

Capítulo

2

Modelos de Criptografia de Chave Pública Alternativos

Denise Goya¹, Mehran Misaghi², Vilc Rufino^{1,3} e Routo Terada¹

¹Departamento de Ciência da Computação – IME–USP

²Instituto Superior Tupy, Sociedade Educacional de Santa Catarina – IST–SOCIESC

³Centro de Coordenação de Estudos da Marinha em São Paulo – CCEMSP–MB

{dhgoya, vilc, rt}@ime.usp.br, mehran@sociesc.org.br

Abstract

In this short course some difficulties about public key infrastructure are reviewed and it is showed how to reduce them with four alternative models in assymetric cryptography: identity-based, self-certified, certificateless and certificate-based. This study starts with the conceptual analysis of each model, and presents advantages and disadvantages, possible applications and use contexts. A selection of protocols and code portions illustrate the behavior of models and introduce to the students useful tools to develop new applications.

Resumo

Neste minicurso são revisadas algumas das dificuldades relacionadas à infraestrutura de chaves públicas (ICP) convencional e como elas podem ser minimizadas por meio do uso de quatro modelos alternativos de criptografia assimétrica: baseado em identidade, auto-certificado, sem certificados (certificateless) e baseado em certificado (certificate-based). A partir da análise conceitual de cada modelo, são expostas vantagens e desvantagens, discussões sobre possíveis aplicações e contextos de uso. Uma seleção de protocolos e pequenos trechos de código ilustram o funcionamento dos modelos e introduzem aos alunos ferramentas úteis para o desenvolvimento de novas aplicações.

¹Projeto Fapesp n° 2008/06189-0

2.1. Introdução

A criptografia de chave pública surgiu com a busca por soluções para dois problemas intrinsecamente relacionados com o modelo de criptografia simétrica: como distribuir uma chave secreta e como autenticar alguém, com garantia de irretratabilidade?

Ambos problemas foram brilhantemente abordados em [Diffie e Hellman 1976] e nasceu um novo paradigma para a criptografia, a de chave pública. Nesse modelo, também chamado de assimétrico, todo usuário possui um par de chaves, uma pública e outra secreta. A chave pública pode ser divulgada por meio de um canal público e ser usada para cifrar mensagens. A chave secreta é usada para decifrar e não precisa ser transmitida. Naquela ocasião, Diffie e Hellman apresentaram um protocolo de acordo de chave secreta sobre canal inseguro e definiram os princípios de funcionamento da assinatura digital.

É natural que um novo paradigma traga consigo novos problemas. Com a criptografia de chave pública, não foi diferente. Um problema central nesse modelo é o de legitimação da chave pública. Como garantir que uma chave pública pertence, de fato, a alguém? Uma solução que se tornou prática comum é a de implantação de uma infraestrutura de chaves públicas (ICP, ou PKI – *Public Key Infrastructure*), cuja marca são os certificados digitais de chave pública. Tal solução, entretanto, embute algumas dificuldades, como as resumidas abaixo:

- processos complexos de implantação e manutenção da infraestrutura;
- custos de emissão, distribuição e armazenamento de certificados;
- custos para recuperar e validar certificados;
- dificuldades com revogação de certificados.

Há pelo menos dois caminhos para minimizar essas dificuldades: modificar o modelo de ICP (que não é nosso foco) e modificar o modelo de criptografia de chave pública. Esta última abordagem dá origem ao que chamamos de **modelos alternativos**.

O objetivo deste texto é apresentar quatro modelos de criptografia de chave pública alternativos que dispensam a ICP convencional ou que a simplificam. São eles, o modelo baseado em identidade, o de chave pública autocertificada, o modelo sem certificados e o baseado em certificado. Este último, apesar do nome sugerir o modelo tradicional com certificados X.509, é alternativo por variar na forma de geração, distribuição e uso do certificado. Optamos por adotar as nomeações dadas originalmente pelos autores, simplesmente as traduzindo respectivamente de *identity-based*, *self-certified public key*, *certificateless* e *certificate-based*.

Mostraremos, na seção 2.2, que pequenas modificações nos parâmetros que concebem o modelo de criptografia de chave pública podem induzir um modelo diferente e criar um novo paradigma de criptografia assimétrica. Conforme dissemos anteriormente, novos paradigmas tendem a criar novos problemas. Às vezes, os problemas não são propriamente novos, mas são inexistentes no modelo convencional com ICP. Portanto, pretendemos confrontar propriedades, vantagens e desvantagens de cada modelo, para que

seja possível uma análise mais apurada e realista por aqueles que intencionam implantar novos criptosistemas de chave pública, minimizando efeitos colaterais de uma ICP.

Na seção 2.3, apresentaremos detalhes de construção, alguns algoritmos e protocolos selecionados e possíveis aplicações. Seguem posteriormente, na seção 2.4, comparações gerais entre os modelos, considerações sobre implementação, sugestões de trabalhos futuros e conclusões.

Antes de tudo, revisamos na próxima subseção alguns conceitos em criptografia de chave pública, importantes para a compreensão dos modelos alternativos. Descrições mais detalhadas de fundamentos de criptografia e segurança podem ser obtidas em livros como [Mao 2003] e [Terada 2008].

2.1.1. Conceitos Preliminares

Em criptografia de chave pública, todo usuário tem um par de chaves (s, P) . A chave pública P está matematicamente relacionada com a chave secreta s , de forma que:

- P é calculada a partir de s , mas
- a partir do valor de P , é computacionalmente inviável descobrir s .

Desse modo, podemos escrever genericamente que $P = f(s)$, onde f é uma função injetora. Às vezes, a função f inclui parâmetros do sistema ou outros dados. Por exemplo, na cifragem de ElGamal, $P = g^s$, onde g é um parâmetro do sistema. Eventualmente, f pode parecer complexa e, em implementações concretas, pode até ser codificada por um algoritmo probabilístico ou s ser calculada a partir de P . Mas, por simplicidade, basta pensar que a chave pública é função da secreta.

Ora, