membros do grupo podem assinar mensagens em prol do grupo;

Anonimidade: O receptor da assinatura pode verificar a validade de uma assinatura, mas não pode identificar qual membro do grupo assinou a mensagem;

Independência: Deve ser computacionalmente difícil saber que duas assinaturas quaisquer foram assinadas por membros de um mesmo grupo;

Não Repúdio: Não deve ser possível ao administrador do grupo nem a um de seus membros assinar por outro membro;

Rastreabilidade: Em caso de disputa, um grupo especial de administração do grupo poderá revelar a identidade do assinante;

Resistência a Conluio: Um ou mais membros de um grupo não devem ser capazes de gerar assinaturas válidas pelo grupo sem que sejam rastreadas.

Este trabalho utiliza o conceito de grupo de instâncias de chaves privadas para se referir as múltiplas possíveis instâncias de uma chave privada. As propostas deste trabalho, entretanto, diferem-se das relacionadas ao tema de assinatura em grupo, nos seguintes pontos:

Anonimidade: No contexto de uma autoridade certificadora, o receptor da assinatura deve poder identificar a instância utilizada na assinatura, sobretudo caso exista um procedimento para isolamento das instâncias comprometidas;

Independência: Ao contrário do que acontece com assinaturas em grupo, deve ser fácil identificar que duas assinaturas diferentes foram geradas pela mesma chave da AC;

Rastreabilidade: A rastreabilidade das chaves não deve ficar restrita a um único grupo, mas ser de conhecimento público.

Além das questões acima enunciadas, deve ainda ser trivial o isolamento de uma instância ou mesmo a revogação de uma chave privada. Criar novas instâncias de uma mesma chave privada deve ser igualmente simples, sendo desejável a criação de quantas instâncias forem necessárias. Dependendo do esquema de assinatura em grupo, tais características podem demandar um alto esforço computacional, que acabam por tornar sua aplicação no contexto de infraestruturas de chaves impraticável.

1.7 ORGANIZAÇÃO DA DISSERTAÇÃO

Este trabalho estrutura-se da seguinte forma: no capítulo 2, serão vistos os principais conceitos envolvidos na gestão de chaves criptográficas, bem como os modelos existentes na literatura. O capítulo 3 fazem um apanhado geral dos componentes de uma infraestrutura de chaves públicas, com um enfoque voltado ao ciclo de vida de suas chaves privadas.

No capítulo 4 é detalhado o conceito de módulo de segurança criptográfico e apresentado um estudo de caso relacionado à extração indevida de uma chave privada, gerenciada por um módulo criptográfico real. O capítulo 5 apresenta algumas formas como vem sendo garantida a disponibilidade da chaves, seus problemas e limitações.

O capítulo 6 propõe duas novas abordagens para a gestão de chaves, com suporte a múltiplas instâncias de uma mesma chave privada, de forma a permitir alta demanda, sem deixar de lado a rastreabilidade. O primeiro modelo baseia-se em novas primitivas criptográficas, que evitariam a cópia da chave. O segundo apresenta um segundo modelo, buscando atingir as mesmas metas do anterior, sem contudo depender das referidas primitivas.

Por fim, o capítulo 7 dispõe algumas considerações finais, além de elencar algumas possibilidades de trabalhos futuros.

2 CICLO DE VIDA DE CHAVES CRIPTOGRÁFICAS

2.1 INTRODUÇÃO

Conforme apresentado no capítulo 1, chaves criptográficas estão presentes nos mais variados sistemas computacionais, aplicações e soluções. Com a adoção da criptografia como forma de proteção de informações e identidades em meio digital, as chaves tomam uma posição de destaque, sendo assim alvo de constantes tentativas de acesso indevido.

Neste capítulo serão apresentados os principais conceitos envolvidos na gestão segura de chaves criptográficas, em cada uma das fases de seu ciclo de vida. Adicionalmente, serão abordadas as garantias de segurança necessárias para uma efetiva gestão desses objetos. Ao fim do capítulo, serão expostos os problemas que dificultam a tarefa de garantir segurança ao ciclo de vida das chaves criptográficas.

2.2 CHAVES E ALGORITMOS CRIPTOGRÁFICOS

Segundo o princípio de Kerckhoffs (1883), o segredo provido por um sistema criptográfico bem desenvolvido deve basear sua segurança unicamente na escolha de suas chaves. Em outras palavras, segundo Kerckhoffs, o mecanismo utilizado para realizar a transformação em si deve ser publicamente conhecido ou ao menos ter sido concebido de forma que sua exposição ao inimigo não afete a segurança por ele proporcionada.

O princípio baseia-se no fato de que é menos custoso atualizar uma chave do que um algoritmo, caso venham a público. Dessa forma, a segurança provida por técnicas de criptografia moderna baseia-se no segredo de suas chaves, bem como no esforço exigido de um atacante, buscando revelá-lo.

Tamanha importância na proteção de dados e transações faz da gestão de chaves criptográficas a parte mais difícil da criptografia (SCHNEIER, 1996). Desenvolver algoritmos e protocolos de segurança não é uma tarefa fácil, mas sem o devido cuidado com suas chaves, de nada vale o esforço. Ainda, ciente da dificuldade em quebrar a segurança de um algoritmo, um atacante possivelmente verá, na tentativa de ataque à chave, uma opção mais atraente do que um ataque sofisticado, com base em criptoanálise.

Entende-se por gestão de chaves criptográficas, como sendo o conjunto de medidas a serem tomadas com o intuito de proteger o material sensível que as compõem, contra qualquer tentativa de acesso não autorizado ao seu conteúdo (MENEZES et al., 1996).

Algoritmos criptográficos, de modo geral, possuem uma estimativa de vida útil baseada no crescimento do poder computacional ao longo

dos anos. Geralmente, fazem uso de problemas matemáticos comprovadamente *difíceis* de serem resolvidos com os recursos atuais, ou mesmo disponíveis em um futuro próximo. Isso faz com que as chaves em si também tenham uma estimativa de validade. Neste caso, há ainda um segundo fator importante, a força da chave, relacionada geralmente ao seu tamanho, em bits, bem como a sua imprevisibilidade (aleatoriedade).

O Instituto Nacional de Padrões e Tecnologias Norte Americano (NIST), reconhecido por suas pesquisas em tecnologia aplicada às mais variadas áreas do conhecimento, publica periodicamente relatórios contendo recomendações de transição entre tamanhos de chaves e algoritmos criptográficos. Os mais recentes, nomeados SP 800-57 e SP 800-131, datando respectivamente de março de 2007 (BARKER et al., 2007) e junho de 2010 (BARKER; ROGINSKY, 2010), apontam os seguintes períodos de transição, para os tamanhos apontados, em bits:

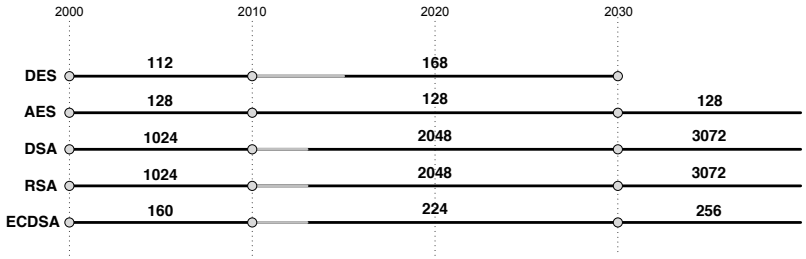

Figura 2.1: Ciclo de vida de chaves e algoritmos criptográficos, de acordo com o NIST.

Segundo a figura 2.1, pode-se observar os pontos de transição para os tamanhos mínimos necessários às chaves criptográficas dos principais algoritmos em uso. Os dois primeiros, *Data Encryption Standard* (*DES*) e *Advanced Encryption Standard* (*AES*), são algoritmos criptográficos simétricos. Os três últimos, por sua vez, representam algoritmos assimétricos, cuja principal finalidade é a assinatura digital.

Além do prazo de validade do próprio algoritmo criptográfico, outros fatores podem determinar a vida útil de uma chave, como o uso a que se destina, a criticidade e validade da informação por ela protegida, ou ainda a descoberta de novas vulnerabilidades, antes ocultas nos algoritmos. Dentro dessa realidade, o NIST aponta períodos de transição sob os quais os tamanhos de chave antigos podem ser utilizados de forma transicional, com um risco aceitável. Esses períodos apresentam-se realçados em tom mais claro, nas linhas do tempo de cada algoritmo.

Dada a natureza transitória de algoritmos e principalmente de chaves criptográficas, os modelos de gestão de chaves baseiam-se em um ciclo de vida bem definido, compreendendo as possíveis fases por que uma chave criptográfica pode transitar, desde sua geração até sua completa destruição. A seção 2.3 define os conceitos relativos ao ciclo de vida de

chaves, além de detalhar os modelos de gestão existentes na literatura.

2.3 CICLO DE VIDA DE CHAVES CRIPTOGRÁFICAS

O ciclo de vida de uma chave consiste em um conjunto de fases, compreendido entre sua geração e a completa destruição de seu material sensível, quando ficará definitivamente indisponível para uso. A "Gestão Segura do Ciclo de Vida de Chaves" implica não somente no armazenamento seguro de chaves durante sua vida útil, como também na geração, distribuição, destruição e arquivamento desses materiais (FUMY; LANDROCK, 1993). Em cada uma das etapas citadas, controles devem ser aplicados, de forma a evitar a divulgação, modificação, substituição, replicação e uso não autorizados.

Chaves criptográficas possuem diferentes propósitos, sendo que cada um exige um nível diferente de cuidados. Uma mesma chave simétrica pode, por exemplo, ser utilizada para criptografar uma comunicação por uma única sessão, sendo destruída após seu encerramento. Uma segunda finalidade para essa chave seria seu emprego na proteção de um conjunto de arquivos a serem armazenados por um longo prazo. Outra prática comum é a cifragem de uma chave criptográfica por uma segunda chave criptográfica, procedimento chamado de embalagem ou *wrapping*, formando dessa forma uma cadeia de confiança. Diferentes finalidades podem implicar em níveis de proteção distintos . Recomenda-se portanto o emprego de cada chave para uma única finalidade, de forma a simplificar a gestão de seu ciclo de vida (BARKER et al., 2007).

2.3.1 Cripto Período

O cripto período, do Inglês *cryptoperiod*, corresponde ao prazo de validade em que uma chave criptográfica poderá ser utilizada na proteção ou recuperação de informações em meio eletrônico, de forma que a proteção seja eficaz.

Um cripto período adequadamente definido contribui para:

- Limitar a quantidade de informação protegida por uma chave, uma vez que quanto mais a chave é utilizada, mais ela se torna atraente aos olhos de um atacante (SCHNEIER, 1996);
- Limitar o prejuízo ocasionado por um eventual comprometimento da chave;
- Limitar o uso de uma chave a um período onde sua proteção é efetiva (vide figura 2.1);
- Limitar o tempo disponível para tentativas de acesso não autorizado à chave;
- Limitar o tempo hábil para ataques baseados em criptoanálise.

A duração do cripto período está intimamente ligada ao propósito a que a chave é destinada. Um par de chaves assimétricas, por exemplo, pode ter cripto períodos distintos para suas porções pública e privada. É o caso de uma chave utilizada para assinatura digital. Muito embora a chave privada tenha seu cripto período pré fixado, a chave pública em si poderá ser utilizada após a expiração desse prazo, desde que a primeira não tenha sido comprometida durante o seu período ativo. Com chaves simétricas, onde a mesma chave é utilizada para cifragem e decifragem de dados, é possível também dividir o cripto período segundo o uso da chave. Ao período onde uma chave simétrica é utilizada para cifrar dados, dá-se o nome de Período de Uso pelo Originador (PUO), do inglês *Originator Usage Period.*. Em contrapartida, o período onde a chave é utilizada para decifrar dados, criptografados durante o PUO, é conhecido como Período de Uso pelo Destinatário (PUD), do inglês *Recipient Usage Period.*

Com relação ao tempo de duração do cripto período, chaves podem ser classificadas como de curto e longo prazo. Chaves de curto prazo compreendem as utilizadas para proteger uma única transação. Esse tipo de chave é comumente conhecida como chave de seção. Já as chaves de longo prazo contam com um cripto período maior, como as utilizadas para proteger dados de arquivamento ou mesmo outras chaves. Após ser atingida a data limite definida para o cripto período de uma chave, esta deverá ser substituída.

A tabela 2.1 apresenta os cripto períodos recomendados pelo NIST (BARKER et al., 2007), para os principais tipos de chaves criptográficas, quando utilizadas para segurança de dados e informações em longo prazo.

Tipo	**Finalidade**	**Cripto Período**	
		Originador	**Destinatário**
Privada	Assinatura	1-3 anos	
	Autenticação	1-2 anos	
	Transporte de chaves	$\leq$ 2 anos	
	Acordo de chaves	1-2 anos	
Pública	Assinatura	limitado pela força da chave	
	Autenticação	1-2 anos	
	Transporte de chaves	1-2 anos	
	Acordo de chaves	1-2 anos	

Simétrica	Cifragem de dados	$\leq$ 2 anos	$\leq$ PUO + 3 anos
	Autenticação	$\leq$ 2 anos	$\leq$ PUO + 3 anos
	Transporte de Chaves	$\leq$ 2 anos	$\leq$ PUO + 3 anos
	Acordo de chaves	1-2 anos	

Tabela 2.1: Cripto períodos recomentados pelo NIST para os principais usos de chaves criptográficas.

Um sistema que faça uso de chaves criptográficas para proteção de dados e informações deve sempre respeitar os cripto períodos acima listados, com o intuito de garantir a efetiva proteção do conteúdo sigiloso. Contudo, muitas vezes a necessidade de proteção supera os cripto períodos recomendados para uma chave. Nesses casos, torna-se fundamental a existência de procedimentos para substituir chaves criptográficas que expiraram, processos estes conhecidos como atualização ou rotação de chaves.

Além da atualização, existem outras preocupações durante o ciclo de vida de chaves que acabam por definir diferentes fases e modelos de gestão. Esses modelos serão apresentados a seguir, nas seções 2.3.2 e 2.3.3.

2.3.2 Fases

O ciclo de vida de chaves criptográficas pode ser subdividido em um conjunto de quatro fases distintas. São elas:

Pré Operacional: Nesta fase as chaves, apesar de terem sido geradas, encontram-se indisponíveis para uso;

Operacional: A fase operacional é aquela em que as chaves já encontram-se disponíveis para utilização, seja para proteção (cifragem), ou para processamento de informações protegidas (decifragem);

Pós Operacional: Na fase pós operacional as chaves não são mais utilizadas para proteger informações. Contudo, é possível obter acesso a elas com o intuito de processar informações protegidas;

Destruída: Nessa fase a chave não mais poderá ser recuperada. Todos os vestígios de seu conteúdo devem ter sido destruídos. Somente atributos da chave podem ser obtidos para fins de auditoria, como nome, tipo e cripto período.

Com base na visão acima descrita, envolvendo as fases do ciclo de vida de chaves, a seção 2.3.3 apresenta alguns modelos de gestão de chaves encontrados na literatura.

2.3.3 Modelos e Estados

Há na literatura diferentes modelos de gestão do ciclo de vida de chaves. Todavia, cada um deles apresenta os estados básicos apresentados no início desta sessão.

Menezes et al. (1996) apresentam um ciclo de vida completo para chaves criptográficas, detalhando as transições entre cada um dos estados. O modelo leva também em consideração as quatro diferentes fases descritas na seção 2.3.2. A figura 2.2 ilustra o modelo proposto.

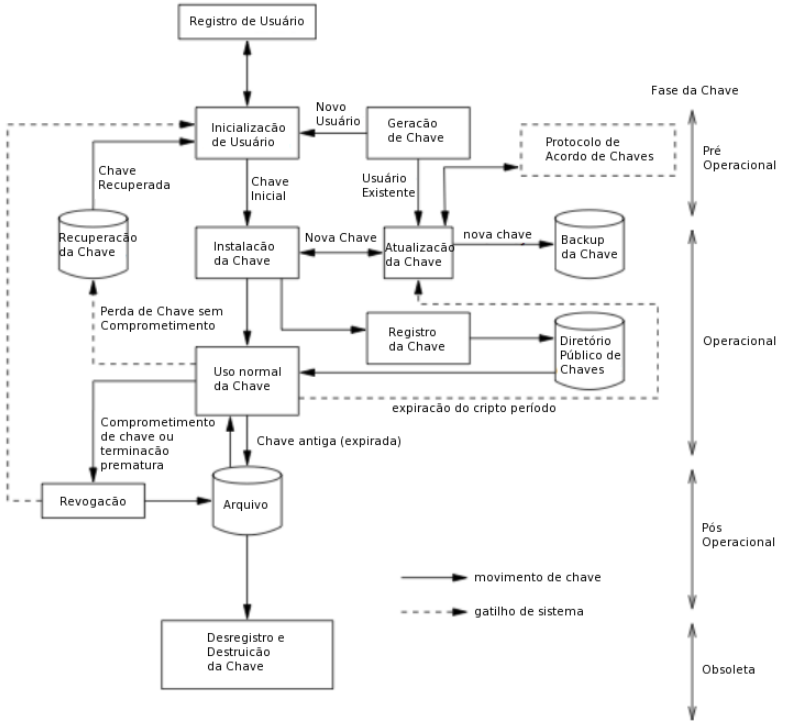

Figura 2.2: Ciclo de vida completo de chaves criptográficas, ilustrando seus estados, transições e fases.

O ciclo de vida apresentado na figura 2.2 inicia com o estado de *geração de chave*, onde esta é atribuída a um novo usuário, transitando para os estados de *inicialização e registro do usuário* titular. Estes estados fazem parte da fase *pré operacional* da chave. Segundo o modelo, na transição para o estado de registro de usuário, este deverá apresentar uma credencial para acesso ao material. A nova chave é então instalada e entra em sua *fase operacional*, em que estará disponível para uso normal.

Em se tratando de uma atualização de chave, o registro e a inicia-

lização do usuário são desnecessários, o que pode ser notado na transição direta entre os estados de *atualização de chave* e *instalação de chave*. Durante o estado de atualização pode ainda ser necessário realizar uma cópia de segurança da nova chave, conforme ilustrado no diagrama, com o estado de *backup da chave*.

No caso de chaves utilizadas para garantir irretratabilidade, deve haver uma associação com uma entidade, representada pelos estados *registro de chaves* e *repositório público de chaves*.

Chaves em uso podem sofrer revogação antes do fim de seu cripto período, em caso de suspeita de comprometimento. Outra possibilidade é tornarem-se indisponíveis por falha na mídia de armazenamento, fato que salienta a importância do procedimento de backup. O primeiro caso implica em transição para o estado de *revogação* e posteriormente ao estado de *arquivamento*. Nos casos de revogação ou fim de seu cripto período, uma chave transitará para a fase *pós operacional*, em que será arquivada, onde encontrar-se-á disponível para casos de futuras disputas. Caso não haja mais nenhum motivo para manter esta chave registrada em nome de um usuário, poderá ainda haver uma desassociação entre a chave e a entidade, com posterior destruição da chave, onde os vestígios de seu material serão apagados com segurança.

O modelo proposto, apesar de completo, apresenta-se demasiadamente complexo, além de misturar estados do ciclo de vida de chaves com procedimentos relacionados à sua utilização. É o caso da publicação em um repositório público, bem como o registro de um usuário. Ao mesmo tempo, o modelo não contempla transições importantes, como por exemplo, o backup de uma chave recém gerada para uma nova entidade, após sua devida instalação. Um segundo problema do modelo diz respeito à transição reversa entre a fase pós operacional e a operacional, o que não faz sentido após uma revogação, ou mesmo expiração do cripto período. Mesmo em casos de disputa, a chave jamais deverá ser devolvida ao seu estado operacional, caso em que poderia ser utilizada para forjar o sigilo, integridade e autenticidade de uma informação falsa.

A publicação especial SP 800-57[1] (BARKER et al., 2007), do NIST, descreve um ciclo de vida de chaves criptográficas semelhante ao anterior, com simplificações que facilitam sua interpretação e implementação, consistindo no seguinte conjunto de estados:

Pré-ativação: Uma chave é dita em estado de pré-ativação logo após ter sido gerada, com seu uso não tendo ainda sido autorizado para a finalidade de proteção de informação. O único uso possível para chaves neste estado é em operações de prova de posse;

Ativa: Nesse estado, a chave pode ser utilizada para criptografar informações, bem como para o processo reverso de decifragem. Esta é a fase que marca de fato o período de utilização da chave;

[1] Do inglês *Special Publication 800-57*.

Desativada: Nesse estado, o período de tempo pelo qual a chave era necessária foi alcançado, sendo não mais utilizada para proteger informações. Contudo, a chave nesse estado ainda pode ser utilizada para processar informações criptografadas durante sua fase ativa. É o caso por exemplo de um par de chaves assimétricas cujo certificado já tenha expirado, caso em que a chave privada não mais serviria para assinatura de dados. No entanto, enquanto houver dados a serem verificados, a chave pública precisará ser mantida. Tal par de chaves estaria no estado de desativação;

Destruída: Nesse estado, as chaves não são mais necessárias e, portanto, foram destruídas. No caso exemplificado anteriormente, seria o momento em que a chave pública deixou de ser necessária, pois não há mais informações relevantes cuja integridade necessite de verificação;

Comprometida: As chaves atingem esse estado por terem tido seu conteúdo exposto seja por acesso não autorizado ao seu material ou na ocorrência de alguma inovação que permitiu sua previsibilidade. Neste estado as chaves em geral não devem mais ser utilizadas para proteção de informações. Contudo, há situações especiais em que ainda pode ser utilizada para verificar integridade de dados;

Destruída e Comprometida: Chaves neste estado podem ter sido comprometidas e por consequência destruídas, bem como o contrário, quando uma chave já havia sido destruída e, posteriormente, notou-se que seu conteúdo havia sido comprometido.

Já na recomendação SP 800-130 (BARKER et al., 2010), foram incorporados ainda dois estados aos supracitados, com o intuito de cobrir os casos em que chaves sejam suspensas ou revogadas durante seu período ativo. O referidos estados são:

Suspensa: Uma chave passa ao estado *Suspensa* quando seu uso é interrompido de forma reversível, ou seja, podendo voltar ao estado ativo. Durante o período em que se encontrar suspensa, uma chave não deverá ser utilizada em nenhuma das finalidades a que se destina;

Revogada: Quando revogada, uma chave não mais poderá ser utilizada, seja para proteção ou verificação de dados. Uma forma de revogar chaves é fazer uso de listas de certificados revogados (COOPER et al., 2008).

A figura 2.3 ilustra os diferentes estágios ao longo da vida de uma chave criptográfica, bem como as possíveis transições entre os estados, segundo os dois documentos da série SP 800:

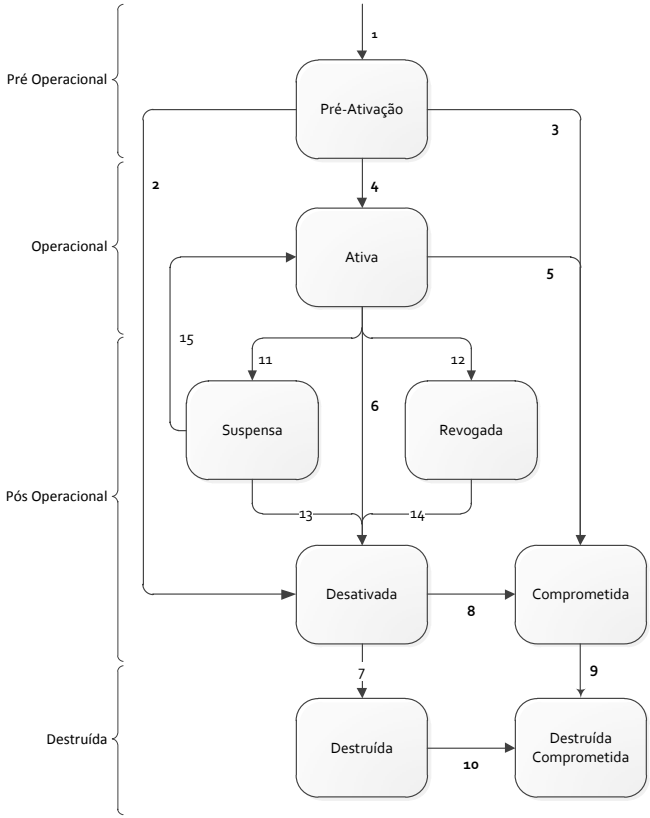

Figura 2.3: Estados do ciclo de vida de chaves criptográficas e suas transições, segundo a SP 800-57 e SP 800-130.

Segundo o modelo da figura 2.3, pode-se perceber que o primeiro estado de uma chave é sempre o de *Pré-ativação*. Logo após ter sido gerada (transição *1*), a chave entra nesse estado. Caso jamais seja utilizada, a chave pode transitar do estado de pré-ativação diretamente para o estado *Destruída* (transição *2*). Esta transição é importante pois, apesar de uma chave não ter sido utilizada, deve haver evidências de que ela existiu e manteve-se íntegra ou secreta até sua destruição. Caso tal integridade ou confidencialidade não tenham sido garantidas e a chave recém gerada tenha sido assim prematuramente comprometida, deverá seguir a transição *3*, passando diretamente ao estado *Comprometida*.

A partir da disponibilização de uma chave recém gerada para proteção de informações, esta passa ao estado *Ativa* (transição *2*). Nesse estado a chave será utilizada para proteger dados, verificar dados protegidos ou mesmo para as duas atividades em paralelo. Caso uma chave ativa

seja comprometida, deverá, a exemplo do que ocorre com uma chave pré-ativada, transitar para o estado *Comprometida*, seguindo a transição *5*. Porém, se ao longo de toda sua vida útil, tal chave manteve-se íntegra, jamais tendo havido indícios de que seu conteúdo tenha sido revelado, chegará ao fim do seu PUO. Pode haver, entretando, necessidade de utilizá-la para verificar informações outrora protegidas (PUD), o que implica em transição para o estado *Desativada*, marcada pela conexão *6*.

Uma vez no estado *Desativada*, uma chave poderá transitar para o estado *Destruída* (transição *7*), caso em que a mesma concluiu seu cripto período da forma esperada, ou ainda para o estado *Comprometida*. No último caso, existiram suspeitas de que a chave possa ter sido comprometida durante o estado de desativação, o que justifica a transição *8*.

Ambos os estados *Destruída* e *Comprometida*, poderão ainda convergir para um estado comum com, respectivamente, um comprometimento ou uma destruição das referidas chaves. Estas transições são ilustradas em *9* e *10*, demarcando o fim da vida útil de uma chave criptográfica.

O ciclo de vida proposto pelo NIST considera apenas o ponto de vista da chave criptográfica em si, não considerando estados importantes, como o registro e a distribuição destes artefatos. O modelo volta-se também a uma única instância da chave privada, não contemplando um procedimento de cópia da chave, para fins de contingência ou atendimento à alta demanda. Um estado interessante introduzido pelo modelo do NIST, no entanto, é o estado de *Suspensão*. Tal estado é bastante útil em caso de suspeita de comprometimento, que poderia então ser avaliada, com procedimentos de auditoria sobre registros de utilização da chave. Em caso de constatação da falha de segurança, a chave seria desativada. Caso contrário, esta poderia voltar ao seu estado operacional.

Em (MOULDS, 2008), um terceiro ciclo de vida de chaves criptográficas é definido, contemplando estados mais condizentes com que realmente é implementado pelos módulos de gestão existentes. A figura 2.4 ilustra o referido modelo.

O modelo apresentado pela figura 2.4 apresenta os principais estados necessários à gestão segura do ciclo de vida de chaves criptográficas. Os destaques são os estados como o de *Distribuição*, *Backup* e *Restauração*, *Rotação* e *Suspensão*. É importante ressaltar que o estado de suspensão, neste modelo, tem um sentido diferente de seu homógrafo, no modelo proposto pelo NIST. Naquele, o estado implica em uma revogação temporária, enquanto neste denota apenas uma consequência da revogação em si, com a suspensão dos serviços a que a chave se destina.

Os estados apresentados na figura2.4, bem como as preocupações em garantir a segurança de cada um deles, são detalhados nas Seções 2.3.4 a 2.3.13.

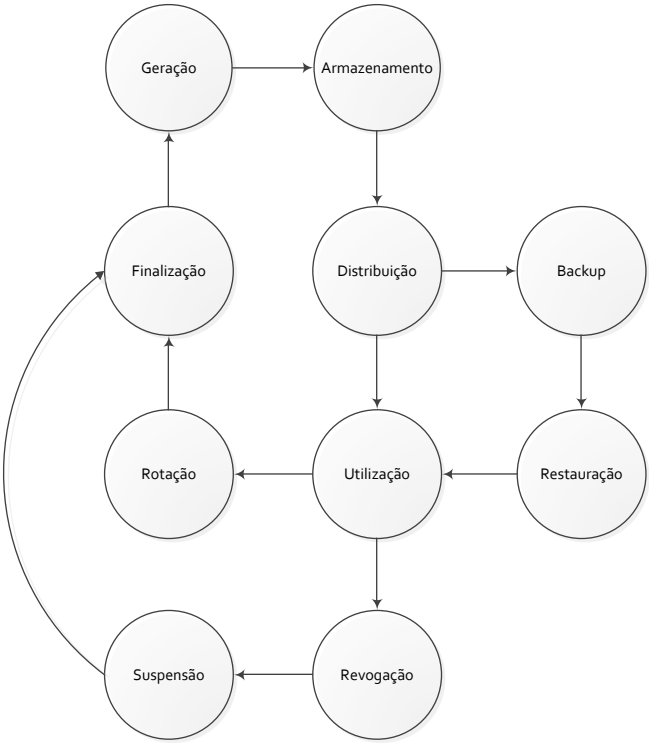

Figura 2.4: Modelo de ciclo de vida de chaves criptográficas comumente encontrado em módulos criptográficos.

2.3.4 Geração

É necessário garantir que as chaves estejam em ambiente seguro desde sua geração, além de haver uma grande preocupação com sua qualidade, devendo estas ser imprevisíveis, ou seja o mais aleatórias possível.

2.3.5 Armazenamento

Após sua geração, uma chave criptográfica deve ser armazenada de forma segura, com o intuito de evitar que seu conteúdo torne-se público. Quanto mais valiosa a chave em questão, mais sofisticados devem ser os dispositivos de segurança que a protegem.

Tais dispositivos vão desde um simples controle de acesso baseado em permissões, do uso de criptografia para cifragem das chaves, até um dispositivo dedicado a gerenciar tais artefatos, repleto de artifícios físicos e lógicos. Estes dispositivos podem, por sua vez, encontrarem-se resguardados em locais rigidamente controlados, conhecidos como salas-cofre.

2.3.6 Distribuição

Chaves em geral são de propriedade de uma ou um conjunto de entidades, frequentemente chamadas de custodiantes. Os custodiantes são os usuários autorizados a fazer uso de uma ou mais chaves, cabendo a eles a gerência (custódia) destes materiais.

As chaves criptográficas devem estar sempre disponíveis aos seus custodiantes, mediante autenticação. Essa garantia de disponibilidade implica em controlar o acesso a estes materiais, através da identificação dos usuários, provendo-lhes ainda um ambiente seguro para a utilização dos artefatos gerenciados.

2.3.7 Backup

Apesar de toda a preocupação com o controle sobre chaves criptográficas, todo o sistema computacional está sujeito a falhas, sejam elas físicas ou lógicas. Para evitar que uma indisponibilidade no dispositivo criptográfico ocasione a perda definitiva dos dados protegidos por suas chaves gerenciadas, frequentemente são utilizadas técnicas de cópia segura do material. Dessa forma, a chave gerenciada é transportada para outro ambiente, com nível de segurança semelhante ao original.

Um sistema gerenciador de chaves deverá garantir que cópias de chaves somente deixem o ambiente protegido em que se encontram na forma cifrada e sejam restauradas apenas em ambiente com nível de segurança igual ou superior ao de origem. Ainda, o procedimento de backup deve manter todas as cópias destes materiais sensíveis sob a vigilância de seus custodiantes, provendo assim formas de gerência de múltiplas cópias do material.

2.3.8 Restauração

Não faz sentido algum manter tantos cuidados com chaves criptográficas se suas cópias puderem ser restauradas em qualquer ambiente, sem quaisquer garantias de segurança. Por este motivo, a restauração de cópias de segurança deve se dar somente em ambientes de segurança igual ou superior ao de origem da chave.

Para garantir estes requisitos, é importante que apenas entidades responsáveis por sua custódia possam restaurar um ambiente operacional, a partir de uma cópia. Ainda, procedimentos de auditoria são fundamentais para a segurança do processo, bem como futuro rastreamento de possíveis incidentes.

2.3.9 Utilização

Chaves criptográficas devem ser atribuídas a propósitos específicos. À medida que uma chave é utilizada, torna-se cada vez mais atrativa

aos olhos de um atacante. A utilização de uma chave em múltiplos processos pode levar ao enfraquecimento destes. Um único ataque bem sucedido a um dos serviços, levará ao comprometimento de todos.

A utilização das chaves criptográficas deve ser monitorada e restrita àqueles que detêm seu controle. A estes, deve-se zelar pela disponibilidade dos serviços. Aos que não o têm, deve-se negar o acesso, bem como tomar contra medidas, em possíveis tentativas de violação dos direitos de acesso. Exemplos de contra medidas são o registro do ocorrido, tal como a destruição do material, com vista a preservar sua confidencialidade. Dessa forma, além de armazenada, uma chave deverá também ser utilizada em ambiente seguro e controlado.

2.3.10 Revogação

Durante sua vida útil, uma chave pode ser comprometida. Os motivos para o comprometimento de uma chave vão desde sua perda, roubo, até a quebra do algoritmo em que é utilizada.

Para que sejam evitados maiores prejuízos frente a essas situações, são necessários mecanismos de revogação capazes de identificá-las. Ao sinal de violação, deve-se existir formas de tornar público o comprometimento do material.

2.3.11 Suspensão

Uma vez comprometida, é importante que uma chave não mais seja utilizada para proteger ou mesmo para verificar informações. Assim, com sua revogação, uma chave deverá também ter seus serviços suspensos, ficando indisponível para uso. É importante que este processo seja efetuado apenas por entidades autorizadas, como custodiantes ou auditores.

2.3.12 Rotação

À medida que a capacidade computacional aumenta, algoritmos, outrora inquebráveis, tornam-se inseguros e devem ser periodicamente substituídos (LABORATORIES, de 2010). Essa realidade já implica em um prazo de validade para as chaves criptográficas, mas é agravado pelo tamanho de chave utilizado.

O prazo de validade de um algoritmo ideal, isto é, para o qual não existam falhas de segurança, será baseado em uma estimativa de tempo e esforço necessários para um ataque de força bruta ter sucesso (FERGUSON; SCHNEIER, 2003). Quanto menor a chave utilizada, menor o número de tentativas necessárias até encontrá-la. Contudo, o uso de chaves muito grandes torna os algoritmos mais lentos e inviáveis de serem aplicados a determinadas situações. O resultado são atualizações no tamanho das chaves ainda mais frequentes do que as do algoritmo criptográfico em si, o que pode ser confirmado na figura 2.1. A essa troca de chaves dá-se

o nome de *Rotação*.

O processo de rotação é bastante importante, pois quanto mais informação foi protegida por determinada chave, maior seu valor para uma terceira parte que obtiver acesso a ela, o que pode aumentar o número de tentativas de acesso não autorizado, bem como em ataques mais sofisticados e com maior chance de sucesso.

2.3.13 Finalização

A finalização de chaves criptográficas, tanto pela conclusão de seu ciclo quanto por uma possível revogação, deverá garantir que nenhum vestígio desta possa ser recuperado. Para tanto, utiliza estratégias de destruição segura, baseada em criptografia, bem como sobrescritas à memória antes ocupada pela chave destruída.

Apresentados os principais modelos de gestão do ciclo de vida de chaves, existentes na literatura, serão abordadas, na próxima seção, as garantias de segurança necessárias às chaves criptográficas para que o procedimento criptográfico surta o efeito esperado.

2.4 GARANTIAS DE PROTEÇÃO

Para proteção de chaves e parâmetros críticos de segurança, podemos citar as seguintes garantias como requisitos necessários:

Autenticidade: O processo de distribuição de chaves deve ser capaz de identificar a fonte de distribuição, de forma a garantir a autenticidade da chave gerada. Um exemplo de garantia de autenticidade é o uso de certificados digitais para a associação de um par de chaves a uma entidade;

Confidencialidade: Para chaves que precisam ser mantidas em segredo, como a parte privada das assimétricas, é necessário garantir a propriedade de confidencialidade, visando que seu conteúdo jamais venha a público;

Integridade: Um princípio necessário tanto a informações criptografadas quanto às chaves, é o da integridade. É necessário prover garantias de que estas não sofreram alteração e, caso tenham sofrido, recuperar os valores originais;

Disponibilidade: Embora sejam artefatos extremamente delicados, as chaves devem estar disponíveis sempre que requisitadas por aqueles que têm direito de acesso;

Controle sobre Acesso: Com a alta disponibilidade necessária às chaves criptográficas, surge também uma preocupação de que o acesso a este material seja feito somente por entidades devidamente credenciadas e autorizadas;

Registro de Utilização: O controle de uso das chaves criptográficas também deve ser devidamente garantido de forma a permitir a identificação precisa de cada momento em que o artefato foi utilizado, destruído, copiado, entre outros;

Rastreabilidade: Como forma de garantir sua disponibilidade, cópias da chave devem ser distribuídas em locais distintos, com segurança controlada. Contudo, é necessário garantir que estas cópias sejam devidamente registradas e possam ser rastreadas sempre que necessário;

Unicidade: É uma característica desejável das chaves assimétricas privadas em particular, tendo em vista que a chave identifica inequivocamente uma entidade. Uma vez que uma chave privada é o objeto responsável por garantir a irretratabilidade, perante o procedimento de assinatura digital, devem existir garantias de que ela seja única, estando sempre sob o controle de seu titular;

Tempestividade: Chaves criptográficas, bem como os algoritmos a que se destinam, têm um tempo de vida finito, onde precisarão ser substituídos por outras soluções ou, no caso das chaves, por versões mais fortes. Assim, é necessário garantir que determinada chave seja atualizada sempre que seu nível de proteção torne-se inadequado ao tipo de material protegido.

As garantias de proteção necessárias a uma chave criptográfica estão diretamente relacionadas à sua finalidade, ou seja, ao uso a que a chave se destina. Exemplificando, algumas garantias necessárias a uma chave pública são diferentes das requeridas por uma chave privada. A mais evidente distinção é a garantia de *confidencialidade*, desnecessária a uma chave pública, mas fundamental a uma chave privada.

Como o foco principal deste trabalho incidirá sobre chaves assimétricas, as propriedades aplicáveis a cada uma de suas componentes, são apresentadas na tabela 2.2. Em (BARKER et al., 2007), são listadas as propriedades necessárias a outros tipos de chaves e informações sensíveis.

Apesar de bem definidas as propriedades necessárias a uma chave criptográfica, garantir tais propriedades não é uma tarefa trivial, tendo em vista que, ao mesmo tempo em que a chave deve ser mantida sob rígido controle, também precisa estar sempre disponível a quem tem o direito de acesso. A próxima seção apresenta os principais desafios ainda em aberto com relação à gestão de chaves.

2.5 DESAFIOS EM GERÊNCIA DE CHAVES CRIPTOGRÁFICAS

Conforme exposto, existem na literatura distintos modelos de gestão de chaves criptográficas. Apesar de contemplarem um subconjunto de

Privada	Autenticidade
	Integridade
	Confidencialidade
	Controle sobre Acesso
	Registro de Utilização
	Rastreabilidade
	Disponibilidade
	Unicidade
	Tempestividade
Pública	Autenticidade
	Integridade
	Disponibilidade
	Unicidade
	Tempestividade

Tabela 2.2: Garantias perseguidas na gestão de chaves assimétricas (BARKER et al., 2007).

estados em comum, como os de *Geração*, *Destruição* e *Utilização*, tais modelos, em geral, divergirem nas formas de gerência propostas. Falta aos modelos, além de padronização, uma maior preocupação com o procedimento de replicação de chaves, uma vez que, quando presentes, os procedimentos de backup reduzem-se a simples cópias do material.

Com a necessidade de replicação, surge um dos principais problemas para a gestão segura do ciclo de vida de chaves: o rastreamento das múltiplas cópias da mesma chave.

Quando uma nova instância da chave passa a existir, deve-se estabelecer um rastro entre a original e sua cópia, evidenciando que um novo exemplar do material existe. Tal realidade vem à tona no momento da destruição de uma chave replicada. Destruir uma das cópias não é suficiente para garantir sua transição para o estado *Finalizada*. Se pelo menos uma das instâncias dessa chave continuar ativa, sem a ciência de seus custodiantes, a destruição da cópias terá um efeito oposto ao pretendido. A presença de uma cópia oculta implicará na falsa sensação de segurança de que a chave não mais existe. A falta de controle sobre as cópias de uma chave poderá permitir que esta caia em mãos erradas, acarretando inúmeros prejuízos.

2.6 CONSIDERAÇÕES DO CAPÍTULO

Este capítulo apresentou os principais modelos de gestão de chaves, presentes na literatura. Pôde-se então perceber a falta de padronização entre as diferentes propostas, que por vezes mostram-se incompletas, demonstrando que ainda há necessidade de estudos nesse campo.

Um dos pontos mais críticos dos modelos abordados, diz respeito ao procedimento de backup das chaves criptográficas, que não mais contemplam as necessidades de uma aplicação atual, onde há necessidade de atendimento à alta demanda.

Com o uso de criptografia assimétrica em aplicações que demandam alto desempenho, a situação torna-se crítica, pois a cópia inadvertida de uma chave privada prejudica uma propriedade importante da chave privada: sua unicidade.

Não ter a certeza acerca de qual instância da chave foi utilizada, pode provocar sérios problemas e comprometer a irretratabilidade, principal propriedade da assinatura digital, como será mostrado na seção 5.

Para melhor entender as necessidades sobre as chaves assimétricas, no ambiente de uma infraestrutura de chaves públicas, o capítulo 3 apresenta uma breve revisão acerca dos principais conceitos, bem como os serviços de uma AC que podem fazer uso dessa alta disponibilidade.

3 INFRAESTRUTURA DE CHAVES PÚBLICAS

3.1 INTRODUÇÃO

Conforme exposto no capítulo 1, para que uma chave possa ser utilizada, esta deve estar ligada a uma entidade. Dentre as várias formas de se estabelecer essa ligação (BISHOP, 2002), encontram-se as infraestruturas de chaves públicas, baseadas em uma cadeia de confiança hierárquica.

O modelo assemelha-se à realidade encontrada em inúmeras ocasiões do cotidiano. Quando se fala em estrutura hierárquica, logo pensa-se a hierarquia militar, com suas patentes e divisões. Contudo, modelo organizacional da grande maioria das empresas é também um exemplo desse tipo de estrutura, assim como a própria estrutura familiar. A ampla aplicabilidade das ICPs contribuiu para a difusão do modelo, sendo hoje empregadas em variadas aplicações, tal qual sistemas de autenticação e controle de acesso, transações e processos eletrônicos diversos, entre outros.

Neste capítulo, serão apresentados os principais conceitos relacionados a tais estruturas. A seção 3.2 apresenta uma visão geral das ICPs. A seção 3.3 detalha os principais componentes da infraestrutura. Na seção 3.4, serão abordadas as principais cerimônias envolvidas no dia-a-dia das autoridades certificadoras de uma ICP.

3.2 VISÃO GERAL

O princípio básico de uma infraestrutura de chaves públicas é a existência de uma autoridade máxima, responsável por certificar todas as demais, tornando-se assim um ponto único de confiança. Essa entidade no topo da infraestrutura, conhecida como Autoridade Raiz, pode ainda delegar a tarefa de certificação a outras autoridades, sendo estas suas subordinadas. O resultado dessa configuração é uma estrutura hierárquica de autoridades.

Para o estabelecimento dessa árvore de confiança, as autoridades emitem atestados, garantindo que determinada chave pertence a uma de suas subordinadas. Por intermédio de uma assinatura digital sobre o atestado, comumente conhecido como certificado digital, a autoridade superior responsabiliza-se pela autenticidade das informações contidas no certificado emitido. O certificado digital da entidade, em conjunto com sua chave de assinatura, compõem a identidade digital da mesma. Como ponto máximo de confiança, a própria AC Raiz assina seu certificado, emitindo assim seu próprio certificado auto assinado.

A entidade do nível mais baixo da estrutura hierárquica é conhecida como entidade final. Confiando na raiz da hierarquia, uma entidade deve procurar estabelecer um caminho de confiança até a raiz, sempre que

necessário verificar a identidade digital de uma segunda. Ao caminho formado pelos referidos certificados, dá-se o nome de caminho ou cadeia de certificação.

A figura 3.1 ilustra, em alto nível, uma ICP contendo dois níveis de autoridades certificadoras subordinadas.

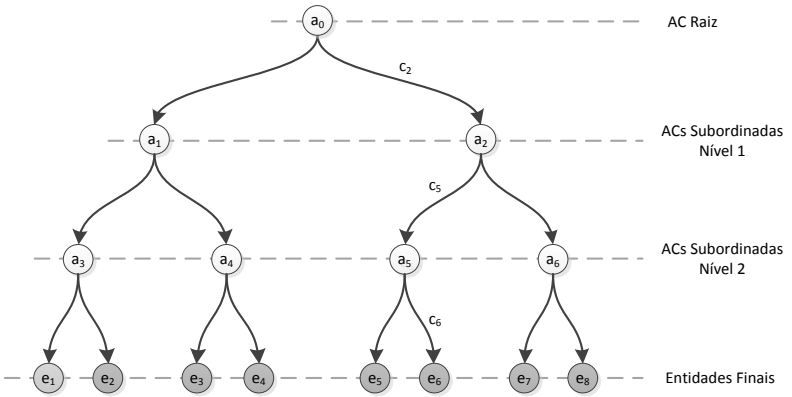

Figura 3.1: Modelo tradicional de uma infraestrutura de chaves públicas.

A figura 3.1 destaca os diferentes níveis de entidades da ICP. No primeiro nível, as ACs a_1 e a_2 representam ACs intermediárias, pois estão entre dois níveis de ACs. Suas subordinadas, contudo, emitem certificados para entidades finais, sendo então conhecidas como ACs finais. No último nível da ICP, nos círculos destacados, pode-se notar as entidades finais.

A figura mostra ainda as conexões entre as entidades, ressaltando o caminho de certificação formado pelos certificados c_2, c_5 e c_6.

Para promover a interoperabilidade e a maior aplicação do modelo, O documentoITU-T (2005) definiu um conjunto de recomendações que acabaram por nortear as distintas implementações de infraestruturas de chaves públicas. A primeira versão do documento que descreve o padrão, conhecido como X.509, data de 1988. Anos mais tarde, em 1995, formou-se dentre a comunidade de pesquisa que rege os padrões e tecnologias envolvidos no uso da Internet, a IETF, do Inglês *Internet Engineering Task Force*, um novo grupo de trabalho para estudar e promover o uso da tecnologia na rede mundial de computadores. Hoje os avanços do grupo de trabalho encontram-se publicados em diversos documentos técnicos, conhecidos como RFC [1] que dividem os inúmeros conceitos relativos a ICPs.

As demais seções deste capítulo fazem um apanhado geral da documentação técnica existente na literatura, descrevendo seus principais

[1] Do inglês *Request for Comment*, é a forma como a IETF publica e valida seus trabalhos.

componentes, seu funcionamento, além dos principais serviços e aplicações da tecnologia.

3.3 COMPONENTES

Uma ICP é composta essencialmente por soluções tecnológicas, políticas, protocolos, mas principalmente por pessoas, responsáveis pela gestão e uso de seus serviços e aplicações. A seguir serão apresentados os principais componentes dessa estrutura.

3.3.1 Certificados Digitais

Certificado digital é um objeto eletrônico capaz de identificar uma entidade através de seus dados, sua chave pública e a assinatura de uma terceira parte confiável. Para tanto, agrega funcionalidades de dois objetos reais bastante comuns, o cartão de crédito e o cartão de visitas (HOUSLEY; POLK, 2001).

Em um cartão de visitas estão todas as informações necessárias à identificação de uma entidade. Nele, geralmente, estão contidas informações como: nome, telefone, empresa onde trabalha e função exercida (no caso de um cartão profissional), endereço de correio eletrônico, entre outras informações.

Todavia, nada garante que em um primeiro momento, duas pessoas que acabaram de se conhecer possam confiar nos cartões uma da outra. Uma pessoa mal intencionada pode forjar um cartão, com os dados de uma segunda, com o intuito de obter vantagens e direitos pertencentes a esta. Para evitar esse tipo de problema, o certificado digital agrega também funcionalidades presentes em um cartão de crédito.

O cartão de crédito convencional tem propriedades que complementam um simples cartão de visitas. Um cartão de crédito exige maior esforço para ser falsificado, possuindo os dados de seu titular em alto relevo, hologramas para dificultar a clonagem, entre outros. A grande maioria dos cartões de crédito hoje dispõem inclusive de um processador criptográfico, que torna sua clonagem praticamente inviável. Além dessas informações, o cartão ainda conta com uma data de validade, buscando assegurar a atualidade das demais informações.

Esses dados presentes em um cartão de crédito são assegurados por uma terceira parte confiável, a empresa operadora do cartão, responsável por sua emissão e gerenciamento. Outra característica fundamental do cartão de crédito é a existência de formas de revogá-lo, em casos de perda, roubo ou extravio.

Todas as características supramencionadas também estão presentes em um certificado digital. Para Housley e Polk (2001), um certificado digital ideal deve possuir as seguintes propriedades:

- deve ser totalmente digital para facilitar a sua distribuição e processamento no meio virtual;

- deve conter as informações necessárias para se identificar seu dono (portador da chave privada);
- deve ser fácil de determinar a validade do mesmo, ou seja, deve possuir a data de emissão e validade;
- deve ter sido criado por uma terceira parte confiável e não pelo próprio dono do certificado;
- deve ser fácil de diferenciar dois ou mais certificados de uma mesma identidade, tendo em vista que não há restrição de número de certificados por entidade;
- deve ser fácil de identificar uma tentativa de adulteração de um certificado;
- deve ser possível verificar agilmente se as informações contidas no certificado são atuais;
- não deve ser possível alterar seu conteúdo;
- deve discriminar o tipo de aplicação para o qual foi designado.

A figura 3.2 ilustra as principais informações presentes em um certificado digital.

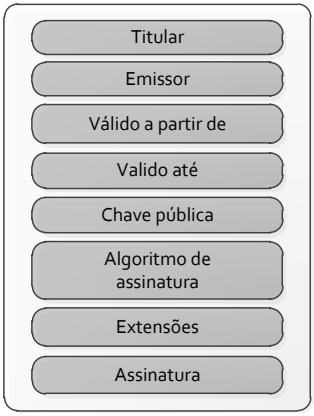

Figura 3.2: Composição básica de um certificado digital X.509 em sua terceira versão.

Conforme apresentado na figura 3.2, um certificado digital deve ter um campo *Titular*, designando e entidade detentora da chave privada correspondente. Como o certificado é digitalmente assinado por uma entidade emissora, ou seja, por uma Autoridade Certificadora, esta deve estar

identificada, através de seu nome, no campo *Emissor*. Uma das propriedades mais importantes do certificado é sua validade. A validade de um certificado é dada pelo período de tempo compreendido entre as datas *Válido a partir de* e *Válido até*. A chave pública ligada ao titular também compõe o certificado, assim como o algoritmo de assinatura utilizado para assiná-lo. Por fim, encontram-se as extensões do certificado, bem como sua assinatura digital. As extensões passaram a fazer parte do certificado digital em sua terceira versão, consistindo em uma forma de flexibilizá-lo. Através de extensões é possível definir quais os usos de uma chave, qual a profundidade máxima de um caminho de certificação, bem como otimizar a montagem do caminho de certificação. Esses são só alguns dos muitos empregos das extensões nos certificados.

3.3.2 Autoridades Certificadoras

A autoridade certificadora, em uma ICP, é a entidade responsável por gerenciar o ciclo de vida de certificados digitais. Dentre as tarefas envolvidas na gestão de certificados, destacam-se:

Emissão de Certificados: Função primordial das autoridades certificadoras, a emissão de certificados a entidades, bem como autoridades subordinadas, forma a árvore de confiança que caracteriza uma ICP;

Revogação de Certificados: Mediante comprovação do comprometimento da chave privada de uma entidade subordinada, é papel da AC providenciar sua revogação de seu certificado;

Renovação de Certificados: Como certificados têm períodos pré determinados de validade, uma AC pode oferecer o serviço de renovação, emitindo um novo certificado para a mesma chave pública;

Emissão de Listas de Certificados Revogados: Como forma de tornar pública a relação dos certificados revogados, a AC emite periodicamente uma lista contendo os números seriais de cada certificado revogado;

Emissão de Respostas OCSP: OCSP consiste em um protocolo para checagem do status atual de um certificado em tempo real. Através dele, uma AC poderá prover um serviço de consulta a estados dos certificados, emitindo um recibo de validade assinado;

Manutenção de Histórico: Uma autoridade certificadora deve manter registros de todos os certificados emitidos, bem como das revogações, para casos de futuras disputas.

Esse conjunto de atribuições fazem da autoridade certificadora uma entidade de suma importância para uma infraestrutura de chaves públicas. Com tamanha importância, a chave privada dessa entidade deve ser rigidamente controlada. Em caso de uma autoridade certificadora raiz, a

segurança é ainda mais crítica, pois a confiança de toda a hierarquia depende de sua integridade.

Dado o enorme valor da chave privada de uma AC, é comum que tais autoridades sejam gerenciadas em ambientes conhecidos com salas cofre, contando com rígido controle de acesso e contando com dispositivos sofisticados para prevenção de catástrofes (LUZ, 2008).

É igualmente comum, como forma de garantir a continuidade dos serviços de uma autoridade certificadora raiz, a replicação completa desse ambiente. Isso significa manter salas cofre em locais geograficamente distintos, onde seja possível operar a mesma autoridade de forma independente. Uma segunda consequência desses múltiplos ambientes é que a chave da AC, em algum momento, precisa ser copiada e transportada de um ambiente para outro.

Outra preocupação que implica em replicação da chave privada de uma AC é o atendimento a altas demandas por assinaturas. Em ICPs de grande porte, onde as ACs emitam respostas a requisições OCSP, por exemplo, pode ser necessário um balanceamento de carga entre mais de um módulo criptográfico, para garantir a disponibilidade do serviço.

3.3.3 Lista de Certificados Revogados

Listas de Certificados Revogados, ou LCRs, são componentes de uma ICP que visam garantir a atualidade das informações constantes em um certificado digital. Uma LCR contém o registro de todos os certificados cujo período de validade ainda esteja válido, revogados por uma AC. Essa lista é assinada pela AC e publicada, de forma que uma entidade buscando conferir a validade de um determinado certificado possa obtê-la, verificando o status de revogação do mesmo. A lista contém informações como serial, data de revogação e opcionalmente um conjunto de extensões com informações adicionais de cada certificado revogado. Dentre as extensões existentes, vale ressaltar a *Reason Code*, contendo um código que descreve a razão da revogação do certificado (COOPER et al., 2008).

Para garantir a atualidade das informações, a LCR deve ser atualizada periodicamente, mesmo que não tenha havido nenhuma nova revogação. Para garantir que um usuário esteja utilizando uma LCR atual, esta possui dois atributos: *thisUpdate* e *nextUpdate*. O primeiro campo informa a data de emissão da LCR, enquanto o segundo limita sua validade. Uma vez que a LCR tenha expirado, uma nova LCR deverá ser obtida no(s) repositório(s) apontado(s) pela extensão *CRLDistributionPoints* do certificado digital em questão.

3.3.4 Requisição de Certificado

A requisição de um certificado digital envolve basicamente cinco fases. Na primeira, um par de chaves é gerado em um dispositivo controlado pela entidade, como um cartão inteligente, um computador pessoal

ou mesmo um módulo criptográfico.

A segunda fase diz respeito à emissão da requisição em si, contendo os dados que farão parte do certificado, além da parte pública do par de chaves gerado. Após o preenchimento da requisição, esta é assinada pela chave privada do requisitante.

Na terceira fase da emissão do certificado, ocorre a validação dos dados contidos na requisição. Essa validação depende das práticas definidas para a AC, que geralmente variam de acordo com a finalidade do certificado. Em certificados utilizados para assinatura de correio eletrônico, por exemplo, a autenticação pode ser feita remotamente, através de um protocolo desafio resposta, enviado para o endereço informado na requisição. Em casos onde uma validação presencial é necessária, esta tarefa geralmente é delegada a uma segunda entidade, a Autoridade de Registro (AR). A RFC 2511 define um formato para mensagens trocadas entre ARs e ACs, contendo requisições por certificado (MYERS et al., 1999).

A quarta fase consiste na emissão do certificado em si, bem como sua publicação para que possa ser obtido pela entidade requisitante. Por fim, a quinta fase consiste na instalação do certificado emitido junto a chave privada gerada na primeira etapa, compondo dessa forma a identidade digital da entidade. A RFC 2986 detalha a sintaxe de uma Requisição de Certificado (NYSTROM; KALISKI, 2000).

3.3.5 Política de Certificação (PC)

As políticas de certificação definem um conjunto de regras que regem o funcionamento de uma ICP e devem ser seguidas por todas as entidades credenciadas. Dentre as questões abordadas em uma política de certificação, podemos citar o protocolo de segurança que cada AC deverá cumprir na emissão de certificados, a validade padrão para os mesmos, bem como as extensões nos certificados digitais. Portanto, é através da análise dos documentos de políticas que terceiras partes decidem se um certificado é confiável e aplicável ao uso pretendido.

Como mencionado, as políticas são definidas em documentos: Políticas de Certificação (PC) e Declaração de Práticas de Certificação (DPC). A PC é um documento de alto nível em que se definem um conjunto de regras a serem seguidas para manutenção da segurança da infraestrutura. Descrevem-se, dessa maneira, a operação de uma AC, além das responsabilidades e obrigações de cada uma das componentes da hierarquia.

Cabe à Declaração de Práticas de Cerficação – documento altamente detalhado – descrever a forma como cada uma das ACs de uma ICP implementa as regras definidas na PC.

Uma PC, por exemplo, pode definir que é necessária a validação presencial dos dados de uma entidade, no momento da emissão da requisição de certificado. A DPC de uma determinada AC descreveria, então, a forma como essa validação seria feita, quais os documentos validados, entre outras disposições.

Portanto, a DPC é o documento que deve ser analisado por auditores para validarem as operações de uma AC. Para validar uma DPC, deve-se garantir que os requisitos da PC estejam sendo devidamente satisfeitos. A PC deve ser genérica e seu uso estimado por vários anos, enquanto a DPC deve ser direcionada para cada AC, sendo bem mais específica. Por último, a PC deve ser públicada, sendo que o mesmo não é necessário para a DPC.

3.4 CERIMÔNIAS EM AUTORIDADES CERTIFICADORAS

Ellison (2007) define o conceito de cerimônias, como uma extensão dos protocolos de comunicação englobando um novo tipo de entidade dentre as interações descritas: os usuários. No projeto e análise dos protocolos de comunicação, apenas são levadas em consideração as trocas de mensagens entre os componentes computacionais, o que pode levar a um modelo falho e incompleto, que trata o usuário como uma entidade robotizada e por vezes não apresenta na prática o efeito pretendido durante seu projeto.

Em cerimônias, segundo Ellison, todas as mensagens devem ser levadas em consideração, inclusive as trocadas entre usuários de outras formas que não por mensagens digitais (telefonemas, conversa cara a cara, entre outros). A partir deste modelo, pode-se então tentar buscar as possíveis falhas e deficiências que podem favorecer uma terceira parte mal intencionada, que pode atacar tanto o protocolo como um usuário com o intuito de obter vantagens.

Analisar essas falhas em cerimônias não é uma tarefa fácil. Mapear a reação de um ser humano ao se deparar com determinada situação é, alem de extremamente complexa, uma tarefa que envolve um trabalho conjunto dentre diversas áreas do conhecimento humano, como psicologia, neurociência, ciência da computação, entre outras. Ao analisar uma cerimônia, um analista deverá buscar uma resposta à seguinte pergunta: "Qual a probabilidade de um atacante enganar o ser humano a tomar uma decisão incorreta?"

Cerimônias, no contexto de Autoridades Certificadoras, correspondem à aplicação dos procedimentos presentes na declaração de práticas de certificação, de acordo com a política de certificação. Essa execução é realizada pelos gestores da AC, que interagem com os componentes tecnológicos.

As principais cerimônias existentes no ambiente de gestão de uma autoridade certificadora são:

Geração do Par de Chaves: Esta é a cerimônia inicial de uma autoridade certificadora. Essa cerimônia envolve a preparação da plataforma criptográfica, bem como a atribuição da custódia da chave privada ao grupo de custodiantes;

Emissão do Certificado Raiz: Essa cerimônia marca o nascimento da au-

toridade certificadora raiz propriamente dita, pois sua validade passa a contar a partir da data constante em seu certificado;

Emissão de Requisição de Certificado: Para autoridades certificadoras intermediárias, uma requisição de certificado deve ser gerada, para posterior assinatura, por parte da AC Raiz;

Backup da Plataforma Criptográfica: É comum que, após estabelecida a autoridade certificadora, seja feito um backup de segurança. No caso da chave privada, esta pode ser armazenada de forma segura ou replicada em múltiplos ambientes;

Restauração do Backup: Em caso de problemas técnicos com o sistema em produção, desde que não haja qualquer indício de comprometimento da chave privada da AC, uma cerimônia de restauração de backup deve ser executada. Ao fim da cerimônia, os serviços da autoridade devem ser restabelecidos em sua totalidade;

Emissão de Certificados: A cerimônia de emissão de certificados, para uma AC subordinada ou para uma entidade final, consiste na importação de uma requisição de certificado, bem como sua assinatura, com a chave privada da AC emissora. A emissão de certificados digitais para entidades finais é o serviço primordial fornecido por uma ICP;

Emissão de Listas de Certificados Revogados: A cerimônia de emissão de LCR é a mais tradicional entre as ACs. Uma AC Raiz, por exemplo, pode ficar meses ou até mesmo anos sem emitir um certificado, mas emitirá LCRs periodicamente. Esses períodos, em geral, variam de minutos a alguns meses, dependendo da atividade da AC;

Arquivamento e Destruição de Chaves Privadas: Uma cerimônia importante é a de destruição de chaves privada de uma AC. Uma chave privada pode ser destruída por vários motivos. Um deles está relacionado à política das ACs, que seguindo as boas práticas de gerência de chaves, atualizam o material a cada renovação de certificado. Outro motivo para a destruição da chave de uma AC é a obsolescência de seu módulo criptográfico. Nesse caso, é importante que haja garantias de que o material realmente foi destruído.

Auditoria: A cerimônia que complementa as atividades gerenciais de uma AC é a auditoria. Fundamental para a segurança de uma AC, a auditoria visa garantir que as políticas de segurança e certificação da AC estão sendo seguidas por seus gestores. Os auditores, em caso de constatação de irregularidades, têm o poder de bloquear as atividades da AC.

3.5 CONSIDERAÇÕES DO CAPÍTULO

Este capítulo apresentou os principais componentes de uma ICP, bem como as características gerais de cada um. Ainda, foram sumarizadas as principais atividades, bem como os serviços providos pelas autoridades certificadoras de uma ICP. Dada a arquitetura apresentada, além dos serviços críticos providos por tais autoridades, deve-se zelar pelas chaves privadas de cada AC, como forma de garantir a segurança provida aos usuários da tecnologia.

Conforme exposto, entre as cerimônias de uma AC estão as de cópia e restauração de uma chave, que tradicionalmente implicam em cópia da chave privada da autoridade. Nessas cópias, em geral, a rastreabilidade das múltiplas instâncias fica a cargo do perfil de auditoria, baseando-se em registros de uso dos múltiplos ambientes criptográficos. Dado ser uma cerimônia altamente dependente do fator humano, não dispondo assim dos formalismos providos por um simples protocolo, a tarefa de auditoria é árdua e passível de erros. O capítulo 5 apresentará os problemas trazidos por essa forma de rastreamento. O capítulo 4, a seguir, discorre acerca dos dispositivos criptográficos, utilizados para proteger a chave das ACs.

4 DISPOSITIVOS CRIPTOGRÁFICOS

4.1 INTRODUÇÃO

Conforme apresentado no capítulo 2, chaves criptográficas são a base fundamental dos principais algoritmos criptográficos existentes, provedores de segurança no ambiente digital. Apesar da existência de soluções que visam à proteção de tais artefatos utilizando-se apenas de procedimentos criptográficos (RSA, 1999b), nenhuma delas mostra-se efetiva à medida que o valor das chaves aumentam ao ponto de demandarem rígido controle de seu ciclo de vida.

Uma evidência desse caso é o armazenamento de uma chave em mídia convencional, sem qualquer proteção física. Por mais completa que seja a proteção provida pela criptografia aplicada na proteção da chave, o acesso a ela será controlado pelo sistema operacional em execução, sendo assim passível de cópia, destruição ou até mesmo substituição, mediante a exploração de uma das inúmeras falhas de segurança inerentes a tais sistemas.

Para o armazenamento seguro de materiais sensíveis, fez-se necessário um dispositivo dedicado exclusivamente à gerenciar o ciclo de vida de chaves. Esses dispositivos são conhecidos como módulos de segurança criptográficos, ou MSCs.

Este capítulo apresentará, na seção 4.2, uma visão geral das soluções existentes para o processamento seguro de informações e gestão de chaves criptográficas. Nas seções 4.3 e 4.4 serão apresentadas ainda as principais normas e padrões envolvidos na regulamentação e interoperabilidade de dispositivos dessa natureza.

Na seção 4.5, será apresentado um ataque bem sucedido contra um MSC de mercado, implementado de acordo com os padrões definidos na seção 4.3 e 4.4. A exploração dessa vulnerabilidade, que permitiu a extração de uma chave privada em claro, mostra que a conformidade as normas não é garantia de que o MSC é inviolável, o que reforça a necessidade de uma gestão de chaves que possibilite a rastreabilidade das múltiplas instâncias.

4.2 VISÃO GERAL

Um módulo criptográfico consiste em dispositivo composto por hardware, software e firmware, dedicado à gestão do ciclo de vida de chaves criptográficas, em cada uma das fases apresentadas no capítulo 2. É comum a utilização de módulos criptográficos também para processamento de dados sensíveis, como comparação de senhas ou operações financeiras.

Para proteger os dados sigilosos armazenados em seu interior, um MSC deve possuir uma área devidamente monitorada e protegida, através de sensoriamento e controles físicos, bem como protocolos criptográficos, consistindo em controles lógicos. O objetivo principal dos MSCs é detectar, evidenciar e mitigar quaisquer tentativas de corte, perfuração, trituração, ou de qualquer outra natureza, que possam levar à exposição do material sensível armazenado em seu interior. Ao perímetro interno em que são aplicadas tais proteções, dá-se o nome de fronteira criptográfica.

A figura 4.1 mostra um esquema simplificado do funcionamento de um MSC, na execução de um serviço que utiliza uma chave armazenada em seu interior.

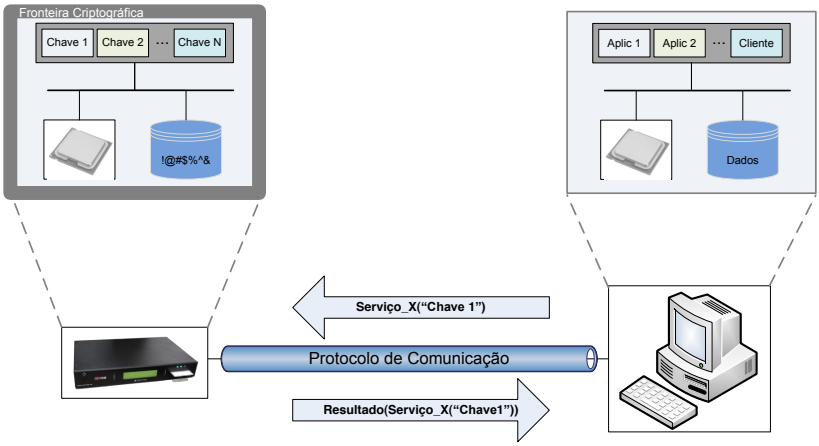

Figura 4.1: Funcionamento básico de um MSC

A figura 4.1 ilustra a principal diferença entre um MSC e um computador convencional. Um computador de propósito geral possui inúmeras aplicações de naturezas distintas, executando de forma concorrente, compartilhando assim uma mesma memória principal. Essa memória não conta com proteções contra acesso físico. Um MSC, por sua vez, encontra-se envolto por uma fronteira criptográfica. Sua memória persistente emprega criptografia para proteção dos dados, além de dispor de um conjunto limitado de aplicações, responsáveis pela gerência das chaves e demais parâmetros críticos.

Ainda de acordo com a figura, os serviços que vierem a fazer uso de uma das chaves gerenciadas, estarão implementados no interior do módulo, sendo lá executados, buscando garantir o sigilo do material ao mesmo tempo que sua disponibilidade . Para garantir a segurança da transação, o protocolo de comunicação emprega criptografia, estabelecendo um túnel seguro entre cliente e servidor. Um exemplo de protocolo com estas características é o SSL/TLS (DIERKS; RESCORLA, 2006).

Dessa maneira, uma chave criptográfica gerenciada pelo MSC ja-

mais deve deixar o perímetro criptográfico, ou seja, seu conteúdo em texto claro jamais deverá ser conhecido em ambiente desprotegido. Cabe ao módulo a capacidade de prover serviços que façam uso dos objetos gerenciados – os serviços criptográficos.

4.2.1 Arquitetura Básica

Não há uma especificação padronizada de como se construir um módulo de segurança criptográfico. Essa falta de padronização reflete-se nas diferentes arquiteturas e protocolos para gerência de chaves entre os módulos criptográficos encontrados no mercado. Cada fabricante desenvolve sua solução e mantém o projeto em segredo, com o intuito de proteger seu produto dos concorrentes e até mesmo evitar o conhecimento acerca dos dispositivos de segurança utilizados.

Há, contudo, padrões que visam manter o mínimo de compatibilidade entre as diferentes implementações, bem como fornecem diretrizes a serem seguidas no projeto e desenvolvimento de módulos criptográficos. Tais especificações visam classificar as diferentes soluções de acordo com o nível de segurança provido, servindo assim como atestados de conformidade com as boas práticas de projeto, bem como com a qualidade dos serviços prestados.

A figura 4.2 apresenta uma arquitetura genérica de um módulo de segurança criptográfico, com base nos requisitos apresentados pela norma FIPS 140-2 (NIST, 2002). A norma, elaborada pelo NIST em seu programa de certificação de hardwares criptográficos, é a mais aceita internacionalmente.

Segundo a figura 4.2, é possível observar que, circundando todos os componentes do módulo, encontra-se a fronteira criptográfica, composta por uma série de sensores, lacres, entre outros recursos, que buscam proteger a área interna ao MSC. Para o constante monitoramento do perímetro e o disparo das contra medidas necessárias à proteção dos artefatos protegidos, o MSC conta ainda com um subsistema de monitoria e registro de intrusões. Esse sistema, mediante qualquer tentativa de violação ou procedimento indevido, deverá proceder com os procedimentos de destruição de quaisquer vestígios que possam vir a comprometer a segurança dos materiais armazenados.

Na arquitetura apresentada, o MSC conta com uma área de memória persistente utilizada para armazenar o ambiente operacional, além de uma base de dados contendo os materiais sensíveis, tais quais as chaves criptográficas, dados de autenticação, entre outros parâmetros críticos de segurança (PCS).

Em geral, os MSCs possuem também um Gerador de Números Aleatórios (GNA), responsável por fornecer uma fonte de dados randômicos com qualidade comprovada, utilizados na geração de chaves secretas.

A principal aplicação componente do Ambiente Operacional é o provedor de serviços criptográficos, sistema responsável por processar as

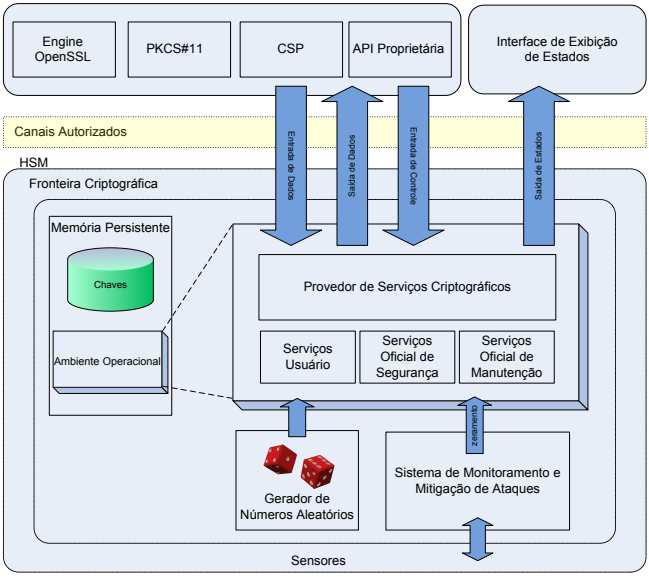

Figura 4.2: Arquitetura genérica de um MSC

requisições advindas dos clientes autorizados, autenticação dos papéis autorizados e posterior devolução dos resultados da operação.

Por fim, toda a interação entre usuários e módulo deverá ser realizada através de canais confiáveis de comunicação, utilizados para entrada e saída de dados e controles, além da exibição do estado interno em que se encontra o MSC.

As principais funcionalidades, bem como as proteções implementadas em módulos criptográficos em geral, são apresentadas na próxima seção, onde serão vistos também os demais critérios de avaliação de MSCs.

4.3 FIPS PUB 140

O documento *Security Requirements For Cryptographic Modules*, ou FIPS PUB 140 (NIST, 2002), como é comumente conhecido, faz parte de uma série de publicações do NIST que visam à padronização e gerencia de sistemas computacionais, tal como de telecomunicações, utilizados pelo governo federal norte-americano. A publicação orienta o projeto e desenvolvimento de módulos criptográficos através de um conjunto de requisitos, cobrindo as diferentes àreas envolvidas na construção de tais sistemas.

Através de um processo de validação, o *Cryptographic Module Validation Program (CMVP)*, laboratórios de análise credenciados ao instituto avaliam, sob demanda dos próprios fabricantes, os mais variados

tipos de hardwares ou até mesmo softwares criptográficos, desde bibliotecas, cartões inteligentes, até MSCs propriamente ditos. Alcançado sucesso no processo, o fabricante do dispositivo criptográfico recebe um certificado de conformidade com o padrão, que atribui a seu produto um determinado nível de segurança, de acordo com a eficácia comprovada nos sistemas de proteção implementados. São quatro os níveis de segurança, sendo que o último nível incorpora todos os requisitos dos níveis anteriores, em adição aos seus.

Este certificado é hoje aceito não só para uso em instituições federais norte americanas, mas também globalmente, tendo se tornado um dos padrões relacionados a dispositivos criptográficos mais aceitos no mundo. O documento encontra-se na segunda versão, daí o sufixo FIPS 140-2.

4.3.1 Especificação do Módulo Criptográfico

Os requisitos relacionados à especificação definem de forma geral como deverá ser composto um módulo criptográfico, bem como a documentação a ser apresentada, detalhando seu desenvolvimento. Nesta seção, um módulo é definido como o conjunto software, hardware e firmware, ou uma combinação destes, que implemente serviços criptográfico utilizando, no mínimo, um dos algoritmos aprovados pelo padrão.

O módulo deve ainda possuir uma fronteira criptográfica bem definida e apresentar documentação que especifique quais os componentes protegidos, detalhando suas características físicas.

A documentação deverá ainda conter:

- as portas físicas, logicas e interfaces de acesso ao módulo;
- os estados internos do módulo e suas interfaces de exibição;
- as funções de segurança aprovadas ou não, suportadas pelo MSC;
- diagramas e esquemas ilustrando os principais componentes e suas interconexões;
- esquemas em linguagens de alto nível para a descrição dos componentes de hardware, firmware e software;
- a política de segurança sob a qual opera o MSC.

4.3.2 Portas e Interfaces

O padrão restringe o acesso ao módulo por um conjunto de canais confiáveis, representados por interfaces físicas e lógicas. O fluxo de informações deve transitar pelas seguintes interfaces lógicas:

Entrada de Dados: através desta interface deverão entrar os dados necessários à execução dos serviços providos pelo módulo, como chaves, parâmetros críticos de segurança, dados de autenticação, entre outros;

Saída de Dados: dados como texto cifrado, assinado, chaves exportadas, backups, entre outros, deverão deixar o módulo criptográfico através da interface de saída de dados;

Entrada de Controle: interface dedicada à entrada de comandos de controle, seja para sua configuração, seja para solicitações de serviços criptográficos;

Exibição de Estados: a interface de exibição de estados é responsável pela externalização dos resultados obtidos após execução de determinado serviço bem como pela sinalização do estado interno atual do módulo;

Embora logicamente distintas, as interfaces acima descritas podem compartilhar uma mesma porta física. Com esta distinção entre o tipo de informação que entra e sai do módulo, torna-se possível e é requisitado que o canal de saída de dados seja desconectado durante operações sensíveis como geração, apagamento ou entrada de chaves.

4.3.3 Papéis, Serviços e Autenticação

O padrão FIPS 140 define dois papéis de acesso obrigatórios a todos os módulos criptográficos, com distinção entre o conjunto de serviços autorizados a cada um deles. Há ainda um conjunto de serviços em que não é necessária a autenticação de nenhum dos dois papéis, como exibição de estado, operações somente leitura de informações públicas, bem como outros serviços que não afetam os parâmetros críticos e chaves protegidas.

Os perfis obrigatórios aos MSCs são:

Perfil de Usuário: perfil responsável por serviços de segurança relacionados a uma ou mais chaves criptográficas, como configuração do uso de chaves em operações criptográficas, entre outros;

Perfil de Oficial de Segurança: perfil responsável por serviços de inicialização e gerenciamento, como configuração do módulo, auditoria e entrada/saída de chaves e PCSs.

Ainda segundo o padrão, caso o módulo permita serviços de manutenção que impliquem em acesso físico ou lógico a dados ou componentes protegidos, o módulo deve conter um perfil adicional nomeado Perfil de Manutenção. Ao autenticar um serviço de manutenção, todos os dados protegidos deverão ser zerados.

O módulo deverá especificar uma lista com todos os serviços disponíveis a cada um dos perfis existentes. As entradas de controle e dados que iniciam um serviço são chamadas de *Entradas de Serviço* enquanto os resultados e estados originados, são chamados de *Saídas de Serviço*. Cada *Entrada de Serviço* deverá resultar em uma *Saída de Serviço*.

4.3.4 Modelo de Estados Finitos

O padrão requer a descrição operacional dos módulos criptográficos através de um modelo de estados finitos representado por autômato ou tabela de transição de estados.

A documentação deverá conter todos os possíveis estados do módulo criptográfico, bem como as transições entre eles, especificando ainda as entradas originadoras destas, bem como as saídas provenientes da transição.

Dentre os estados obrigatórios do modelo, o padrão salienta:

Ligado/Desligado: distinguindo entre estados de alimentação primária, secundária ou de backup;

Estados do Oficial de Segurança: estados onde os serviços do oficial de segurança podem ser executados;

Estados de Entrada de Chaves/PCSs: estados para entrada de chaves e PCSs;

Estados do Usuário: estados onde os serviços disponíveis ao usuário poderão ser executados;

Estados de Autotestes: estados onde o módulo está executando os autotestes;

Estados de Erro: estados representado a ocorrência de erros impedindo a operação normal do módulo;

Estados de Manutenção: estados onde o módulo permitirá manutenção física ou lógica.

4.3.5 Segurança Física

A norma requer uma série de medidas preventivas de proteção dos dados sensíveis, contidos no interior do perímetro criptográfico, contra acesso não autorizado. Como garantias físicas, o documento especifica desde lacres, que evidenciem uma tentativa de violação, a bloqueios sofisticados baseados em sensoriamento.

Ainda, para módulos de nível de segurança mais alto, são necessárias também formas de mitigação de ataques através da destruição ou zeramento[1] das informações sensíveis.

As diferenças entre as proteções físicas exigidas para cada nível de segurança serão detalhadas na seção 4.3.12.

[1] Do inglês *zeroization*.

4.3.6 Ambiente Operacional

Os requisitos relacionados ao ambiente operacional do módulo criptográfico referem-se à gerência de softwares, firmwares ou mesmo hardwares necessários ao seu funcionamento. O sistema operacional utilizado é um importante componente do ambiente operacional. Estes sistemas podem ou não ser modificáveis, através de atualizações, sendo classificados da forma que se segue:

Ambiente Operacional de Propósito Geral: este tipo de ambiente engloba os sistemas operacionais disponíveis comercialmente, gerenciando os processos, recursos e usuários no interior do perímetro criptográfico;

Ambiente Operacional Limitado: trata-se de um ambiente virtual não modificável, tal qual uma máquina virtual java, ou um ambiente proprietário em memória somente leitura (ROM);

Ambiente Operacional Modificável: um ambiente é dito modificável se form passível de reconfiguração por seu operador, de forma a adicionar, remover ou alterar suas funcionalidades, por procedimento de atualização de software/firmware.

4.3.7 Gerência de Chaves

O módulo criptográfico deverá gerenciar todo o ciclo de vida de chaves criptográficas, incluindo as fases de geração de chaves e números aleatórios, estabelecimento de chaves, distribuição, entrada e saída, armazenamento, e zeramento. Mecanismos devem ser providos com o intuito de preservar a confidencialidade, integridade, autenticidade e disponibilidade de chaves e parâmetros críticos, quando aplicável.

Para garantir a satisfação dos requisitos especificados em cada uma das fases acima citadas, o padrão enumera uma série de métodos, mecanismos, algoritmos e os respectivos modos de operação aprovados. Abaixo são sumarizados os principais requisitos aplicáveis a cada uma das fases do ciclo de vida de chaves criptográficas:

Geração de Números Aleatórios: a geração de números aleatórios poderá ser feita por geradores não aprovados, contanto que sejam utilizados apenas para gerar sementes ou vetores de inicialização, que por sua vez alimentem um algoritmo determinístico aprovado pelo padrão;

Geração de Chaves: a geração de chaves criptográficas deverá utilizar um método de geração de números aleatórios aprovado, respeitando a regra descrita anteriormente. O comprometimento do mecanismo de geração de chaves deve ser no mínimo tão custoso quanto adivinhar o conteúdo de uma das chaves por ele geradas;

Estabelecimento de Chaves: um módulo criptográfico pode permitir a distribuição segura de chaves entre módulos. Caso este procedimento seja fornecido, métodos aprovados de estabelecimento de chaves deverão ser empregados. O processo poderá ocorrer de forma manual, automática ou através de uma combinação destes;

Entrada e Saída de Chaves: se uma chave criptográfica puder ser importada ou exportada do módulo criptográfico, esta operação poderá ocorrer de forma manual, como através de um teclado, ou eletrônica, com o uso de um cartão inteligente ou outro dispositivo de carga de chaves. Se as chaves forem importadas e exportadas de forma cifrada, deverão utilizar um algoritmo aprovado;

Armazenamento de Chaves: as chaves poderão permanecer no interior da fronteira criptográfica na forma cifrada ou em texto claro. Contudo, as chaves jamais deverão ser expostas fora do perímetro monitorado, a usuários não autorizados;

Zeramanto de Chaves: um módulo criptográfico deverá prover mecanismos capazes de apagar quaisquer vestígios de chaves criptográficas e demais PCSs, armazenados em texto claro em seu interior. Chaves protegidas física ou logicamente por mecanismos aprovados não se aplicam a este requisito.

4.3.8 Interferência e Compatibilidade Eletromagnética

O padrão estabelece requisitos a serem cumpridos com relação à emissão de interferências, bem como a compatibilidade com normas internacionalmente aceitas. Os módulos criptográficos devem seguir as normas estabelecidas pela agência do governo norte americano chamada de Comissão Federal de Comunicações (FCC, 1998).

4.3.9 Auto testes

Um módulo criptográfico deverá prover a funcionalidade de auto testes para suas funções de segurança. Os testes devem ser executados durante a inicialização do módulo (auto testes de inicialização) ou sob demanda (auto testes condicionais), sempre que a respectiva função for invocada. Em caso de falha no teste, o módulo deverá entrar em um estado de erro, externalizando-o via interface de saída de estado. O módulo não deverá permitir a execução de nenhum serviço criptográfico enquanto encontrar-se nesse estado.

Entre os auto testes de inicialização, pode-se citar os testes de resposta conhecida para algoritmos criptográficos. Neles, a partir de entradas para as quais já se sabe de antemão o resultado esperado, testa-se o correto funcionamento do algoritmo criptográfico. Adicionalmente, devem ser realizados testes de integridade nos componentes de software e firmware, através de códigos de detecção de erro.

Dentre os testes condicionais estão os utilizados na inicialização, além de outros, como o do gerador de números aleatórios contínuo. Neste, é solicitado ao GNA que forneça um número aleatório, durante a inicialização, que não será utilizado, mas sim armazenado para comparação com os próximos a serem gerados. Caso o número se repita, o teste deverá acusar falha.

4.3.10 Garantias de Projeto

As garantias de projeto têm como objetivo avaliar as metodologias e boas práticas de desenvolvimento utilizadas na produção do módulo de segurança criptográfica. As garantias envolvem o projeto, desenvolvimento, testes, entrega, instalação, configuração e operação.

Dentre os requisitos avaliados, pode-se citar o manual do usuário, que deve conter informações que garantam a segurança no manuseio, transporte, configuração e utilização do dispositivo.

4.3.11 Mitigação de Outros Ataques

Embora tenha sido concebido para mapear requisitos relacionados às mais variadas áreas envolvidas no desenvolvimento de um módulo criptográfico, dia após dia surgem novos ataques, o que torna difícil a atualidade da norma.

Para tratar esta questão, há no padrão uma seção abordando os ataques que ainda não haviam sido elaborados ou divulgados no momento de seu lançamento, bem como aqueles para os quais ainda não existiam formas de mitigação.

Segundo o padrão, a política de segurança do módulo deverá especificar os mecanismos utilizados para mitigar esses ataques específicos. Tais mecanismos serão então avaliados e incorporados ao processo, à medida que testes adequados sejam desenvolvidos.

4.3.12 Níveis de Segurança

Segundo o padrão, um dispositivo criptográfico aprovado deverá ser enquadrado em um dos quatro níveis de segurança possíveis. Tais níveis são organizados de forma incremental, sendo que um módulo classificado como de nível n cumpre todos os requisitos necessários para ser classificado como nível $n - 1$.

Para que um módulo possa alcançar um determinado nível de segurança n, todos os requisitos de cada área de avaliação, para o nível n deverão ter sido satisfeitos. Em outras palavras, se o módulo criptográfico atingiu nível 3 em todas as áreas, mas apenas nível 2 nos relacionados ao seu modelo de estados finitos, por exemplo, o nível constante em seu certificado de conformidade será 2.

As principais diferenças entre os quatro níveis de segurança são apresentadas, sintetizadamente, a seguir:

Nível de Segurança 1: O nível de segurança 1 não requer nenhuma proteção física, aplicando-se até mesmo a bibliotecas criptográficas em software. Um módulo certificado em nível 1 deverá comprovar o uso de, no mínimo, um algoritmo em modo de operação aprovado, possuir as funcionalidade de autotestes, bem como uma política de segurança, descrevendo os serviços, papéis e modelo de estados.

Nível de Segurança 2: No nível de segurança 2, o módulo criptográfico deverá restringir o acesso a seus serviços através da autenticação de cada um dos papéis providos. Em termos de segurança física, é requerido que um módulo no nível 2 empregue travas e lacres para proteger seu perímetro criptográfico, visando evidenciar possíveis tentativas de violação.

Quanto ao ambiente operacional, o módulo deverá utilizar um sistema validado pela norma *Common Criteria - CC*[2] (CC, 2009a; CC, 2009b) com nível de garantia mínimo *EAL2*.

Nível de Segurança 3: As principais adições do nível de segurança 3, com relação ao anterior, são a autenticação dos operadores autorizados, que neste nível passa a ser baseada em identidade, bem como a introdução de mecanismos de resposta a ataques, aplicados às portas e interfaces do módulo. Nesse nível, ainda, o ambiente operacional deverá satisfazer os requisitos do nível *EAL3*, da norma *CC*.

Um módulo nível 3 só deverá permitir a entrada de chaves e PCSs no formato cifrado ou empregando técnicas de compartilhamento de segredo. Por fim, com relação às garantias de projeto, o nível 3 requer implementação do módulo em linguagem de alto nível.

Nível de Segurança 4: Em seu nível mais restritivo, a norma FIPS 140 exige uma segurança física rígida capaz de evidenciar e responder a quaisquer tentativas de ataques, perpetradas contra quaisquer dos componentes do módulo. O nível 4 ainda exige uma das duas estratégias: o monitoramento das condições do ambiente externo ao módulo, com vista a detectar quaisquer variações que fujam às condições para as quais o módulo foi projetado; ou a comprovação, por meio de testes, que tais variações não afetarão a segurança do módulo criptográfico.

4.4 PKCS #11

Um outro padrão largamente empregado por desenvolvedores de MSCs é o padrão *Public Key Cryptographic Standard #11* (RSA, 1999a), proposto pelos Laboratórios RSA (RSA, 2010). Essa empresa foi fundada

[2]Norma internacionalmente reconhecida que estabelece requisitos de segurança comuns a diferentes sistemas computacionais (SOs, firewalls, roteadores), com diferentes níveis de confiança, os chamados *Evaluation Assurance Levels*, ou *EALs*.

pelos criadores do próprio algoritmo criptográfico de chave pública RSA (CC, 2002).

O padrão PKCS #11, também conhecido como *Cryptoki*[3] foi desenvolvido com o intuito de permitir que diferentes implementações de módulos criprográficos pudessem se comunicar de forma padronizada com as aplicações terceiros, usuárias de seus serviços.

O padrão define uma API[4] na linguagem C, a ser implementada pelos fabricantes das soluções, contendo um conjunto de funções gerenciais e criptográficas. A figura 4.3 ilustra o emprego da especificação na padronização do acesso a módulos criptográficos.

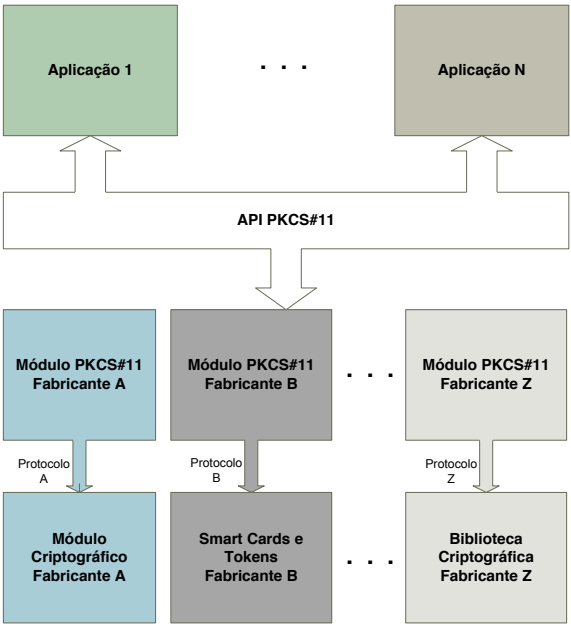

Figura 4.3: Arquitetura externa da solução proposta pelo padrão PKCS #11

De acordo com a figura 4.3, é visto que uma mesma API, contendo um conjunto de funções, pode se comunicar com diferentes módulos criptográficos, correspondendo a implementações do padrão, conhecidas como módulos PKCS #11. De forma mais prática, a API é composta por um conjunto de arquivos de cabeçalho[5], enquanto as implementações proprietárias são agrupadas em bibliotecas dinâmicas[6].

[3] Do inglês *Cryptographic Token Interface*.

[4] Do ingles Application Programming Interface, consiste em uma interface bem definida para acesso a funcionalidades de uma aplicação, tal qual biblioteca, framework ou, neste caso, dispositivo criptográfico.

[5] Do inglês *header files*.

[6] Do inglês *dynamic-link libraries* ou *shared libraries*.

É importante notar que os módulos PKCS #11 são específicos a um dispositivo ou mesmo a um conjunto de dispositivos que se comunicam sob um mesmo protocolo. É frequente o uso de uma mesma implementação de um fabricante abranja toda uma família de soluções ou mesmo o total delas, conforme ilustrado no módulo *B*. Adicionalmente, um módulo PKCS #11 pode ser totalmente baseado em software, sem o acesso a um dispositivo, como é mostrado pelo módulo *Z*, que emprega uma biblioteca criptográfica em seus serviços.

4.4.1 Arquitetura Interna

Internamente, a Cryptoki provê uma abstração de detalhes de implementação, através dos conceitos de *slots* e *tokens* lógicos. Os tokens são containers de chaves e outros objetos, que devem se conectar a um slot, para que possam ser acessados. A abstração dá a idéia de um cartão inteligente sendo inserido em uma leitora, apesar de não ser necessária a existência de uma leitora e cartão físicos. Para MSCs, esse modelo promove uma separação do espaço do módulo, para cada perfil existente, permitindo o acesso paralelo aos objetos, bem como o gerenciamento de chaves distintas, por custodiantes independentes.

4.4.2 Perfis de Acesso

Em termos de perfil de acesso, o padrão PKCS #11 opera com os mesmos perfis descritos pela norma FIPS: o oficial de segurança e o operador do token. Para autenticação do perfil, são utilizadas senhas pessoais, sendo que a maioria dos dispositivos possui formas de bloqueio, mediante a um número máximo de tentativas inválidas.

4.4.3 Objetos Gerenciados

Segundo o padrão PKCS #11, é possível o armazenamento dos seguintes tipos de dados, no interior do token:

Dados: representação de um segredo genérico, na forma de um conjunto de bytes;

Certificado: objeto que representa um certificado digital X.509;

Chave Pública: objeto que representa uma chave assimétrica pública;

Chave Privada: objeto representando uma chave assimétrica privada;

Chave Secreta: objeto representando uma chave simétrica.

Atrelados a esses objetos, há um conjunto de atributos, que definem as características de cada um. Os atributos tem o formato `CHAVE=VALOR`, podendo assumir os valores `CK_TRUE` e `CK_FALSE` , para verdadeiro e

falso, respectivamente. Tais valores podem ser definidos durante a criação do objeto ou alterados durante a vida útil do objeto. Nem todos os atributos, contudo, permitem a modificação de seus valores, após a criação de um objeto.

Dos atributos definidos pelo padrão, os mais relevantes são descritos a seguir, sendo que os não modificáveis estão marcados com a sigla *(RO)*:

CKA_SENSITIVE (RO): Este atributo define se um objeto trata-se ou não de informação crítica. Quando marcado como sensível, um objeto só poderá deixar o perímetro controlado pelo HSM na forma cifrada. Um objeto pode ter este valor alterado, após sua criação, apenas para `CK_TRUE` e jamais poderá ter este atributo alterado para `CK_FALSE`;

CKA_PRIVATE (RO): Este atributo define se o objeto poderá ser acessado por qualquer usuário, sem a necessidade de autenticação ou se será requerida a autenticação do usuário do token para que possa ser utilizado;

CKA_ENCRYPT: Este atributo se aplica a chaves, indicando se estas podem ser utilizadas para cifragem de dados;

CKA_DECRYPT: Este atributo se aplica a chaves, indicando se estas podem ser utilizadas para decifragem de dados;

CKA_WRAP: Indica que uma chave pode ser utilizada para cifrar outras chaves (agindo como chaves de transporte), de forma a permitir que parâmetros marcados como `CKA_SENSITIVE` deixem o perímetro criptográfico. Esta é a forma utilizada para fazer backups das chaves. As chaves a serem exportadas devem conter a flag `CKA_EXTRACTABLE`, descrita a seguir;

CKA_UNWRAP: Tem o efeito inverso à flag anterior, significando que a chave em questão pode ser utilizada para decifrar outras chaves;

CKA_EXTRACTABLE: Este atributo, quando marcado como verdadeiro, permite que o objeto deixe o perímetro criptográfico do MSC, podendo assim ser salvo em uma mídia insegura, conquanto que esteja devidamente cifrado por uma chave de transporte (marcada com `CKA_WRAP`).

CKA_SIGN: Parâmetro também aplicável a chaves, que permite definí-las como chaves de assinatura. Uma chave privada utilizada para assinar dados deverá estar com este atributo marcado como `CK_TRUE`;

CKA_VERIFY: Parâmetro também aplicável a chaves que permite definí-las como chaves de verificação de assinaturas;

CKA_NEVER_EXTRACTABLE (RO): Este atributo estará marcado como verdadeiro caso o objeto jamais tenha deixado o perímetro criptográfico do módulo;

CKA_ALWAYS_SENSITIVE (RO): Este atributo estará marcado como verdadeiro se o objeto sempre tiver sido considerado um parâmetro crítico de segurança, desde sua geração.

4.4.4 Serviços Criptográficos

Para prover acesso aos serviços de seu módulo criptográfico, os fabricantes devem implementar uma série de funções da API. As funções subdividem-se entre funções para gerencia de tokens e slots, estabelecimento e manutenção de uma sessão de uso, criação e destruição de objetos, entre outras.

Sempre que necessário utilizar uma chave secreta, a API permite que seja feita uma busca pela mesma. Esta busca retornará não o objeto em si, mas um identificador único para o objeto, de forma que este possa ser utilizado nas funções que demandam a indicação de uma chave.

As principais funções que dão acesso aos serviços criptográficos do módulo, bem como as destinadas à gestão de chaves, são listadas a seguir:

C_Encrypt: Função disponível para cifragem de dados;

C_Decrypt: Função disponível para decifragem de dados;

C_Sign: Função para assinatura digital;

C_Verify: Função disponível para validação de assinaturas.

Além das funções para acesso aos serviços, vale ressaltar também as envolvidas diretamente na gestão de chaves:

C_GenerateKey: Função disponível para geração de chaves criptográficas simétricas;

C_GenerateKeyPair: Função disponível para geração de chaves criptográficas assimétricas;

C_WrapKey: Função disponível para extrair uma chave, cifrada por uma segunda, chamada de chave de transporte;

C_UnwrapKey: Função para importar uma chave, criptografada por uma chave de transporte;

C_DeriveKey: Função para derivar uma segunda chave simétrica a partir de uma chave já existente.

Com a definição das funções e objetos em um nível conceitual, o padrão PKCS #11 possibilita uma independência de fabricante, permitindo que os desenvolvedores de aplicações abstraiam detalhes relacionados à implementação dos dispositivos. Todavia, o padrão não obriga os fabricantes a implementarem todas as funções, o que pode gerar algum trabalho ao migrar de um módulo para outro.

4.5 ESTUDO DE CASO: EXPORTAÇÃO DE CHAVES DE UM MSC DE MERCADO

Com a adoção de dispositivos criptográficos para a gestão do ciclo de vida de chaves, recaem sobre esses dispositivos muitas desconfianças com relação ao nível de segurança provido às chaves gerenciadas. A principal preocupação nesses casos está relacionada à impossibilidade em se extrair uma chave privada gerenciada em formato de texto claro. Uma vez que a maioria dos módulos criptográficos utiliza procedimentos de cópia proprietários e/ou direcionados a um segundo dispositivo idêntico (NCIPHER, 2001; SAFENET, 2008), o nível de segurança é muitas vezes uma incógnita.

Com o objetivo de verificar a segurança de um módulo criptográfico de mercado, avaliando existência de vulnerabilidades que possam permitir a um atacante a obtenção de quaisquer informações ou privilégios, suficientes para comprometer a segurança de uma ou mais chaves criptográficas gerenciadas, foi avaliado o módulo criptográfico *ProtectServer Gold*.

As seções 4.5.1 a 4.5.7 relatam os procedimentos realizados nos teste, bem como os resultados obtidos.

4.5.1 Módulo Criptográfico ProtectServer Gold

O *ProtectServer Gold PCI* consiste em um dispositivo criptográfico, acoplado a uma máquina hospedeira através de interface *Peripheral Component Interconnect* (*PCI*), provendo funcionalidades como gestão de chaves simétricas e assimétricas. O módulo conta ainda com aceleração criptográfica, que permite a execução dos principais algoritmos, de forma otimizada. O ProtectServer possui certificado FIPS 140-2 nível 3.

4.5.2 Objetos Gerenciados

O MSC permite o armazenamento de objetos do tipo Certificado, Chave Pública, Chave Privada, Chave Secreta e Dado Genérico. Cada um desses objetos, ao serem criados ou importados para o módulo, passam a carregar consigo um conjunto adicional de atributos. Tais atributos têm o formado `CHAVE=VALOR`, onde o último pode assumir valores `CK_TRUE` ou `CK_FALSE`.

4.5.3 Modos de Operação

O ProtectServer Orange permite que seus administradores definam uma série de configurações de uso, que implicarão diretamente nas funcionalidades do módulo. Uma das opções passíveis de configuração, por exemplo, é a *Sem PIN em claro*. Esta opção, quando ativa, não permite que os PINs informados pelos usuários dos `tokens` trafeguem da máquina hospedeira para o MSC em texto claro, demandando assim sua cifragem. Através da composição de conjuntos de configurações, o módulo disponibiliza diferentes modos de operação. Um exemplo de modo de operação é o Modo FIPS, em que o MSC só operará com algoritmos aprovados pela norma FIPS 140-2, além de ativar outras cinco opções:

Sem criptografia púbica: o MSC não permitirá criptografia sem a autenticação do usuário do `token` ;

Proteção de Autenticação: Dados sensíveis, trocados entre hospedeiro e MSC estão sendo assinados;

Apagar antes de Atualizar: Em caso de necessidade de atualização de seu firmware, o MSC será completamente apagado, restaurando suas configurações de fábrica e destruindo todas as informações previamente armazenadas;

Sem PIN em claro: Os PINs utilizados na autenticação dos usuários e oficiais de segurança deverão trafegar de forma cifrada sobre o barramento PCI, durante a comunicação com o MSC;

Modo bloqueado: O MSC não permitirá a violação do modo de operação. A alteração de configurações que violem o modo não será permitida.

4.5.4 Procedimento de Backup

O ProtectServer prevê a cópia de chaves entre um MSC e outro, do mesmo fabricante, dispondo para tanto de duas opções.

Na primeira, a exportação de chaves é feita por meio de cartões inteligentes, onde a chave é dividida em um total de N pedaços e distribuída entre N cartões, de forma que para recompor seu material, serão necessários um número mínimo M de custodiantes, sendo $0 < M \leq N$.

O segundo método exige uma chave de transporte, que será utilizada para criptografar a chave a ser exportada e transportá-la de um hardware para outro.

O procedimento de exportação de chaves via chave simétrica de transporte é esquematizado na figura 4.4.

O esquema mostra que, para exportação de uma chave ou outra informação sensível, uma segunda chave deverá ser utilizada como veículo de transporte, do Inglês *Key Encryption Key* (*KEK*), sendo gerada no MSC de destino, a partir de um conjunto de componentes. Esses componentes

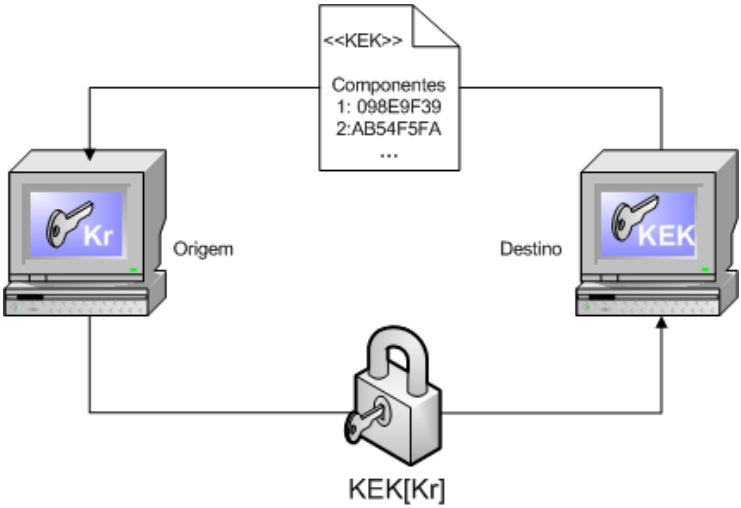

Figura 4.4: Esquema de backup baseado em chave de transporte.

podem ser informados pelo usuário manualmente ou mesmo gerados aleatoriamente e anotados pelo usuário, para utilização no passo subsequente.

A partir daí, os componentes devem ser inseridos na máquina conectada ao MSC de origem, cuja chave será exportada, afim de recompor neste a chave secreta de transporte, gerada no MSC de destino.

Com a recomposição da chave de transporte, é finalmente possível exportar a chave assimétrica em questão para o MSC de destino, que já conhece a chave de transporte utilizada para a cifragem, sendo assim capaz de desfazer a operação.

4.5.5 Metodologia Utilizada no Ataque

Para compreensão da interface de programação provida pelo fabricante do MSC, foi analisada a documentação técnica contida nos manuais fornecidos com o produto. Ainda, buscou-se na literatura por vulnerabilidades conhecidas do ProtectServer Orange ou mesmo do padrão PKCS #11 (vide seção 4.4), uma das formas de acesso ao módulo.

Clulow (2003) apresenta uma série de ataques à interface padrão PKCS #11, capazes de enfraquecer a segurança ou mesmo permitir a obtenção de material sensível sem qualquer proteção, contrariando assim as boas práticas sugeridas pela norma FIPS 140-2, apresentadas na seção 4.3.

Com base em um modelo formal para a *Cryptoki*, empregado na análise do padrão PKCS #11, Delaune et al. (2008) conduziram uma série de experimentos que revelaram ainda outras fragilidades relacionadas ao padrão.

Uma ferramenta capaz de testar diferentes implementações do padrão PKCS #11 foi desenvolvida por Bortolozzo et al. (2009), contemplando os ataques descobertos por Clulow.

Um artigo recente do mesmo grupo de pesquisa apresentou uma segunda ferramenta, batizada de *Tookan*, implementando o modelo formal apresentado por Delaune, Kremer e Steel. Esse artefato possibilitou uma análise detalhada de alguns dispositivos de mercado, quanto aos ataques revelados pelos referidos autores (BORTOLOZZO et al., 2010). O artigo apresenta o resultado de testes realizados com a ferramenta sobre vários dispositivos criptográficos de mercado, de fabricantes diversos, em uso na atualidade. Os testes tinham como foco principal os dispositivos de uso pessoal, comumente conhecidos como *tokens*, além dos populares cartões inteligentes. O resultado dos testes mostra uma realidade preocupante, onde mais de 40% dos dispositivos testados mostraram-se vulneráveis a pelo menos um dos ataques executados.

Uma interface de acesso a módulos criptográficos comprovadamente segura é apresentada por Cachin e Chandran (2009), mas é fato que a *Cryptoki*, definida no padrão PKCS #11, é a interface padrão de mercado, consistindo na mais adotada para acesso a dispositivos criptográficos, tanto por parte dos fabricantes, quanto por parte dos usuários dos serviços de um MSC.

A partir dos ataques encontrados na literatura, foi desenvolvido um protótipo, intitulado *pkcs8_extractor*, com a finalidade de explorar uma das vulnerabilidades reportadas por Clulow, que resultariam na extração de uma chave assimétrica no formato definido pelo padrão PKCS #8 (RSA, 1993).

4.5.6 Vulnerabilidade Explorada

O ataque utilizado, nomeado por Clulow como Ataque da Separação de Chaves[7], aproveita-se do fato de que é possível, segundo a especificação do padrão, atribuir a uma mesma chave de transporte dois atributos conflitantes que, combinados, permitem extrair chaves e informações sensíveis do MSC em formato de texto claro.

Os referidos atributos são `CKA_WRAP` e `CKA_DECRYPT`. Quando marcada com o primeiro, uma chave tem a permissão para atuar como chave de transporte ou, em outras palavras, pode ser empregada na cifragem de outras chaves, através da função `C_WrapKey`. Uma chave, com o segundo atributo, pode ser utilizada para decifrar qualquer dado por ela cifrado, quando submetido ao MSC através da função `C_Decrypt`. As respectivas assinaturas das funções são apresentadas no fragmento de código 4.1.

As funções definidas na tabela 4.1 apresentam alguns parâmetros específicos da arquitetura implementada pela interface, mas permitem observar alguns detalhes importantes na execução do ataque pretendido. A

[7] Do Inglês *Key Separation Attack*

```
CK_DEFINE_FUNCTION(CK_RV, C_WrapKey)
(
    CK_SESSION_HANDLE hSession,
    CK_MECHANISM_PTR pMechanism,
    CK_OBJECT_HANDLE hWrappingKey,
    CK_OBJECT_HANDLE hKey,
    CK_BYTE_PTR pWrappedKey,
    CK_ULONG_PTR pulWrappedKeyLen
);

CK_DEFINE_FUNCTION(CK_RV, C_DecryptInit)
(
    CK_SESSION_HANDLE hSession,
    CK_MECHANISM_PTR pMechanism,
    CK_OBJECT_HANDLE hKey
);

CK_DEFINE_FUNCTION(CK_RV, C_Decrypt)
(
    CK_SESSION_HANDLE hSession,
    CK_BYTE_PTR pEncryptedData,
    CK_ULONG ulEncryptedDataLen,
    CK_BYTE_PTR pData,
    CK_ULONG_PTR pulDataLen
);
```

Fragmento 4.1: Funções utilizadas na execução do ataque de separação de chaves.

função `C_WrapKey` recebe como entrada o identificador de uma chave de transporte `hWrappingKey`, utilizando-a na proteção (cifragem) de uma segunda chave, identificada pelo parâmetro `hKey`. É importante notar que ambas as chaves já encontram-se no interior do perímetro criptográfico do módulo, tratando-se os referidos parâmetros apenas de identificadores. O resultado do procedimento, contudo, implica na extração da chave identificada por `hKey`, que encontrar-se-á acessível através da variável `pWrappedKey`.

O procedimento de decifra mento de dados, por sua vez, é realizado em dois passos distintos. O inicial apenas define o algoritmo e a chave a serem utilizados. O segundo, `C_Decrypt`, submete os dados cifrados ao dispositivo, através do parâmetro `pEncryptedData`. O MSC, por sua vez, decifra os dados e os devolve, armazenando-os na variável `pData`.

Segundo a definição do padrão PKCS #11, uma chave privada deverá possuir o atributo `CKA_SENSITIVE` ativo e somente poderá deixar o ambiente seguro no formato cifrado. Para que uma chave possa ser extraída do ambiente, mesmo que de forma cifrada, esta deverá possuir o

atributo `CKA_EXTRACTABLE` ativo.

Partindo-se do princípio de que existe, no interior do módulo criptográfico, uma chave privada k_p, passível de extração, o ataque subdivide-se em três passos principais:

1. Gera-se uma chave k_{td}, dando a ela permissão para transporte de chaves e deciframento de dados, através dos atributos `CK_UNWRAP` e `CK_DECRYPT`, respectivamente;

2. Utiliza-se a chave k_{td} em conjunto com a função `C_WrapKey`, para extrair a chave k_p (aproveitando-se do atributo `CKA_WRAP`), de forma que o resultado será $\{k_p\}k_{td}$;

3. Submete-se o resultado do passo anterior novamente ao módulo, desta vez utilizando-se a função `C_Decrypt`, tirando proveito da segunda propriedade de k_{td} , `CKA_DECRYPT`. O resultado obtido é k_p, em texto claro, no formato PKCS #8.

A figura 4.5 ilustra o procedimento, passo a passo, desde a importação da chave de transporte até a decifragem da chave assimétrica.

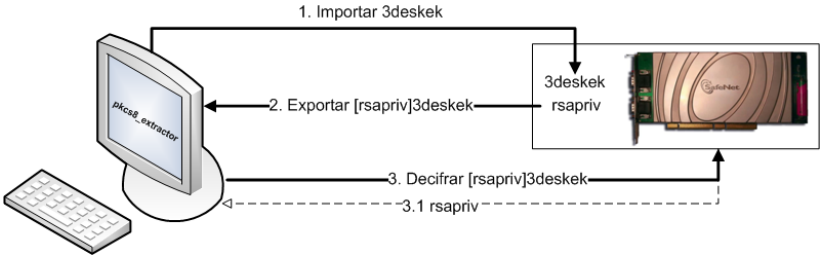

Figura 4.5: Esquema ilustrando o processo de extração de uma chave assimétrica com a aplicação *pkcs8_extractor*.

É importante perceber que, no procedimento descrito, o atacante nem ao menos precisa conhecer o conteúdo da chave de transporte, visto que o passo de decifração é realizado no interior do próprio módulo.

4.5.7 Implementação do Ataque

Para analisar a suscetibilidade do MSC *Safenet ProtectServer Gold* ao ataque descrito na seção 4.5.6, foi implementada uma aplicação em linguagem C++, fazendo uso da interface de programação *Cryptoki*, bem como o módulo implementando o padrão, fornecido pelo fabricante.

A aplicação teve como objetivo a execução das seguintes tarefas:

1. Abrir uma sessão de uso com o `token`onde uma chave privada encontrava-se armazenada;

2. Autenticar o usuário do `token` , usando para isso seu PIN;

3. Gerar uma chave com as propriedades necessárias a sua utilização como chave de transporte e de deciframento;

4. Procurar por uma chave assimétrica privada, candidata à exportação;

5. Exportar a chave assimétrica cifrada pela chave de transporte;

6. Utilizar a própria chave de transporte para efetuar o deciframento da chave privada, expondo seu conteúdo em claro;

7. Salvar o resultado do deciframento em arquivo;

8. Fechar a seção e terminar.

Pode-se perceber que, no passo 2, é necessário o conhecimento de uma informação sigilosa, o PIN do usuário, para execução do processo.

Com a execução da rotina acima descrita, foi possível obter a chave privada do módulo criptográfico, em texto claro, mostrando que o módulo é suscetível ao ataque revelado por Clulow, mesmo quando operando em modo aprovado pela norma FIPS 140-2.

4.6 CONSIDERAÇÕES DO CAPÍTULO

Este capítulo apresentou de forma geral as características dos módulos de segurança criptográficos, dispositivos altamente especializados que implementam protocolos de gestão de chaves. Foram apresentados ainda dois padrões amplamente adotados no contexto de módulos criptográficos. O primeiro, FIPS 140-2, orienta o projeto e implementação de MSCs, buscando certificar que o módulo foi desenvolvido com as garantias necessárias à gestão segura de chaves.

O segundo padrão diz respeito à abstração das inúmeras implementações de MSCs, apresentando aos desenvolvedores de aplicações, consumidores dos serviços do módulo, uma interface de programação padrão, votada apenas aos serviços oferecidos.

O capítulo buscou mostrar ainda que, mesmo seguindo padrões no projeto e desenvolvimento dos módulos criptográficos, ou mesmo que se utilize padrões consolidados para o acesso aos mesmos, o risco sobre as chaves gerenciadas por um módulo criptográfico persistem. Essa constatação reforça a necessidade de um sistema de rastreamento entre as múltiplas instâncias de uma chave privada. O sucesso na extração de uma chave secreta, em texto claro, de um módulo criptográfico real e amplamente utilizado no mercado, além de ser extremamente preocupante, permite comprovar algumas questões:

- Basear a segurança do procedimento de backup em métodos proprietários ou em transferências de dispositivo para dispositivo, sem manter um rastro entre as instâncias, é uma abordagem arriscada, visto que mesmo em um padrão conhecido e amplamente difundido, existem vulnerabilidades que possibilitaram o acesso indevido;

- O fato de um módulo criptográfico ter sido certificado, como ocorre com a norma FIPS 140-2, de maneira alguma garante que o módulo é inviolável. De fato, a certificação de um módulo criptográfico só busca garantir que este foi concebido considerando-se algumas das melhores práticas conhecidas, até o presente momento;

- A necessidade do conhecimento do PIN para autenticação do usuário, na obtenção do conteúdo da chave em texto claro, reduz a segurança do módulo à de um arquivo cifrado em uma mídia desprotegida, com a desvantagem de implicar uma falsa sensação de segurança. A falsa sensação pode dificultar ainda mais a evidenciação do vazamento.

Diante dos resultados e das reflexões apresentadas, fica evidente a necessidade de um procedimento de replicação de material sensível que busque não só evitar o vazamento desses materiais, mas que também evidencie toda e qualquer tentativa de acesso indevido ao seu conteúdo sigiloso, protegido no interior do perímetro criptográfico de um módulo.

5 GESTÃO DE MÚLTIPLAS CÓPIAS DE CHAVES ASSIMÉTRICAS

5.1 INTRODUÇÃO

No capítulo 2, foram descritos os modelos existentes na literatura, voltados ao ciclo de vida de chaves criptográficas. Contudo, conforme exposto na seção 2.5, os modelos de gestão existentes não preveem a gestão das múltiplas cópias de uma mesma chave criptográfica. A única menção feita, nos modelos existentes, com relação à cópia de chave, restringe-se a cópias de segurança para fins de backup. A ideia é de que, em caso de indisponibilidade da chave original, a cópia de segurança poderia ser restaurada.

Chaves assimétricas são extremamente sensíveis, pois são empregadas na identificação de certa entidade, como forma de prover a autoria e autenticidade das informações em meio eletrônico. Com a ampla adoção da criptografia de chaves públicas, em conjunto com a certificação digital, na garantia de integridade, sigilo, autenticidade e irretratabilidade no meio eletrônico, emergem inúmeras aplicações onde o custo de restauração de uma cópia de segurança implica em grande prejuízo, como por exemplo:

AC Raiz: Uma autoridade certificadora raiz em uma ICP de alta segurança tem um custo operacional bastante elevado, além de exigir cerimônias rígidas para cada uma de suas atividades. Dependendo da política adotada, pode ser necessário invalidar uma cerimônia, em caso de falha por indisponibilidade da chave privada da autoridade certificadora. Nesse caso, uma nova cerimônia deve ser agendada em caráter de urgência, tornando-se inconveniente para os envolvidos e acarretando custos indesejáveis;

AC Online Final: Para autoridades certificadoras online, em que há a emissão de certificados para entidades finais, a disponibilidade do serviço passa a ser um fator crítico. Nesses sistemas, uma interrupção do serviço pode representar um enorme prejuízo financeiro à instituição que mantém a AC online;

Sistema de Emissão de Notas Fiscais Eletrônicas: Com a implantação de sistemas de emissão de notas fiscais eletrônicas, as empresas precisam gerenciar suas chaves criptográficas, utilizadas para assinar as notas fiscais emitidas. Para uma empresa com grande número de emissões, pode ser necessária a replicação do material criptográfico entre suas filiais, visando redundância e descentralização do processo. Para esses sistemas, a alta demanda pode ser um fator impactante, que exige um balanceamento da carga de assinaturas.

Para cada um dos exemplos acima citados, faz-se necessário um modelo que suporte a gestão simultânea do ciclo de vida de mais de uma instância da chave privada. Com base nesse modelo, torna-se possível o espelhamento do ambiente de gestão, permitindo satisfazer as exigências relacionadas ao alto desempenho, às necessidades de distribuição do material em ambientes desconectados, bem como aos requisitos relacionados à manutenção da disponibilidade do serviço.

Este capítulo apresenta uma proposta para a gestão de múltiplas cópias de uma mesma chave criptográfica utilizando módulos criptográficos. A ideia é resolver os problemas de contingência e disponibilidade distintamente. No primeiro caso, propõe-se utilizar um módulo de produção, ou seja, com a chave liberada para uso sob controle dos custodiantes e um ou mais módulos de backup, não operacionais. Esses últimos seriam colocados em operação caso haja algum problema de indisponibilidade do módulo de produção. A seção 5.3 apresenta diversos cenários, considerando a questão da contingência.

A seção 5.3.1 apresenta o cenário contendo um módulo de produção e um módulo de backup. Como é utilizado, neste cenário, somente um módulo de backup, caso este apresente algum problema, o que pode ocorrer sem ser percebido, fica-se sem a possibilidade de recuperação de backup no caso de um sinistro. Para contornar esta dificuldade, a Seção 5.3.2 mostra como utilizar mais de um módulo de backup para um mesmo módulo operacional, destacando suas vantagens e desvantagens.

A seção 5.3.3 mostra que um módulo de backup pode ser usado para diversos módulos de produção, cada um com sua própria e distinta chave privada. Este cenário interessa muito se o objetivo for minimizar o custo dos backups, pois um único módulo seria usado para isso. Entretanto, como argumentado, o risco de perda do backup seria ainda maior que o cenário anterior.

E para gerir melhor o custo de backup num cenário de vários módulos operacionais distintos, pode-se usar vários módulos de backup, sendo que cada um desses poderia estar preparado para receber o backup de qualquer um dos módulos de produção. Este cenário é discutido na seção 5.3.4.

A seção 5.4 discute o uso de vários ambientes operacionais distintos para tratar a disponibilidade. A ideia é ter vários vários módulos criptográficos, cada um deles com cópia da mesma chave privada. Neste cenário, para manter a rastreabilidade de uso de cada um das cópias da chave privada, como será visto, é necessária a adoção de procedimentos mais complexos de gestão, que devem ser acompanhados de um sistema de auditoria e análise de registros, além de cerimônias mais extensas.

5.2 ESQUEMA DE BACKUP

Para a avaliação das diferentes estratégias de gestão de chaves, foi utilizado o protocolo de backup proposto por Souza et al. (2007), onde

não só as chaves e demais parâmetros críticos são exportados, na forma cifrada, como são também os perfis de custódia das chaves, de administração e de auditoria do sistema. O trabalho ressalta trambém a importância dos procedimentos de auditoria para a segurança do processo e define uma cerimônia simplificada para controle das cópias. A auditoria é feita com base em um conjunto de registros de uso, produzidos pelo MSC a cada evento realizado, que devem ser exportados periodicamente.

Uma visão geral da solução proposta pelos autores é ilustrada na figura 5.1, que mostra as diferentes fases do esquema de backup, até sua restauração.

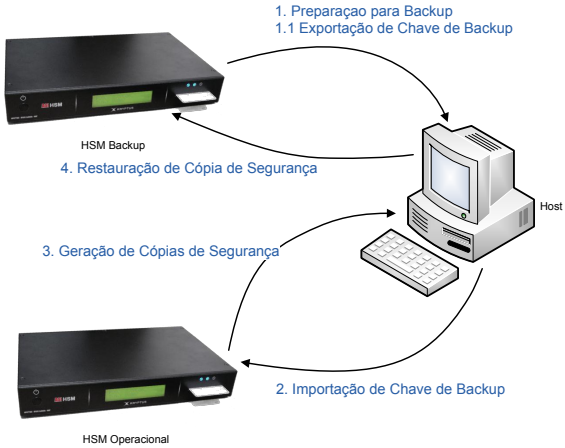

Figura 5.1: Esquema de backup proposto por Souza et al. (2007).

Segundo o esquema proposto, um MSC, em seu estado inicial, pode ser configurado de duas maneiras. A primeira é para uso como unidade operacional. A segunda seria como uma unidade de backup. Como unidade operacional, o sistema será capaz de gerenciar chaves criptográficas, ou seja, vai servir ao seu propósito principal. Como unidade de backup, ficará em espera até que uma cópia de segurança seja restaurada em seu interior, momento em que tornar-se-á operacional.

Há quatro fases principais no esquema de backup, como mostra a figura 5.1. A primeira consiste na preparação de um MSC, ainda não configurado, como ambiente de backup. Nesse momento ocorrerá a exportação de uma chave assimétrica pública, nomeada chave de backup, para uma máquina hospedeira.

Em um segundo momento, marcado pelo passo 2, a chave de backup é importada da máquina host para um MSC em operação. Com a conclusão desse passo, o MSC em operação poderá exportar cópias de segurança direcionadas ao MSC de backup, pois estarão cifradas com sua chave pública de backup. Como a chave privada correspondente encontra-se no MSC de backup, somente ele poderá decifrar o pacote, restaurando assim

as configurações originais.

A partir da importação de, pelo menos, uma chave de backup, o MSC em operação torna-se capaz de gerar uma ou mais cópias do seu ambiente operacional, procedimento representado pelo passo 3. Tais cópias podem ser armazenadas periodicamente em mídias ou mesmo na máquina hospedeira. Apesar de estarem devidamente cifradas, as cópias do ambiente, armazenadas em mídia convencional, estão sempre sujeitas a substituição ou destruição inadvertida. Logo, deve-se tomar o devido cuidado para garantir a disponibilidade das cópias.

O passo 4 mostra a restauração de uma das cópias de segurança geradas, no MSC de backup. A partir desse ponto, o MSC torna-se também um MSC operacional, funcionalmente idêntico ao original. O MSC restaurado pode ter no máximo uma defasagem em termos de registros de log, relativo às atividades executadas no MSC operacional, após a exportação do último backup.

Por conter dados sensíveis e confidenciais, antes protegidos pelo sistema de gestão de chaves, um esquema de backup deve lhes prover as mesmas garantias encontradas no interior do módulo criptográfico. Assim, pode-se apontar como propriedades desejáveis ao conteúdo replicado em um esquema de backup: autenticidade, integridade, confidencialidade e disponibilidade. Outra característica importante é a rastreabilidade da chave replicada, como forma de facilitar o controle sobre as múltiplas instâncias.

Com base no esquema de backup apresentado, bem como nas premissas de um sistema de backup seguro, será apresentado um conjunto de estratégias, visando dirimir os problemas apresentados no início do capítulo.

5.3 AMBIENTE OPERACIONAL ÚNICO

Este cenário consiste em manter apenas um MSC em operação, sendo que os demais serão utilizados para armazenar backups do primeiro. Com esta configuração, é necessária a exportação periódica de cópias de segurança, para que, em caso de qualquer falha do MSC em produção, o ambiente de contingência possa ser restabelecido de forma ágil e com mínimo ou nenhum prejuízo, através da restauração da cópia de segurança. O referido prejuízo seria tão grande quanto a defasagem do backup utilizado para recompor o ambiente operacional, sobretudo com relação aos registros de log.

Este esquema favorece a auditoria, pois sempre há apenas uma instância da chave ativa, visto que a restauração da cópia de segurança será realizada somente em caso de sinistro. Trata-se da abordagem proposta pelos principais modelos de gestão de chaves.

Seguindo a premissa de um único ambiente operacional, pode-se optar por beneficiar a tolerância a faltas ou a flexibilidade da estrutura de gerência de chaves, conforme será exposto nas seções abaixo.

5.3.1 Esquema 1:1

Neste esquema, cada MSC em operação conta com um MSC de backup capaz de receber suas cópias, estando este dedicado a tal função, conforme ilustrado na figura 5.2.

Figura 5.2: Ambiente operacional único com uma unidade de backup

Quando uma nova chave for criada no MSC, basta que uma nova cópia de segurança seja exportada para que o MSC de backup possa ser restaurado com a versão mais atual do ambiente.

O principal risco deste esquema é a falha do MSC de backup. Caso esta só seja detectada no momento da falha do MSC operacional, não será possível restabelecer as chaves privadas gerenciadas.

O sistema promove ainda uma subutilização do MSC de backup, que possivelmente passará toda a sua vida útil ocioso.

5.3.2 Cenário 1:N

Este cenário visa a solução do problema de ociosidade do MSC de backup, através de sua utilização como unidade de contingência de mais de um MSC operacional (cada um gerenciando chaves distintas), conforme mostrado na figura 5.3. Assim, aumenta-se a chance do backup ser posto em operação. É importante salientar que se deve, para tanto, importar a chave de backup em cada um dos MSCs em operação.

Esta configuração diminui ainda o custo total da estrutura de gerência de chaves, pois não é mais necessário um MSC de backup para cada MSC em produção.

Tal como no esquema 1:1, a principal desvantagem diz respeito à falha do MSC de backup, pois o risco de um comprometimento permanente do material é multiplicado pelo número de MSCs em operação.

5.3.3 Cenário N:1

Caso o valor da(s) chave(s) gerenciada(s) justifique um maior zelo com relação a possíveis falhas nos MSCs, é possível manter N MSCs de backup, preparados para receber cópias de um mesmo ambiente. Para isto, basta importar as três diferentes chaves de backup em um mesmo MSC operacional. O MSC passará então a cifrar os backups com as três chaves, de forma que qualquer dos MSCs possa decifrá-los. Este esquema é ilustrado na figura 5.4.

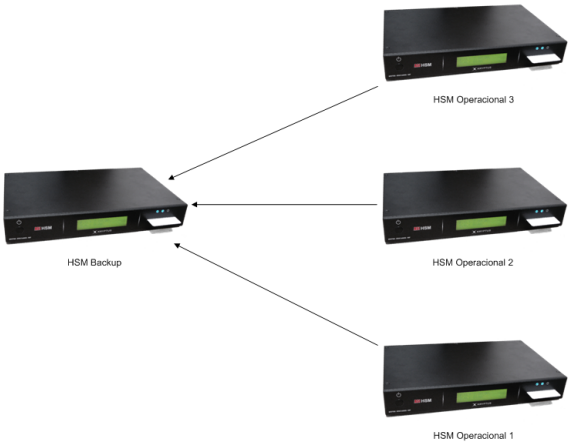

Figura 5.3: Ambientes operacionais distintos compartilhando uma mesma unidade de backup

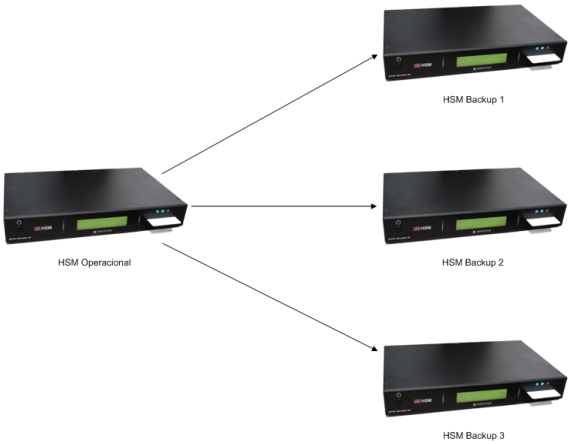

Figura 5.4: Ambiente operacional único exportando cópias para três unidades de backup

A configuração promove uma maior ociosidade dos MSCs de backup e, consequentemente, maior custo em termos de equipamentos e manutenção do ambiente.

No cenário proposto na figura 5.4, caso o ambiente operacional precise ser restaurado em um dos MSCs de backup, este tornar-se-á operacional, já com a capacidade de exportar suas cópias para os outros dois ambientes.

5.3.4 Esquema N:M

Neste esquema, é possível se trabalhar com um conjunto de MSCs de backup, preparados para receber chaves de um conjunto de MSCs em operação, apresentando assim uma maior segurança e flexibilidade no momento em que a restauração de um backup for necessária.

O esquema de backup N:M é ilustrado na figura 5.5.

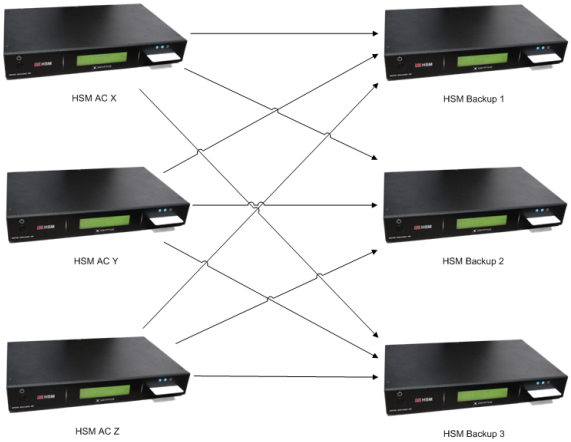

Figura 5.5: Vários ambientes operacionais distintos compartilhando um mesmo conjunto de unidades de backup

Este esquema de backup mostra-se adequado a situações onde seja necessário garantir a segurança de várias unidades operacionais, com um nível satisfatório de tolerância a faltas, controlando os custos com os módulos criptográficos de contingência. A gestão desse esquema, contudo, é bastante delicada, pois sempre que um MSC de backup for restaurado, deverá haver a inclusão de um novo MSC no grupo de backup. A chave do novo módulo de backup deverá ser importada nos MSCs operacionais, para que estes passem a cifrar os pacotes também para o novo MSC.

5.4 MÚLTIPLOS AMBIENTES OPERACIONAIS SIMULTÂNEOS

Para situações onde não seja adequada a geração periódica de cópias de segurança, uma alternativa é trabalhar com múltiplas cópias do mesmo ambiente operacional simultaneamente, bastando para isso executar todos os passos descritos no início da sessão, logo após a geração da(s) chave(s).

Este esquema promove uma restauração do ambiente de forma praticamente automática, necessitando apenas de um chaveamento da conexão com o MSC operacional para o MSC de contingência.

Este modelo permite maior garantia de disponibilidade da chave,

bem como torna possível o balanceamento da carga entre vários MSCs.

Dentro deste cenário, são possíveis ainda duas abordagens distintas, sob o ponto de vista da gestão dos módulos. A primeira seria manter os mesmos perfis de acesso do MSC original no ambiente replicado, tornando-os ambientes espelhados sob a mesma custódia. A segunda abordagem diz respeito a tornar os ambientes independentes, alterando os perfis de acesso após a replicação do ambiente.

Para manter os ambientes simultâneos independentes, após a restauração do backup é necessária a troca dos três perfis de acesso. Tal ação implica em grupos distintos de usuários gerenciando cada MSC, como pode ser visto na figura 5.6.

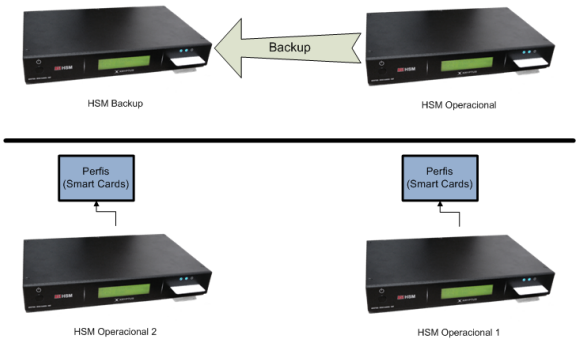

Figura 5.6: Ambientes Independentes

Já na gerência de ambientes espelhados, os grupos são os mesmos para ambos os ambientes, ou seja, o mesmo perfil de operadores, administradores e auditores detêm o controle dos dois módulos. A figura 5.7 ilustra esse cenário.

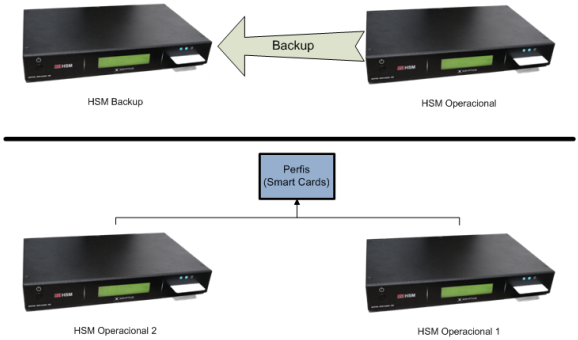

Figura 5.7: Ambientes Espelhados

5.5 CONSIDERAÇÕES DO CAPÍTULO

Este capítulo apresentou duas estratégias distintas de backup. A primeira consiste no armazenamento de cópias em mídia, que serão recuperadas somente em caso de indisponibilidade da chave operacional. Essa abordagem favorece um esquema de auditoria, que precisará se preocupar somente com os registros provenientes de um único MSC.

Os esquemas de backup baseados na existência de um único ambiente operacional apresentam-se viáveis para sistemas onde haja necessidade de garantir contingência sobre o material criptográfico em uso. Os esquemas apresentados, todavia, exigem cerimônias de recuperação, trazendo os problemas de custo, sumarizados no início deste capítulo. Outro ponto a ser considerado é a forma como estes backups serão armazenados, pois a indisponibilidade dos arquivos implica em impossibilidade de restauração do ambiente operacional.

O problema da disponibilidade das chaves pode ser contornado com múltiplos ambientes operacionais simultâneos, onde há a replicação dos ambientes operacionais. Com o uso do esquema, tanto a questão da alta demanda, quanto a da disponibilidade da chave, são beneficiadas. Dois critérios são prejudicados com essa abordagem, entretanto. Um deles é a auditoria, que precisa avaliar tantos registros de utilização da chave quantos forem os MSCs em operação. O outro diz respeito à escalabilidade, dado que os ambientes, após replicados, comportar-se-ão como MSCs independentes. A cada novo par de chaves gerado em um dos MSCs, por exemplo, seria necessário uma nova geração de backup, para proteger a nova chave. Os ambientes já replicados, contudo, ficariam desatualizados, sendo necessária uma nova replicação em outros MSCs. Essa realidade, na prática, torna a solução não escalável.

Em ambos os modelos apresentados, a rastreabilidade das chaves deve ser gerenciada através de procedimentos de auditoria, o que coloca os registros de uso dos MSCs em uma posição de evidência. Uma vez que as cópias das chaves em si são idênticas, cabe aos registros de utilização dos módulos criptográficos determinar qual das instâncias foi utilizada. Registros de uso, contudo, são passíveis de exportação e de exclusão, mediante autenticação dos auditores. Pode-se, seja por negligência ou mesmo por uso mal intencionado, fazer com que uma utilização da chave torne-se impossível de ser rastreada.

Faz-se assim necessário um modelo de gestão que possibilite de forma mais segura, determinar-se qual das instâncias de uma determinada chave privada foi utilizada para determinado fim, como por exemplo, uma assinatura digital.

6 NOVAS ABORDAGENS PARA A GESTÃO DE MÚLTIPLAS CÓPIAS DE CHAVES ASSIMÉTRICAS

6.1 INTRODUÇÃO

Este capítulo apresenta duas novas propostas que permitam a gestão de múltiplas instâncias de uma mesma chave privada, de forma que sejam unicamente identificadas.

A primeira abordagem baseia-se na derivação de chaves criptográficas assimétricas, em vez da simples replicação do material criptográfico. Dessa forma, evita-se a cópia desses materiais, sem contudo abrir mão da disponibilidade. Propõe-se novas primitivas criptográficas que possibilitem a geração chaves derivadas, que embora diferentes das chaves originais, possuam propriedades tais que permitam seu emprego como se fossem uma única. Resolve-se assim a questão da disponibilidade, sem ser necessário realizar a cópia da chave.

A seção 6.2 apresenta a proposta de derivação de chaves assimétricas, bem como define as propriedades das primitivas criptográficas em que se baseia. Em seguida, a seção 6.2.3 apresentará a aplicação das primitivas em um modelo de gestão que possibilite a utilização de múltiplas instâncias de uma mesma chave assimétrica, sem que seja necessário copiar a chave. A seção apresenta ainda três estratégias diferentes de derivação, de forma a flexibilizar a aplicação da proposta às diferentes necessidades de Autoridades Certificadoras. A seção 6.2.5 analisa os impactos da adoção do modelo proposto sobre cada uma das fases do ciclo de vida de chaves criptográficas, conforme descrito no capítulo 2.

A segunda proposta descreve um conjunto de procedimentos que, uma vez aplicados a um dos modelos de gestão apresentados no capítulo 2, possibilite garantir as mesmas propriedades atingidas com o procedimento de derivação de chaves, baseando-se apenas na modificação de alguns conceitos na gestão de certificados digitais. Assim, dispensa-se a necessidade de novas primitivas criptográficas, bem como minimiza-se a dependência do procedimento de cópia de chaves.

A seção 6.4 apresenta uma análise comparativa entre as abordagens propostas e as existentes e, por fim, a seção 6.5 dispõe algumas considerações a respeito das abordagens propostas.

6.2 DERIVAÇÃO DE CHAVES ASSIMÉTRICAS

Até o presente momento, os modelos de gestão de chaves apresentados previam a existência de uma única instância de uma determinada chave ativa. Conforme apresentado ao longo dos capítulos 3 e 5, todavia, há situações onde é importante que haja mais de uma instância ativa

da chave, por motivos de agilidade na recuperação de sinistros, disponibilidade ou mesmo atendimento a uma alta demanda, que exceda as capacidades de um módulo criptográfico.

Com os métodos de backup existentes na literatura, baseados na simples cópia do material sensível, é difícil a manutenção de algumas garantias de segurança desejáveis a uma chave privada, como sua unicidade, o rígido controle sobre sua utilização, bem como a rastreabilidade entre suas instâncias.

Para garantir as propriedades supracitadas, sem contudo abrir mão das referidas garantias, necessárias à gestão segura das chaves, bem como das infraestruturas de chaves públicas, propõe-se o uso de duas funções criptográficas, consistindo em novas primitivas, para um sistema criptográfico convencional, a ser apresentado na seção 6.2.1.

No capítulo 3 discutiu-se as preocupações, por parte de uma Autoridade Certificadora, com suas chaves privadas, sobretudo no que se refere à replicação desses materiais. Conforme exposto, os custodiantes da chave de uma AC sentem-se mais seguros quando têm a certeza de que este valor jamais será copiado, tendo-se em vista que a gestão de uma única instância da chave torna-se mais simples, além de enquadrar-se melhor aos modelos de gestão existentes, apresentados em 2. Contudo, conforme exposto ao longo desta dissertação, um modelo de gestão que preveja o uso simultâneo de múltiplas instâncias de uma mesma chave privada é de vital importância para a confiabilidade e escalabilidade da infraestrutura como um todo. No capítulo 6 foi proposta uma nova abordagem que permitisse a gestão de múltiplas instâncias de uma mesma chave, de forma simultânea, sem abrir mão das características de segurança desejáveis a um sistema de gestão. A proposta baseia-se na existência de novas primitivas criptográficas, não tendo assim aplicação prática com as técnicas de criptografia conhecidas até o presente momento.

Uma segunda abordagem, apresentada no capítulo **??**, utiliza técnicas já existentes para o estabelecimento de uma ICP interna ao contexto de uma AC. apresentando as mesmas características alcançadas pela técnica de derivação.

Este capítulo discute ainda os aspectos referentes a cada uma das abordagens propostas para a gestão de múltiplas instâncias de uma mesma chave, comparando-as com as abordagens existentes, na seção 6.4.

6.2.1 Sistema Criptográfico Convencional

Segundo Mao (2003), um sistema criptográfico convencional consistem em um mecanismo formado por:

- um espaço de mensagens em claro M, composto por um conjunto de strings sobre um alfabeto Γ;
- um espaço de mensagens cifradas C, composto por um conjunto das possíveis cifras sobre o mesmo alfabeto Γ;

- um espaço de chaves de cifração K e um espaço de chaves de decifração K^{-1};
- um algoritmo eficiente de geração de chaves $G : \mathbb{N} \rightarrow K \times K^{-1}$;
- um algoritmo eficiente de cifração $E : M \times K \rightarrow C$;
- um algoritmo eficiente de decifração $D : C \times K^{-1} \rightarrow M$;

Dado $n_t \in \mathbb{N}$, a geração das chaves dá-se da forma:

$$G(n_t) = \{(k_t, k_t^{-1}) \,|\, k_t \in K \ \wedge \ k_t^{-1} \in K^{-1}\} \tag{6.1}$$

Na equação acima, t consiste em um natural representando o tamanho das chaves k e k^{-1}.

O processo de cifração, responsável pela transformação de texto em claro em texto cifrado, é representado por:

$$E(k, m) = \{c \,|\, m \in M \ \wedge \ c \in C\} \tag{6.2}$$

De forma análoga, o processo de transformação de texto cifrado em texto claro, ou decifração, é dado por:

$$D(k^{-1}, c) = \{m \,|\, m \in M \ \wedge \ c \in C\} \tag{6.3}$$

O sistema acima descrito aplica-se tanto à criptografia simétrica quanto à assimétrica. A distinção entre as duas, contudo, está relacionada às chaves k e k^{-1}, que na criptografia simétrica consistem em uma mesma chave, enquanto na assimétrica, tratam-se de chaves distintas, compondo o par de chaves (k, k^{-1}).

Apesar de existirem na literatura formas de derivação de chaves simétricas, não foram encontrados vestígios de uma função equivalente, que operasse sobre chaves assimétricas. A proposta a seguir baseia-se na existência das referidas funções, ou seja, na existência de novas primitivas criptográficas, cujas propriedades serão descritas na seção 6.2.2.

6.2.2 Primitivas Criptográficas de Derivação de Chaves Assimétricas

Sejam, para o sistema criptográfico definido anteriormente, operando sobre chaves assimétricas, duas funções de derivação de chaves contemplando as características definidas a seguir:

Definição 1. *Seja K_t um subconjunto de K, composto por todas as chaves de tamanho t. Seja $\Delta : K_t \times K_t^{-1} \times \mathbb{N} \rightarrow K_t \times K_t^{-1}$ tal que:*

$$\Delta(k_0, k_0^{-1}, i) = \{(k_i, k_i^{-1}) \,|\, k_0, k_i \in K_t \ \wedge \ k_0^{-1}, k_i^{-1} \in K_t^{-1}\} \tag{6.4}$$

$$\forall i \in \mathbb{N} \,|\, 0 < i \leq 2^t,$$

com as seguintes propriedades adicionais:

1. Δ *é uma função não inversível ou cuja inversão é de difícil computo por uma máquina de Turing;*
2. $\Delta(k_i, k_i^{-1}, j) \neq \Delta(k_i, k_i^{-1}, l), \forall i, j, l \in \mathbb{N} \,|\, j \neq l$*;*
3. $\Delta(k_i, k_i^{-1}, j) = \Delta(k_0, k_0^{-1}, i + j), \forall i, j \in \mathbb{N} \,|\, i + j \leq 2^t$.

A função Δ, definida em 6.4, deriva um novo par de chaves (k_i, k_i^{-1}) a partir de um par inicial (k_0, k_0^{-1}), bem como um natural i. A primeira propriedade visa garantir que, a partir de um par de chaves derivado, não é possível obter o par original.

A segunda propriedade garante a derivação de pares de chaves distintos, a partir de um mesmo par de chaves de entrada (k_i, k_i^{-1}), para cada natural distinto. Esta característica torna a função uma injetora.

A terceira e última propriedade garante que um par derivado (k_i, k_i^{-1}) possa ser derivado tanto pela aplicação da função Δ sobre o par de chaves inicial (k_0, k_0^{-1}), com um dado natural i, quanto a partir de sucessivas aplicações da função Δ, sobre pares já derivados de (k_0, k_0^{-1}), a partir da mesma função.

A figura 6.1 permite melhor ilustrar o funcionamento da função Δ, segundo as propriedades definidas. A figura apresenta o grafo de derivadas formado, bem como as ligações existentes entre as instâncias

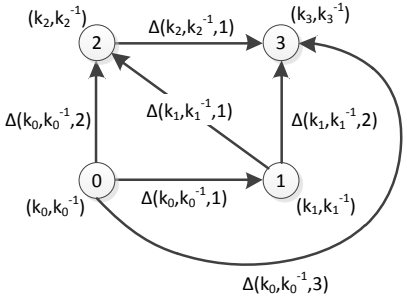

Figura 6.1: Grafo de derivação gerado para um grupo composto por quatro instâncias de pares de chaves derivados a partir de um par inicial (k_0, k_0^{-1}).

Como se pode notar na figura 6.1, sucessivas aplicações da função Δ produzem um grafo ordenado, sendo suas arestas definidas de acordo com as entradas da função, enquanto os vértices representam as instâncias derivadas. A identificação de cada instância é dada pelo um identificador numérico utilizado na derivação, a partir do par de chaves inicial.

A função de derivação permite assim o estabelecimento de vínculos, ou rastros, entre as chaves ancestrais e suas descendentes, relação esta que permite que as múltiplas instâncias sejam consideradas como um único par de chaves (k, k^{-1}).

Ainda de acordo com a figura, pode-se perceber que caminhos diferentes podem levar a uma mesma instância, devido à terceira propriedade da função Δ, conforme ilustrado na equação 6.5.

$$\Delta(k_0, k_0^{-1}, 3) = \Delta(k_1, k_1^{-1}, 2) = (k_3, k_3^{-1}) \tag{6.5}$$

Definidas as características da função de derivação de pares de chaves Δ, faz-se necessária também a existência de uma segunda função de derivação, desta vez operando somente sobre as inversas k_i^{-1}. Esta função, nomeada função Ω, é definida a seguir:

Definição 2. *Seja* $\Omega : K_t^{-1} \times \mathbb{N} \rightarrow K_t^{-1}$ *tal que:*

$$\Omega(k_0^{-1}, i) = \{k_i^{-1} \,|\, k_i^{-1} \in K_t^{-1}\} \tag{6.6}$$

$$\forall i \in \mathbb{N} \,|\, 0 < i \leq 2^t,$$

com as seguintes propriedades adicionais:

1. $\Omega(k_i^{-1}, j) \neq \Omega(k_i^{-1}, l), \forall i, j, l \in \mathbb{N} \,|\, j \neq l$;
2. $\Omega(k_i^{-1}, j) = \Omega(k_0^{-1}, i + j), \forall i, j \in \mathbb{N} \,|\, i + j \leq 2^t$.
3. $\Delta(k_i, k_i^{-1}, j) = (k_{i+j}, \Omega(k_i^{-1}, j)), \forall i, j \in \mathbb{N} \,|\, i + j \leq 2^t$

Como pode-se notar, a partir da definição 6.6, as propriedades *1* e *2* da função Ω se assemelham às encontradas na função Δ, contudo aplicadas apenas à parte pública do par de chaves assimétrico.

A terceira propriedade, contudo, visa garantir que o par de chaves derivado a partir da aplicação de um natural j ao *i-ésimo* par derivado de (k_0, k_0^{-1}), via função Δ, possua a mesma chave pública obtida com a função Ω, quando utilizados os mesmos parâmetros de entrada.

Definidas as primitivas criptográficas necessárias à gestão segura de múltiplas instâncias de uma mesma chave privada, a seção 6.2.3 apresenta a proposta do novo modelo.

6.2.3 Gestão de Múltiplas Instâncias de uma Chave Utilizando Derivação

Utilizando-se o algoritmo G para obter um par de chaves inicial (k_0, k_0^{-1}), o procedimento de derivação dar-se-á com a aplicação da função Δ ao par, em conjunto com um identificador único, com o intuito de distinguir cada uma das instâncias derivadas. Tal instância inicial, bem

como todas as derivadas a partir dela, direta ou indiretamente, comporão o par de chaves (k, k^{-1}).

Dessa forma, o conceito de par de chaves, no âmbito da abordagem proposta, distingue-se do utilizado nos modelos convencionais. Um par de chaves, para o modelo proposto, consiste no conjunto de pares de chaves (dos modelos convencionais), derivados a partir do par de chaves inicial (k_0, k_0^{-1}). Afim de evitar ambiguidades, o par de chaves (k, k^{-1}) composto por um conjunto de chaves derivadas, será assim definido, enquanto cada um dos elementos que o compõem serão chamados de instâncias do par de chaves, representados por $\{(k_0, k_0^{-1}), (k_1, k_1^{-1}), .., (k_n, k_n^{-1})\}$.

A ilustração 6.2 permite melhor visualizar o conceito de par de chaves Δ, para o modelo proposto.

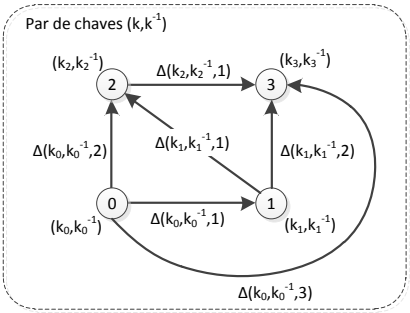

Figura 6.2: Conceito de um par de chaves (k, k^{-1}), segundo o modelo proposto.

Partindo-se desse conceito, sempre que necessário efetuar a replicação de uma chave criptográfica, por qualquer um dos motivos já citados, tal reprodução basear-se-ia não na cópia, mas sim na derivação do material criptográfico. Dessa forma, nunca haveria duas instâncias iguais do mesmo par de chaves, mantendo-se a unicidade da mesma.

Tal como foi visto na seção 6.2.2, a função delta permite que caminhos distintos levem a uma mesma instância (k_i, k_i^{-1}). No contexto da gestão de chaves, essa propriedade não é desejada, pois é importante que haja um caminho único entre o par de chaves ancestral e cada um de seus derivados. Esta restrição é fundamental para a garantia da rastreabilidade das chaves, bem como para garantir que não haja colisões entre as múltiplas instâncias do par de chaves.

Como forma de restringir a formação das cadeias de chaves derivadas, faz-se necessária a adoção de uma estratégia de replicação. As seções 6.2.3.1 a 6.2.4.1 apresentarão possíveis abordagens quanto à produção de múltiplas instâncias derivadas, de forma a permitir uma rastreabilidade precisa.

6.2.3.1 Derivação em Estrela

Partindo-se sempre da instância inicial (k_0, k_0^{-1}), e possível derivar n pares, variando-se somente o identificador único. Esta é a abordagem da derivação em estrela, para composição de um par de chaves (k, k^{-1}).

A figura 6.3 detalha o esquema de derivação, baseado em sucessivas aplicações da função Δ, sobre um mesmo par de chaves inicial.

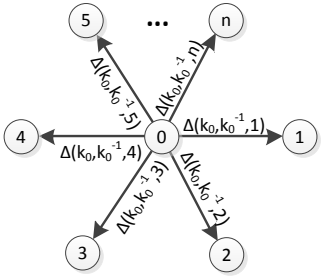

Figura 6.3: Cópia de chaves através de sucessivas aplicações da função Δ sobre a instância inicial (k_0, k_0^{-1}).

O algoritmo 6.1 detalha o procedimento de replicação de chaves na forma de estrela.

Algoritmo derivaInstanciaEmEstrela()

Deriva um par de chaves i, a partir do par (k_0, k_0^{-1}), bem como um contador de instâncias c, pré inicializado com o valor 1. Esse contador deve ser mantido em memória persistente, representada no algoritmo por bd.

1: $c \leftarrow$ leiaContador(bd)

2: $(k_i, k_i^{-1}) \leftarrow \Delta(k_0, k_0^{-1}, c)$

3: $c \leftarrow c + 1$

4: atualizeContador(c, bd)

5: retorne (k_i, k_i^{-1})

Algoritmo 6.1: Algoritmo de derivação de chaves utilizando a estratégia de derivação em Estrela.

6.2.4 Derivação em Lista

A estratégia de derivação em lista consiste na aplicação da função Δ sempre sobre a última chave derivada. Assim, o par de chaves (k, k^{-1}) será formado por uma lista de instâncias derivadas.

A figura 6.4 ilustra o processo de derivação de chaves, para esse cenário.

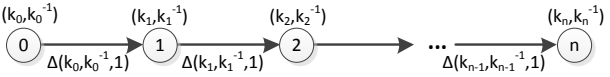

Figura 6.4: Estratégia de derivação de chaves em lista.

A abordagem tem como característica principal a fixação do identificador i, variando-se apenas o par de chaves utilizado para produzir a derivada. O algoritmo 6.2 ilustra o procedimento utilizado para derivar pares de chaves, segundo esta abordagem.

Algoritmo derivaInstanciaEmLista(k_i, k_i^{-1})

Deriva um par de chaves j, a partir do par (k_i, k_i^{-1}), baseando-se na estratégia de derivação em lista.

1: $(k_j, k_j^{-1}) \leftarrow \Delta(k_i, k_i^{-1}, 1)$

2: retorne (k_j, k_j^{-1})

Algoritmo 6.2: Algoritmo de derivação de chaves utilizando a estratégia de derivação em lista.

6.2.4.1 Derivação em Árvore

Uma alternativa às abordagens de derivação em estrela e lista é a derivação de chaves na forma de uma árvore. Nesta, as instâncias compõem uma árvore n-ária, a partir do par inicial (k_0, k_0^{-1}), representando o nodo raiz.

A figura 6.5 ilustra a composição da árvore de derivação, onde cada instância derivada possui 2 pares de chaves filhas, bem como 3 níveis distintos de profundidade.

Segundo a figura 6.5, é possível visualizar, no interior de cada nodo, o identificador de cada instância. Este identificador deverá ser utilizado em conjunto com a função Δ, para garantir a unicidade das instâncias. A ilustração mostra ainda os diferentes níveis da árvore, através da profundidade $p = \{0..3\}$, com $p = 0$ representando a raiz da árvore.

Para organizar a distribuição dos identificadores ao longo das derivações, cada par de chaves da hierarquia deverá conhecer seu identificador i, o grau n da árvore, além de uma terceira variável, $0 \leq c < n$. Esse contador representa o número de chaves filhas já produzidas a partir da chave i. A fórmula 6.7 poderá então ser utilizada para o cálculo do índice do par a ser derivado:

$$j = (i \times n) + c + 1 \tag{6.7}$$

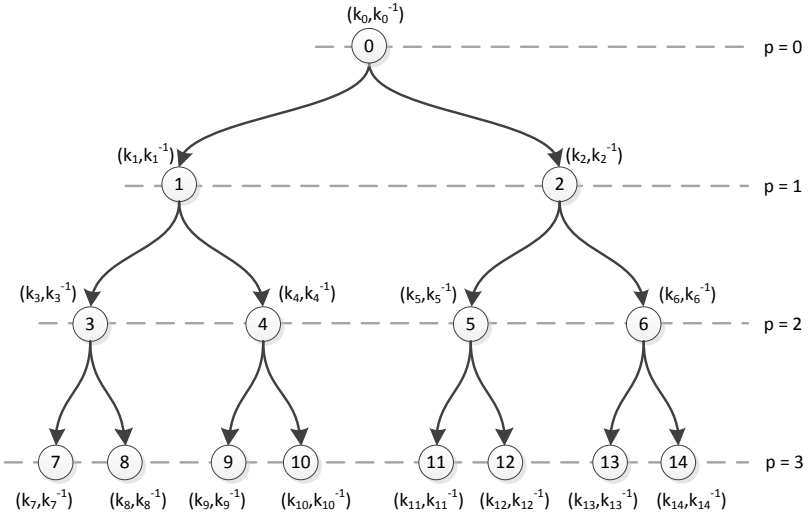

Figura 6.5: Replicação de chaves na forma de uma árvore binária, de profundidade $p = 3$.

Como um exemplo prático, para a árvore de chaves derivadas apresentada na figura 6.5, de grau $n = 2$, a definição dos identificadores das chaves filhas, a partir do par (k_4, k_4^{-1}) dar-se-ia da seguinte forma:

$$c = 0 \rightarrow j = (4 \times 2) + 0 + 1 = 9$$
$$c = 1 \rightarrow j = (4 \times 2) + 1 + 1 = 10$$

O algoritmo 6.3 descreve o procedimento de derivação de chaves para estrutura hierárquica com base em uma árvore *n-ária*.

A principal diferença entre o método de replicação em árvore e os métodos anteriores, consiste no fato de que, neste, cada instância derivada terá também a capacidade de gerar novas instâncias filhas, sem prejudicar a unicidade de cada uma delas.

É possível também definir, durante a geração do par de chaves (k_0, k_0^{-1}), além do grau n da árvore, sua profundidade p. Do ponto de vista da derivação de chaves, no momento da geração inicial estará sendo definido o número máximo de instâncias possíveis que comporão o par de chaves (k, k^{-1}).

Com base na pré-definição dos valores supracitados, o número máximo de pares nd, é obtido através da fórmula 6.8:

$$nd = \frac{n^{p+1} - 1}{n - 1} \tag{6.8}$$

Apesar de não ser necessário, restringir o tamanho máximo do con-

Algoritmo derivaInstanciaEmArvore(k_i, k_i^{-1}, i, n, c)

Deriva um par de chaves j, a partir do par (k_i, k_i^{-1}), fazendo uso do identificador i para a composição do índice do novo par. O algoritmo recebe ainda o grau n da árvore, além do número de chaves já derivadas a partir desta instância, c.

1: $(k_j, k_j^{-1}) \leftarrow 0$

2: $j \leftarrow 0$

3: se $c < n$ faça

4: $\quad j \leftarrow (i * n) + c + 1$

5: $\quad l \leftarrow j - i$

6: $\quad (k_j, k_j^{-1}) \leftarrow \Delta(k_i, k_i^{-1}, l)$

7: $\quad c \leftarrow c + 1$

8: fim se

9: retorne $(k_j, k_j^{-1}), j$

Algoritmo 6.3: Algoritmo de derivação de chaves segundo na forma de uma árvore *n-ária*.

junto de instâncias que compõem um par de chaves é uma garantia adicional para seus custodiantes, que garantirão um número limite para as instâncias derivadas.

Propostas as abordagens de derivação das instâncias de pares de chaves que comporão o par (k, k^{-1}), a próxima seção analisará os impactos da adoção da técnica sobre as diferentes fases do ciclo de vida de chaves criptográficas, prevendo a utilização de MSCs para sua gestão.

6.2.5 Ciclo de Vida com Suporte a Múltiplas Instâncias de uma Chave

A adoção do conceito de par de chaves como um conjunto de múltiplas instâncias de pares, tornam necessárias algumas mudanças na forma de gerir o ciclo de vida das chaves criptográficas. Esta seção detalhará a aplicação da proposta em um novo modelo de gestão do ciclo de vida das chaves privadas, que modificará inclusive a forma como estas são empregas na proteção de informações.

Ao londo desta seção serão analisados os impactos da adoção da abordagem proposta, sobre cada uma das fases do ciclo de vida de chaves criptográficas, conforme descrito no capítulo 2.

6.2.5.1 Geração de Par de Chaves

Para permitir o suporte a múltiplas chaves ao longo de todo o ciclo de vida, deverá ser escolhida, nesta fase, uma estratégia de derivação a ser adotada. Embora o procedimento de derivação de chaves só venha a ser utilizado em fases subsequentes, um par de chaves deverá manter a mesma estratégia durante toda a sua vida, como forma de evitar colisões. Logo, é importante, já na fase de geração, que uma delas seja selecionada.

Com a escolha da estratégia desejada, o par de chaves (k, k^{-1}) poderá ser gerado. Nesta fase, ele será representado somente pela instância inicial (k_0, k_0^{-1}), obtida a partir da função de geração G.

É importante ainda, no momento ou mesmo antes da geração da chave criptográfica, a composição do perfil de custódia do par de chaves, distribuindo-se a responsabilidade sobre a mesma.

6.2.5.2 Armazenamento

O armazenamento do material recém criado deverá ser protegido física e logicamente, A proteção física consiste nas proteções providas pelos dispositivos criptográficos, conforme descrito no capítulo 4. A proteção lógica consiste em utilizar uma chave gerada durante a criação do perfil de custodiantes para criptografar a chave privada gerada. Dessa forma garante-se que somente o perfil poderá ter acesso a esta chave.

6.2.5.3 Distribuição

A distribuição de uma chave privada, tal como nos modelos de gestão atuais, deverá primar pelo rígido controle sobre o material sensível. Como o conceito de chave criptográfica, no modelo proposto, estende-se a um conjunto de instâncias, esse controle deverá ser aplicado a cada uma destas.

Na fase de distribuição, a chave deverá ser registrada, ou seja, ligada a uma entidade. Essa ligação dar-se-á através da publicação da instância k_0^{-1}, de acordo com a técnica em uso. Utilizando-se o padrão X.509, por exemplo, esta instância deverá ser publicada em um certificado digital, de forma análoga a que ocorre nos modelos de gestão tradicionais.

A seção 6.2.5.4, a seguir, trará maiores detalhes acerca do processo de validação de assinaturas com base no certificado digital contendo a instância inicial da chave pública k^{-1}.

6.2.5.4 Utilização

Relembrando as principais finalidades de uma chave assimétrica, de acordo com o capítulo 2, temos a assinatura digital, onde um resumo criptográfico de um dado qualquer é cifrado com uma chave privada, garantindo a integridade, autenticidade e a autoria do conteúdo. Uma segunda finalidade das chaves é no provimento de sigilo, onde a chave pú-

blica é utilizada para criptografar os dados na íntegra, de forma que só a entidade mantenedora da respectiva privada poderá recuperar o acesso ao conteúdo. Todas as outras finalidades de utilização das chaves, como autenticação, transporte, entre outras, utilizam um dos dois procedimentos acima descritos.

Com a abordagem proposta, o processo de assinatura não precisará sofrer alterações práticas. A mudança será apenas conceitual, uma vez que múltiplas instâncias de uma mesma chave poderão participar dos processos de assinatura. Para assinar determinado conteúdo, qualquer uma das instâncias da chave privada k podem ser empregadas. Todavia, uma vez se tratando de várias instâncias distintas, um atributo adicional precisará ser incluído à assinatura: o identificador único da instância utilizada para produzi-la.

O esquema da figura 6.6 ilustra o processo de criação da assinatura digital, de forma simplificada, com suporte a múltiplas instancias de uma mesma chave criptográfica k.

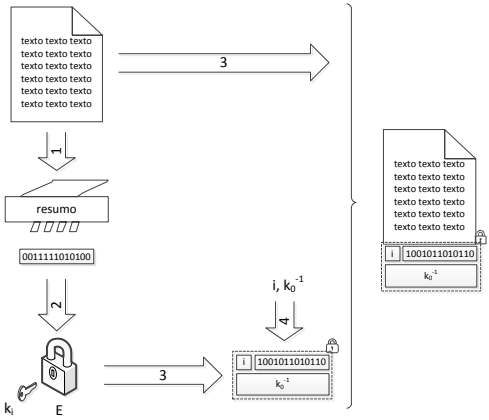

Figura 6.6: Processo simplificado de assinatura digital com suporte a múltiplas instâncias derivadas da chave k_0.

Segundo o modelo apresentado na figura 6.6, no passo 1 o documento é sintetizado, através de uma função resumo, para ser subsequentemente assinado, no passo 2, com uma das instâncias da chave k, representadas por k_i. Para tanto, é utilizada a função de cifração E. Para o modelo básico proposto, incorporam a assinatura: a chave pública k_0^{-1}, o resumo criptográfico da informação, bem como o identificador i da instância utilizada no processo de criação, conforme ilustrado nos passos 3 e 4.

Com base nessas informações, apensadas ao documento, será possível então a verificação da integridade e autenticidade da informação nele contida.

No processo de verificação de uma assinatura realizada pela instân-

cia k_i, a entidade buscando validar a autenticidade da informação deverá decompor a assinatura, de forma a recuperar as informações nela constantes. De posse do resumo assinado s, do identificador i da instância utilizada, bem como da chave pública k_0^{-1}, ela será então capaz de executar o algoritmo proposto para verificação da assinatura, ilustrado na tabela 6.4.

Algoritmo verificaAssinatura(d, s, k_0^{-1}, i)

Processo de validação de uma assinatura digital de um documento d, com suporte a múltiplas instâncias de uma mesma chave. O método recebe como parâmetros, além do documento, um resumo assinado s, a instância inicial k_0^{-1}, além do identificador da instância utilizada para produzir a assinatura k_i. O algoritmo utiliza a primitiva criptográfica Ω para o estabelecimento da cadeia de confiança.

1: $k_i^{-1} \leftarrow \Omega(k_0^{-1}, i)$

2: $h_1 \leftarrow hash(d)$

3: $h_2 \leftarrow D(s, k_i^{-1})$

4: retorne $h_1 = h_2$

Algoritmo 6.4: Algoritmo de validação suportando assinaturas geradas por múltiplas instâncias de uma mesma chave criptográfica.

O algoritmo 6.4 utiliza como entradas as informações contidas na assinatura para efetuar a derivação, com a função Ω, necessária à obtenção da chave pública k_i^{-1}. Uma vez obtida a inversa, torna-se possível a verificação da autenticidade do conteúdo, através da função D. Como resultado, o algoritmo retorna um valor booleano representando a validade, ou invalidade, da assinatura verificada.

A maior vantagem do método proposto reside justamente na possibilidade de identificar a instância i, utilizada na assinatura do documento. Assim, em caso de qualquer evidência de fraude, será possível identificar com precisão a origem do problema, propriedade intitulada como rastreabilidade das múltiplas instâncias da chave privada.

O procedimento de cifragem e decifragem das informações, contudo, precisarão de cuidados adicionais. Como no método de cifragem dos dados, a instância da chave pública disponível para cifragem será k_0^{-1}, será necessário saber de antemão o número de instâncias já derivadas a partir do par inicial (k_0, k_0^{-1}). Tais operações, contudo, estão fora do escopo deste trabalho, sendo elencadas como proposta para trabalhos futuros.

6.2.5.5 Backup e Restauração

O backup e a restauração de chaves criptográficas são os processos mais beneficiados pela adoção do modelo baseado na derivação de chaves. O uso da função Δ na derivação de novas instâncias, seguindo a estratégia definida durante a fase de distribuição, permitem o mapeamento preciso de cada uma das componentes de uma determinada chave k.

Em termos práticos, cada instância da chave deverá estar armazenada em módulos criptográficos distintos, o que garante maior disponibilidade da chave, bem como possibilita atendimento a altas demandas. Com o rastreamento proporcionado pelas funções Δ e Ω, é possível determinar qual das instâncias foi utilizada em uma determinada assinatura, permitindo assim o rígido controle sobre o ciclo de vida da chave.

A figura 6.7 ilustra o esquema de backup baseado na abordagem de derivação de uma chave (k, k^{-1}). O esquema considera que o perfil de custodiantes já tenha sido pré-estabelecido em cada um dos MSCs, através de um protocolo de acordo de chaves (RESCORLA, 1999).

De acordo com a figura, pode-se perceber a distribuição das instâncias, entre os MSCs que gerenciarão o par de chaves (k, k^{-1}). A ilustração ainda mostra o controle do número de cópias, através de um contador de derivações c.

6.2.5.6 Indisponibilidade e Comprometimento de Chaves

Um dos pontos mais críticos da gestão do ciclo de vida de uma chave, diz respeito ao seu comprometimento. Módulos criptográficos, conforme visto no capítulo 4, são dispositivos voltados a minimizar esses riscos, contando com monitoramentos e estratégias de defesa contra possíveis ataques. Modelos de gestão, entretanto, devem prever o comprometimento de uma chave privada, provendo formas de evidenciar e mitigar qualquer tentativa de acesso indevido. Adicionalmente, deve-se também fornecer uma forma de revogar as chaves privadas comprovadamente comprometidas.

Além do risco de comprometimento das chaves, como qualquer dispositivo eletrônico, módulos criptográficos estão sujeitos a falhas. A falha de um MSC pode levar à indisponibilidade das chaves gerenciadas. Como nas estratégia de derivação de chaves em estrela o procedimento de derivação é realizado apenas sobre a instância inicial de um par de chaves, é importante que este par seja copiado, por motivos de contingência. Este procedimento de cópia, contudo, restringe-se a um único módulo criptográfico, o que facilita o controle sobre este procedimento. Mesmo que seja feita uma cópia de segurança, esta poderá ficar inativa até que haja necessidade de restauração, o que não compromete o processo de auditoria. Adicionalmente, pode-se convencionar que a instância inicial, utilizada na derivação das demais, não seja utilizada para outro fim, servindo unicamente como geradora de novas instâncias.

Já no esquema de derivação em lista, o ponto mais crítico diz res-

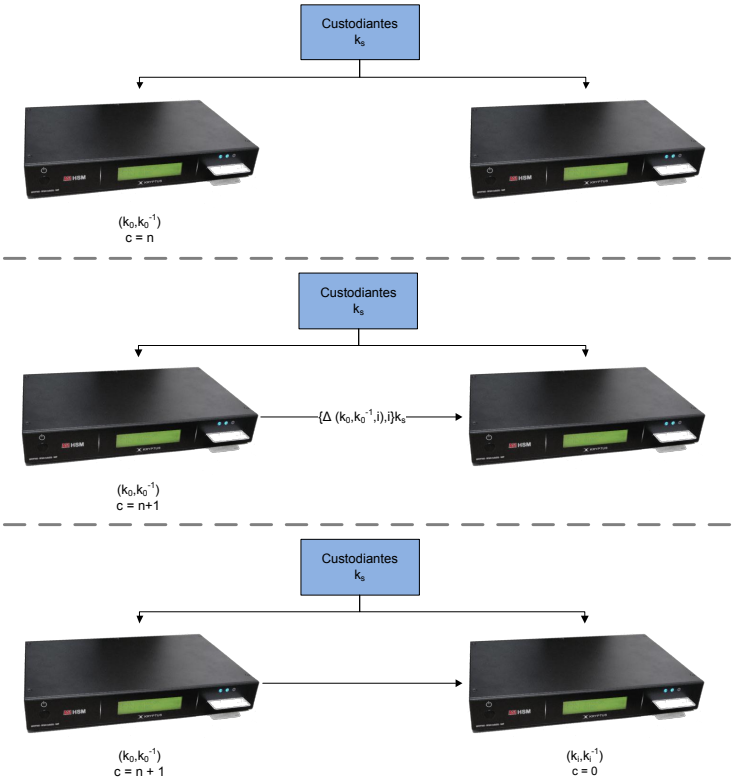

Figura 6.7: Esquema de backup baseado no modelo de derivação de instâncias de uma mesma chave.

peito à indisponibilidade da última instância derivada, o que impossibilitaria a derivação de novas instâncias. É possível, contudo, criar um método de recuperação desta, a partir de instâncias anteriores. Assim, em caso de falha do MSC contendo a última chave derivada, torna-se possível a recuperação da capacidade de gerar novas componentes do grupo.

O esquema de derivação em árvore, por sua vez, torna mais flexível a recuperação do sistema frente à indisponibilidade de uma das instâncias, por falha no MSC. Na derivação em árvore, cada instância tem a capacidade de gerar novas componentes do grupo. Portanto, em caso de falha de um dos MSCs gerenciando uma chave k, qualquer um dos outros poderá ser utilizado para derivar uma nova instância, sem qualquer prejuízo.

Com relação ao comprometimento de uma das instâncias da chave privada k, os métodos tradicionais de revogação devem ser aplicados. Em termos práticos, a revogação de uma chave privada está relacionada à inclusão da chave pública correspondente em uma lista de revogação. No

padrão X.509, essa lista é conhecida como Lista de Certificados Revogados (LCR), pois a revogação ocorre sobre o certificado digital em si. O padrão permite definir a causa da revogação através do atributo *revocationReason*. Em caso de comprometimento da chave privada, o campo deve assumir o valor *keyCompromise* (COOPER et al., 2008).

Já com o uso do PGP, a revogação é realizada sobre a chave privada propriamente dita. Para revogação da chave, utiliza-se uma senha de revogação, além da chave privada correspondente. Caso um dos objetos necessários encontre-se indisponível, torna-se impossível revogar a chave pública correspondente. Para evitar esse problema, pode ser gerado antecipadamente um certificado de revogação (GARFINKEL, 1994). O certificado contém a chave pública da entidade a quem o par de chaves pertence, sendo assinado pela própria chave privada a ser revogada. O certificado de revogação serve como uma apólice de seguro em caso de uma catástrofe (perda da senha ou da chave privada), devendo dessa forma ser armazenado em local seguro e de acesso restrito.

No método proposto, com a existência de múltiplas instâncias da chave privada, a revogação da instância inicial k_0^{-1} é também suficiente para revogar uma determinada chave k. De fato, o procedimento é transparente, quando aplicado ao padrão X.509. Uma vez encontrado o identificador do certificado entre a lista de revogados, nenhuma das instâncias deverá ser considerada confiável. Métodos mais eficientes, que possibilitem a revogação de uma, ou um subconjunto de instâncias, são deixados como uma proposta de trabalhos futuros. Com o uso de PGP, pode-se alterar o algoritmo de validação do certificado de revogação para aceitar as múltiplas instâncias de k, ou simplesmente assinar o certificado de revogação com a instância k_0.

6.2.5.7 Destruição de Chaves

Como cada instância compondo a chave k pode ser unicamente identificada, o procedimento de destruição definitiva do material consistirá na destruição de todas as componentes do grupo. Controlando-se o número máximo de chaves derivadas, pode-se basear o procedimento de destruição em recibos de destruição, assinados por cada uma das instâncias destruídas. Procedimentos de destruição das múltiplas instâncias, contudo, estão fora do escopo deste trabalho, sendo deixados como proposta para trabalhos futuros.

6.2.6 Controle sobre o Uso

Uma vez que preserva-se a unicidade de cada instância da chave privada, o modelo de derivação permite contemplar uma política de custódia de chaves, onde é possível definir se cada instância do par (k, k^{-1}) será gerida por um único ou vários grupos distintos de custodiantes. Exemplos de possíveis políticas são:

Custódia Centralizada: A política de custódia centralizada consiste na atribuição do controle total, sobre todas as instâncias da chave gerenciada, a um único grupo de custodiantes;

Custódia Distribuída: Com a adoção desta política, cada nova instância deverá ser gerida por um grupo de custodiantes distinto e independente, distribuindo-se a responsabilidade pela custódia da chave privada;

Custódia Compartilhada: Esta política baseia-se no princípio da custódia distribuída, mantendo-se contudo o poder dos custodiantes da chave predecessora sobre as chaves derivadas a partir dela. Dessa forma, cria-se níveis de controle, onde o nível superior detém controle também sobre todas as instâncias presentes em níveis inferiores.

A figura 6.8 ilustra as possíveis políticas de custódia, aplicadas a cada uma das estratégias de derivação apresentadas na seção 6.2.

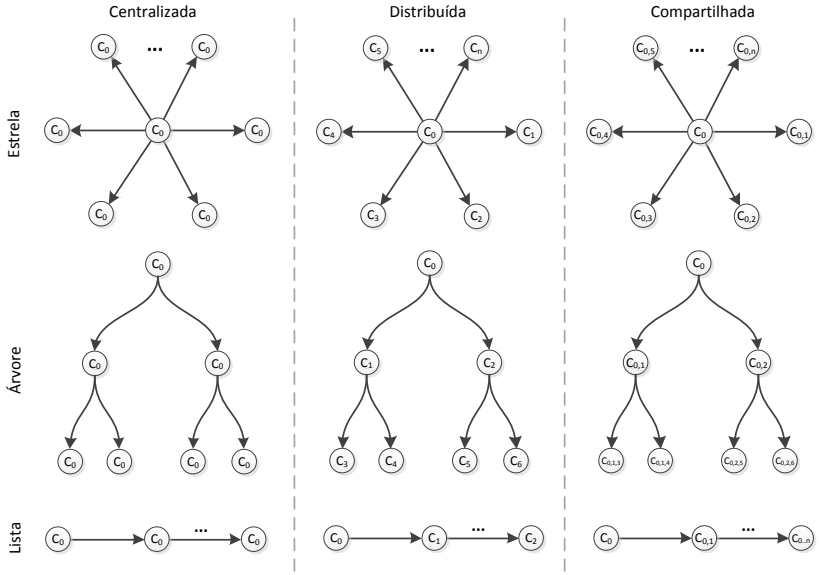

Figura 6.8: Políticas de controle sobre chaves, para cada uma das estratégias de derivação apresentadas.

6.2.7 Aplicabilidade da Abordagem Proposta

Conforme apresentado neste capítulo, a abordagem proposta apresenta várias vantagens quando comparada aos métodos existentes. Todavia, não há registros da existência de tais primitivas criptográficas, que possibilitem a aplicação da técnica. Este trabalho limita-se a apresentar

as possíveis aplicações dessas primitivas, caso suas existências venham a ser comprovadas. A pesquisa de funções matemáticas que possibilitem a derivação de chaves criptográficas assimétricas é uma das propostas de trabalhos futuros.

6.3 CERTIFICAÇÃO DE MÚLTIPLAS CHAVES ASSIMÉTRICAS

A proposta de certificação de chaves baseia-se na existência de mais de uma chave privada para cada AC de uma infraestrutura. Essas chaves encontrar-se-ão ligadas, por sua vez, não por uma primitiva de derivação, mas por intermédio de uma cadeia de confiança. Tal cadeia será formada por uma infraestrutura interna de chaves, ou seja, uma ICP interna ao contexto de cada uma das ACs da ICP.

A figura 6.9 ilustra uma AC Raiz, segundo o modelo de certificação de múltiplas chaves.

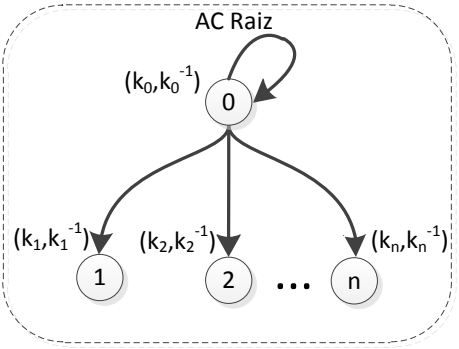

Figura 6.9: AC composta por uma ICP interna.

Segundo o modelo proposto pela figura 6.9, a AC Raiz possui um par de chaves inicial, para o qual emitirá um certificado auto assinado. Em uma ICP convencional, este certificado digital auto assinado define a âncora de confiança da infraestrutura, sendo utilizado para emitir certificados digitais para ACs subordinadas. No modelo proposto, ele continua sendo a âncora de confiança da ICP, mas será utilizado para emitir certificados para outras chaves da própria AC. Na figura, os certificados são representados pelas arestas do grafo.

Para que os certificados pertençam à mesma entidade, devem possuir todos os campos relacionados ao seu titular idênticos. Os únicos campos divergentes, entre os certificados, seriam seus números seriais, suas chaves públicas, suas validades, bem como as extensões relacionadas a esta chave pública.

O par de chaves inicial, auto assinado, deverá ser utilizado somente para credenciar novas chaves, o que o torna menos exposto a riscos des-

necessários, dado que, conforme descrito no capítulo 2, quanto mais utilizada, maior o valor de uma chave criptográfica, bem como maior o risco de comprometimento. É importante também efetuar uma cópia de segurança da chave raiz de cada AC, tendo em vista que seu comprometimento implicará na impossibilidade de credenciamento de novas chaves. Como estas chaves serão utilizadas com menor frequência, o impacto sobre a auditoria do sistema será mínimo.

A figura 6.10 expande o conceito para o âmbito de uma ICP, onde cada uma das autoridades é formada também por um conjunto de chaves distintas, operando como se fossem uma única, por pertencerem a uma mesma entidade.

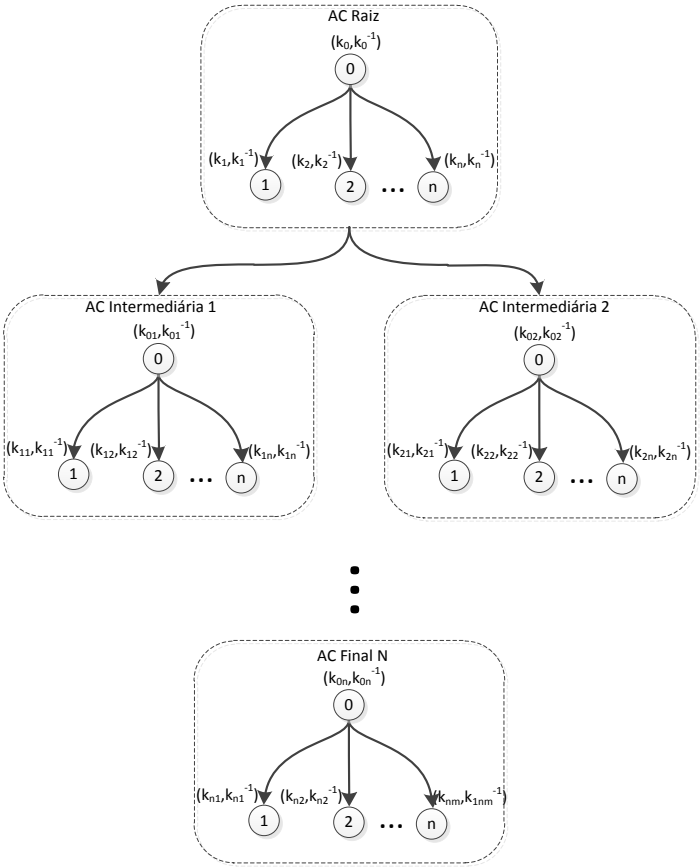

Figura 6.10: ICP baseada em autoridades compostas por múltiplas chaves credenciadas a uma mesma entidade.

6.3.1 Procedimentos para Certificação de Múltiplas Chaves

Para o estabelecimento da infraestrutura interna a uma autoridade certificadora, os conjunto de passos abaixo deverá ser seguido. Os passos referem-se à criação de uma AC Raiz.

1. Um novo par de chaves deverá ser gerado em um primeiro MSC;

2. Um certificado digital X.509 auto assinado deverá ser emitido para a AC detentora do novo par de chaves;

3. Para cada nova chave desejada, um novo MSC deverá ser utilizado para gerar um par de chaves;

4. Uma requisição de certificado deverá ser gerada para cada um dos pares de chaves gerados no passo anterior, com os mesmos dados de titular contidos no certificado auto assinado;

5. A partir das requisições, novos certificados deverão ser emitidos pela AC, para a própria AC;

Para a criação de uma AC intermediária ou final, basta substituir a emissão do certificado auto assinado, no passo 2, pela emissão de um certificado assinado por qualquer uma das chaves da AC imediatamente superior.

Com a execução dos passos acima, o resultado serão múltiplas chaves distintas, operando como uma única chave, pois levarão a uma mesma âncora de confiança. A abordagem é interessante, pois permite que cada chave privada de uma AC possa ser gerenciada de forma independente, pertencendo contudo a uma mesma entidade. A disponibilidade da chave é garantida através da existência de múltiplas instâncias.

Como a carga da demanda por assinaturas é distribuída entre as n chaves da AC, o atendimento à altas demandas é garantido. A unicidade das chaves é parcialmente mantida, dada a necessidade de contingência das chaves primárias de cada AC. Por fim, a rastreabilidade das chaves é mantida através dos certificados digitais, sendo que em caso de comprometimento de uma das chaves.

Uma vantagem desta técnica, com relação à de derivação de chaves, consiste no fato de que uma das chaves pode ser revogada independentemente, através da emissão de uma LCR, sem afetar as demais. No método de derivação de chaves, esse tipo de revogação é possível, mas demandaria customizações no formato da LCR.

Outro benefício do emprego da certificação quando comparado à derivação de chaves, é a possibilidade do uso da técnica para sigilo de conteúdo digital sem qualquer necessidade de customização.

O procedimento, contudo, implica em modificações não só técnicas, mas também políticas nas regras de uma ICP. Um exemplo são as

ICPs que utilizam a extensão *BasicConstraints*, onde a profundidade máxima do caminho de certificação é limitado pela restrição *pathLenConstraint*. Para a utilização do modelo, esta restrição deverá passar de uma profundidade *p* para *2p*, tendo em vista que cada nível da cadeia de certificação desdobrar-se-á em 2. Outro impacto significativo com a adoção da técnica será sobre o custo de validação, que também será dobrado.

Uma chave secundária de uma AC, adicionalmente, precisará ter um cripto período menor que a primária. Os cripto períodos de cada chave deverão ser refletidos na validade de seus certificados.

O método de certificação de múltiplas chaves acaba por transferir para os sistemas de gestão de certificados, a responsabilidade pela gerência de múltiplas chaves assimétricas de uma mesma entidade. Embora não tenham sido preparados para essa tarefa, os sistemas de gestão de certificados possuem mais recursos para esta tarefa, tendo em vista que certificados digitais contam com procedimentos mais difundidos e padronizados para emissão, renovação, revogação e encadeamento de seus certificados.

6.4 ANÁLISE COMPARATIVA ENTRE OS MODELOS DE GESTÃO

Os modelos de gestão de chaves baseados na cópia de chaves privadas permitem o balanceamento de carga sobre uma chave, bem como garantem sua disponibilidade, através da replicação do material. Essa replicação, todavia, acaba por tornar os procedimentos de auditoria demasiadamente complexos e suscetíveis a erros.

O método de derivação de chaves proporciona o mapeamento das múltiplas instâncias de uma chave privada, transformando o conceito de um par de chaves em algo mais abrangente como um conjunto de instâncias. Esta abordagem possibilita um modelo que ofereça as mesmas propriedades dos métodos atuais, sem contudo comprometer a unicidade das chaves, bem como a rastreabilidade entre as múltiplas instâncias.

Como o modelo de derivação é baseado em novas primitivas criptográficas, até que estas passem a existir, faz-se necessária ainda uma maneira de minimizar a dependência existente entre a escalabilidade dos modelos de gestão com o procedimento de cópia das chaves.

Com o método de certificação de chaves, os problemas dos métodos tradicionais são contornados. Partindo-se de técnicas já existentes, é possível garantir a disponibilidade e o atendimento a altas demandas, minimizando-se a necessidade de replicação de chaves, com a vantagem de permitir o isolamento de uma chave privada comprometida ou indisponível, sem afetar as demais.

6.4.1 Rastreabilidade

Com relação à propriedade de rastreabilidade, os métodos tradicionais deixam a cargo dos procedimentos de auditoria o controle das múltiplas instâncias de uma chave. Como os registro de uso de um MSC

podem ser apagados ou mesmo interpretados de forma incorreta, a rastreabilidade de chaves copiadas é fraca. Há MSCs, como o apresentado no capítulo 4, inclusive, que não possuem um subsistema de registro interno de uso. Nesses sistemas a rastreabilidade de chaves é perigosamente comprometida, sempre que a propriedade de unicidade é quebrada.

Com o procedimento de derivação, o controle das múltiplas instâncias torna-se independente dos registros de uso do módulo, tendo em vista que, a partir da instância inicial, existe um caminho único a ser precorrido, de forma que qualquer instância derivada possa ser unicamente identificada. Os procedimentos de auditoria, contudo, continuam sendo importantes e não devem ser negligenciados, pois auxiliam a determinação do momento exato em que uma chave foi gerada, utilizada, bem como outros eventos envolvendo o uso de MSCs.

Em caso de uso indevido de uma das instâncias da chave para o processamento de uma assinatura, será possível identificar, através do próprio conteúdo assinado, qual a chave utilizada, o que apontará com precisão o módulo criptográfico a ser auditado.

O método de certificação de chaves, tal como o de derivação, possibilita o rastreamento das múltiplas chaves, através dos respectivos certificados, emitidos para uma mesma AC.

Uma segunda vantagem do método de certificação, sobre o método de derivação, tem relação com a assinatura digital, que não precisa conter nenhum atributo diferenciado.

O método de certificação deixa a desejar com relação ao desempenho, pois degrada a performance da validação de assinaturas pela metade. Ainda, a técnica está intimamente ligada ao padrão X.509, não se aplicando a outras abordagens de criptografia de chaves públicas.

6.4.2 Unicidade

Com os métodos atuais a unicidade das chaves privadas não é mantida. Isso impacta diretamente em um maior esforço de auditoria, que consiste em procedimentos manuais, passíveis de erros.

No método de derivação, cada instância da chave é univocamente identificada, mantendo-se assim a unicidade. Dessa forma, há um maior controle das múltiplas instâncias, sendo que com a análise da própria assinatura é possível identificar a utilizada para assinar o conteúdo.

No método de certificação de chaves, são utilizados pares distintos para cada ambiente operacional simultâneo da AC. Dessa forma, reduz-se o procedimento de cópia apenas para as chaves primárias da AC. Como a assinatura de certificados será realizada pelas chaves secundárias, estas poderão ser unicamente identificadas, mantendo a rastreabilidade.

6.4.3 Controle sobre as Múltiplas Instâncias

O controle sobre as múltiplas instâncias de uma mesma chave é mais eficiente quando é possível identificar cada uma delas, simplificando

o processo de auditoria. Os métodos atuais não possibilitam a identificação das múltiplas cópias de uma chave, dificultando o controle sobre o uso das inúmeras cópias possíveis.

Com o método de derivação de chaves, tal identificação inequívoca é possível. Através da certificação de chaves, apesar das chaves serem distintas e independentes, torna-se trivial também a identificação de cada uma delas. Ambos os métodos destacam-se quando comparados aos tradicionais, onde a cópia é idêntica à chave original, impossibilitando sua distinção.

6.4.4 Comprometimento de uma Chave

Em caso de comprometimento de uma das cópias da chave privada, nos métodos atuais, deve-se imediatamente revogar a chave pública k^{-1}.

No método de derivação, um procedimento semelhante deverá ser executado, contudo visando a revogação da chave pública inicial k_0^{-1}. Um trabalho mais aprofundado pode ser feito, visando buscar uma forma mais eficiente de revogação de chaves, onde seja possível revogar apenas um subconjunto das instâncias da chave.

No método de certificação, esta revogação parcial de uma das instâncias, ou isolamento desta, é possível através da revogação do certificado da chave comprometida.

6.4.5 Quadro Comparativo Resumido

A tabela 6.5 apresenta o quadro resumido com as diferenças apresentadas, de acordo com as propriedades de disponibilidade, rastreabilidade, unicidade, controle sobre o uso e comprometimento de chaves privadas.

	Atuais	**Derivação**	**Certificação**
Disponibilidade	Cópia	Derivadas	Secundárias
Rastreabilidade	Logs	Identificador	Certificado
Unicidade	Não	Sim[1]	Parcial
Controle	Difícil	Assinatura	Assinatura
Comprometimento	Revogar k^{-1}	Revogar k_0^{-1}	Isolamento

Tabela 6.5: Comparativo entre os métodos propostos e atuais

Segundo a tabela 6.5, pode-se perceber a diferença entre os métodos atuais, de derivação e certificação.

Com relação à disponibilidade, os métodos atuais são baseados em cópias do material. O método de derivação baseia-se em múltiplas deri-

vadas de um par inicial. Já o método de certificação utiliza chaves secundárias independentes, ligadas por um certificado digital.

No quesito rastreabilidade, surge o principal diferencial das abordagens propostas, que independem do registro de eventos, identificando unicamente as múltiplas instâncias de uma chave.

Com relação à unicidade da chave privada, surge mais uma vantagem dos modelos propostos. Com a cópia da chave privada, os métodos atuais não proporcionam a unicidade. Já com a derivação, tal unicidade pode ser mantida, conquanto que se utilize uma estratégia que não demande a cópia da instância inicial. Na certificação de chaves, a cópia é reduzida à instância primária, que pode ser melhor controlada e utilizada apenas para certificar as demais.

O controle sobre as chaves é possível, nos métodos propostos, através da própria assinatura gerada. Nos modelos atuais, o controle sobre o uso da chave torna-se difícil à medida que o número de cópias aumenta, uma vez que é baseado na análise dos registros de eventos dos múltiplos ambiente operacionais.

O comprometimento de uma das cópias, tanto nos motelos atuais quanto na derivação, implicarão na revogação de todas as demais instâncias. Com o método de certificação, entretanto, é possível isolar a instância comprometida, minimizando o impacto de revogação

6.5 CONSIDERAÇÕES DO CAPÍTULO

Este capítulo abordou duas estratégias distintas para a gestão de múltiplas instâncias de uma chave criptográfica, que proporcionam o rastreamento de cada uma delas.

A primeira estratégia de gestão de chaves é baseada não na simples cópia destes materiais, mas sim na existência de primitivas criptográficas que permitam a derivação de chaves assimétricas. Com base nessa derivação, foi proposto um modelo de gestão que suportasse a gestão de múltiplas instâncias de uma mesma chave, de forma simultânea. Este modelo tem como premissas básicas o atendimento à alta demanda, o privilégio à disponibilidade da chave privada, bem como preservação da unicidade e rastreabilidade entre as múltiplas instâncias.

O segundo método, baseado na certificação das chaves, possibilita as mesmas garantias que a técnica de derivação, com algumas vantagens, como a possibilidade de uso para o sigilo de informações, bem como a possibilidade de revogação isolada de uma das chaves de uma AC.

Como desvantagens, o método de certificação apresenta um maior custo na validação do caminho de certificação, tendo em vista que a profundidade do caminho será duplicada. Alternativamente, pode-se usar esta técnica somente nas ACs que necessitem de múltiplas cópias operacionais da mesma chave.

Com a exposição das propostas, realizou-se uma análise comparativa entre os métodos propostos e os existentes na literatura, buscando

elencar as vantagens e desvantagens de cada um deles. Nela, a abordagem de certificação de chaves mostrou resultados satisfatórios, apresentando aplicação prática, por não necessitar de nenhum subterfúgio adicional aos já existentes atualmente.

O próximo capítulo apresentará as considerações finais acerca das abordagens apresentadas, bem como indicará possíveis trabalhos futuros, com base neste.

7 CONSIDERAÇÕES FINAIS

Como descrito nos capítulos 1 e 2, não há na literatura qualquer referência quanto à gestão do ciclo de vida de múltiplas cópias de uma chave criptográfica assimétrica. Os modelos de gestão, todos, consideram o ciclo de vida de uma única chave e, com respeito à disponibilidade, sugerem algum esquema de cópia de segurança. No entanto, encontram-se na prática vários exemplos de ambientes onde os custodiantes de uma chave criptográfica são obrigados a replicá-la com o intuito de garantir, não somente a contingência, mas também a demanda por assinaturas digitais. A falta de um modelo de gestão faz com que cada chave seja tratada, mesmo sendo a mesma, como independente, tal que o modelo de gestão existente possa ser usado. Isso, todavia, impede o correto rastreamento da chave, e dificulta sobremaneira a auditoria. E muitas vezes, geram-se relatórios não condizentes com o verdadeiro uso da chave, o que pode implicar em análises incorretas e levar a conclusões e tomadas de decisão indevidas.

Para minimizar as dificuldades desses esquemas práticos, os custodiantes lançam mão de cerimônias rígidas, que buscam registrar, passo a passo, cada interação com a chave. O problema é que tais ceriômias envolvem o homem, e os registros são normalmente feitos para o homem, sem qualquer preocupação com formalismos, o que impede que esses possam fazer parte de relatórios gerados por sistemas automáticos de auditoria.

E para complicar ainda mais, essas cerimônias, devido principalmente à fragilidade do controle de rastreamento das chaves, são feitas em ambientes seguros denominados Sala Cofre. Essas salas cofre, como se sabe, tem alto custo de implantação de de manutenção.

Assim, tais esquemas não podem ser adotados pela maioria das aplicações, o que fragiliza as soluções existentes na presença de cópias usáveis das chaves.

Para uma melhor compreensão dos problemas apontados na gestão segura do ciclo de vida de chaves assimétricas, foco do trabalho, foram revisitados os conceitos tradicionais, relacionados às infraestruturas de chaves públicas, no capítulo 3. Para um melhor entendimento acerca do funcionamento dos módulos criptográficos, utilizados na proteção das chaves de Autoridades Certificadoras, uma visão geral de suas funcionalidades foi também demonstrada, no Capítulo 4. Por fim, comprovou-se que, pelo simples fato de serem considerados seguros, ou mesmo possuindo certificados atestando as boas práticas de projeto e desenvolvimento, módulos criptográficos podem ocultar vulnerabilidade que, por sua vez, poderão expor o material sensível de suas chaves gerenciadas. Com isso, evidenciou-se a necessidade de modelos claros e robustos, que prevejam o mal uso e a falha de tais dispositivos, permitindo em sua essência um controle mais rígido das várias instâncias de uma mesma chave.

Tendo em visa a utilização de chaves criptográficas assimétricas no contexto das ICPs, foi apresentado, no capítulo 5, um estudo de como cópias de uma mesma chave vem sendo em parte geridas.

Um novo modelo foi então proposto, no capítulo 6, com base em novas primitivas criptográficas que possibilitassem a identificação precisa de cada uma das instâncias de uma mesma chave privada, suportando assim a gestão de múltiplas instâncias simultâneas.

Ainda no capítulo 6, um segundo modelo, baseado em um conjunto de procedimentos sobre a arquitetura de uma ICP hierárquica, possibilitou alcançar os objetivos almejados com a técnica de derivação, sem contudo necessitar de novas primitivas criptográficas.

Os modelos propostos neste trabalho são mais completos que aqueles usados atualmente, pois permitem a identificação inequívoca de cada uma das chaves criptográficas, evitando a sobrecarga sobre os procedimentos de auditoria.

Como principais contribuições deste trabalho, pode-se destacar:

- A análise crítica das diferentes propostas existentes na literatura, relacionadas à gestão segura do ciclo de vida de chaves criptográficas;
- A confirmação de que, por mais que sejam utilizados módulos criptográficos certificados por normas de segurança reconhecidas, tais sistemas estão sempre suscetíveis a falhas de segurança, podendo expor as chaves gerencias a riscos;
- O desenvolvimento de um protótipo, capaz de avaliar a segurança de módulos criptográficos, através da tentativa de extrair uma chave privada gerenciada, na forma de texto claro;
- O relato de possíveis abordagens, que vêm sendo utilizadas como forma de gerir múltiplas cópias de uma mesma chave privada em autoridades certificadoras;
- A proposta de um modelo de gestão baseado em novas primitivas criptográficas de derivação de chaves assimétricas, cujas propriedades foram definidas, tornando possível a garantias de disponibilidade e atendimento a altas demandas, sem contudo abrir mão das propriedades de unicidade, rastreabilidade, bem como do rígido controle dos custodiantes sobre o material sensível;
- A proposta de um conjunto de procedimentos, que contornem os problemas apresentados pelas propostas existentes, sob as mesmas premissas do modelo baseado na derivação de chaves, sem contudo demandar a criação de novas primitivas.

7.1 TRABALHOS FUTUROS

Dentre os possíveis trabalhos futuros, ressalta-se:

1. A pesquisa e o desenvolvimento de métodos matemáticos ou computacionais para a derivação de chaves com as propriedades que foram propostas no capítulo 6;

2. A proposta de uma forma otimizada de revogação de chaves, segundo a proposta de derivação de chaves, que permita o isolamento de uma instância comprometida;

3. A proposição de novas estratégias de derivação, bem como uma análise comparativa com as demais;

4. A definição de um protocolo para destruição segura de chaves, segundo o modelo de derivação, que permita atestar que todas as instâncias de uma chave foram destruídas com segurança;

5. A formulação de uma proposta que preveja a possibilidade de aplicar o modelo de gestão, baseado nas novas primitivas, no sigilo de dados e informações.

REFERÊNCIAS

ATENIESE, G.; TSUDIK, G. Some open issues and new directions in group signatures. In: . [S.l.]: Springer-Verlag, 1999. p. 196–211.

BARKER, E.; BARKER, W.; BURR, W. *NIST SP 800-57: Recommendation for Key Management - Part 1: General (Revised)*. 2007. 1–142 p.

BARKER, E. et al. *SP 800-130: A Framework for Designing Cryptographic Key Management Systems*. [S.l.]: National Institute of Standards and Technology, maio 2010. 01–88 p.

BARKER, E.; ROGINSKY, A. *NIST SP 800-131: Recommendation for the Transitioning of Cryptographic Algorithms and Key Lengths*. [S.l.]: National Institute of Standards and Technology, 2010. 27 p.

BISHOP, M. *Computer Security: Art and Science*. Addison Wesley, 2002. 1136 p. ISSN 1540-7993. Disponível em: <http://ieeexplore.ieee-.org/lpdocs/epic03/wrapper.htm?arnumber=1203217>.

BORTOLOZZO, M. et al. Attacking and Fixing PKCS #11 Security Tokens. In: *CCS '10: Proceedings of the 17th ACM Conference on Computer and Communications Security*. Chicago: ACM, 2010. p. 260–269.

BORTOLOZZO, M. et al. Secure your PKCS#11 token against API attacks! In: *ASA '09: 3rd International Workshop on Analysis of Security APIs*. [S.l.: s.n.], 2009.

CACHIN, C.; CHANDRAN, N. A Secure Cryptographic Token Interface. In: *2009 22nd IEEE Computer Security Foundations Symposium*. Ieee, 2009. p. 141–153. ISBN 978-0-7695-3712-2. Disponível em: <http:/-/ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5230619>.

CHAUM, D.; HEYST, E. V. Group signatures. In: *Proceedings of the 10th annual international conference on Theory and application of cryptographic techniques*. Berlin, Heidelberg: Springer-Verlag, 1991. (EUROCRYPT'91), p. 257–265. ISBN 3-540-54620-0. Disponível em: <http://portal.acm.org/citation.cfm?id=1754868.1754897>.

CLULOW, J. On the Security of PKCS # 11. In: *CHES '03*. [S.l.]: Springer-Verlag, 2003. p. 411–425.

COOPER, D. et al. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. IETF, 2008. RFC 5280 (Proposed Standard). (Request for Comments, 5280). Disponível em: <http://www.ietf.org/rfc/rfc5280.txt>.

DELAUNE, S.; KREMER, S.; STEEL, G. Formal Analysis of PKCS #11. In: *CSF '08: Proceedings of the 2008 21st IEEE Computer Security Foundations Symposium*. Washington, DC, USA: IEEE Computer Society, 2008. p. 331–344.

DIERKS, T.; RESCORLA, E. *The Transport Layer Security (TLS) Protocol Version 1.1*. [S.l.]: IETF Network Working Group, 2006. 1–87 p.

DIFFIE, W.; HELLMAN, M. E. New Directions in Cryptography. In: *IEEE Transactions on Information Theory*. [S.l.: s.n.], 1976. p. 644–654.

ELLISON, C. *Ceremony Design and Analysis*. 2007. 1–17 p.

FEDERAL COMMUNICATIONS COMMISSION. *Code of Federal Regulations, Title 47, Part 15, Subpart B, Unintentional Radiators, Digital Devices*. http://frwebgate.access.gpo.gov/cgi-bin/get-cfr.cgi?TITLE=47&PART=15&SECTION=101&YEAR=1998&TYPE=TEXT: FEDERAL COMMUNICATIONS COMMISSION, October 1998.

FERGUSON, N.; SCHNEIER, B. *Practical Cryptography*. New York, NY, USA: John Wiley & Sons, Inc., 2003. ISBN 0471223573.

FUMY, W.; LANDROCK, P. Principles of key management. *IEEE Journal on Selected Areas in Communications*, v. 11, n. 5, p. 785–793, jun. 1993. ISSN 07338716. Disponível em: <http://ieeexplore.ieee.org-/lpdocs/epic03/wrapper.htm?arnumber=223881>.

GARFINKEL, S. *PGP: Pretty Good Privacy*. First edition. [S.l.]: O'Reilly & Associates, 1994. 432 p.

HOUSLEY, R.; POLK, T. *Planning for PKI: Best Practices Guide for Deploying Public Key Infrastructure*. New York, NY, USA: John Wiley & Sons, Inc., 2001. ISBN 0471397024.

INTERNATIONAL TELECOMMUNICATION UNION. *The Directory: Public-key and attribute certificate frameworks*. [S.l.]: ITU-T, 2005. 1–174 p.

KERCKHOFFS, A. La cryptographie militaire. *Journal des sciences militaires*, IX, n. 1, p. 5–38, 1883.

LABORATORIES, R. *Has DES been broken?* 07 de 2010. Http://www.rsa.com/rsalabs/node.asp?id=2227.

LUZ, C. P. da. *Centro de Certificação Digital : construção, administração e manutenção*. [S.l.]: Ciência Moderna, 2008. 1–338 p.

MAO, W. *Modern Cryptography: Theory and Practice*. [S.l.]: Prentice Hall, 2003. 648 p. ISBN 0-13-066943-1.

MARTINA, J. E. *Projeto de um Provedor de Serviços Criptográficos Embarcado para Infra-estrutura de Chaves Públicas e suas Aplicações*. 1–167 p. Tese (Mestrado) — Universidade Federal de Santa Catarina, 2005.

MENEZES, A. J.; OORSCHOT, P. C. V.; VANSTONE, S. A. *Handbook of Applied Cryptography*. [S.l.]: CRC Press, 1996. 816 p.

MOULDS, R. *Key Management for Dummies*. ncipher special edition. [S.l.]: Wiley Publishing, Inc, 2008.

MYERS, M. et al. *Internet X.509 Certificate Request Message Format*. IETF, mar. 1999. RFC 2511 (Proposed Standard). (Request for Comments, 2511). Obsoleted by RFC 4211. Disponível em: <http://www.ietf.org/rfc/rfc2511.txt>.

National Institute of Standards and Technology. *FIPS 140-2: SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES*. National Institute of Standards and Technology, 2002. 1–69 p. Disponível em: <http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf>.

nCipher Corporation Ltd. *nCipher Security World White Paper*. nCipher Corporation Ltd., 2001. White Paper. Disponível em: <http:-//iss.thalesgroup.com/Resources/White Papers/˜/media/Files/White Papers/ncipher_security_world_wp.ashx>.

NIST. *Data Encryption Standart*. [S.l.], 1977.

NYSTROM, M.; KALISKI, B. PKCS #10: Certification Request Syntax Specification Version 1.7. nov. 2000. Disponível em: <http://www.ietf.org/rfc/rfc2986.txt>.

RESCORLA, E. *Diffie-Hellman Key Agreement Method*. IETF, jun. 1999. RFC 2631 (Proposed Standard). (Request for Comments, 2631). Disponível em: <http://www.ietf.org/rfc/rfc2631.txt>.

RIVEST, R. L.; SHAMIR, A.; ADLEMAN, L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems Public-Key Cryptosystems. *Communications of the ACM*, v. 21, n. 2, p. 120–126, 1978.

RSA Laboratories. *PKCS #8: Private Key Information Syntax Standard*. [S.l.]: RSA Laboratories, 1993. 1–5 p.

RSA Laboratories. *PKCS #11 v2.10: Cryptographic Token Interface Standard*. RSA Laboratories, 1999. Disponível em: <http://www.rsa-.com/rsalabs/node.asp?id=2133>.

RSA Laboratories. *PKCS #12 v1.0: Personal Information Exchange Syntax*. RSA Laboratories, 1999. 1–23 p. Disponível em: <ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-12/pkcs-12v1.pdf>.

RSA Laboratories. *PKCS #1 v2.1: RSA Cryptography Standard*. RSA Laboratories, 2002. Disponível em: <http://www.rsa.com/rsalabs/node-.asp?id=2125>.

RSA LABORATORIES. *RSA Laboratories Web Site*. Acesso em 08 de 2010. Disponível em: <http://www.rsa.com/rsalabs>.

Safenet Inc. *ProtectToolkit C: Administration Manual*. [S.l.]: Safenet Inc, 2008. 180 p.

SCHNEIER, B. *Applied Cryptography, Second Edition: Protocols, Algorthms, and Source Code in C*. [S.l.]: John Wiley & Sons, Inc., 1996.

SOUZA, T. C. S. D.; MARTINA, J. E.; CUSTóDIO, R. F. Audit and backup procedures for Hardware Security Modules. In: *SBSEG '07: VII Simpósio Brasileiro de Segurança em Computação*. [S.l.: s.n.], 2007.

The Common Criteria Recognition Agreement. *Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model July 2009 Revision 3 Final Foreword*. The Common Criteria Recognition Agreement, 2009. Disponível em: <http://www-.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R3.pdf>.

The Common Criteria Recognition Agreement. *Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components July 2009 Revision 3 Final Foreword*. The Common Criteria Recognition Agreement, 2009. Disponível em: <http://www-.commoncriteriaportal.org/files/ccfiles/CCPART3V3.1R3.pdf>.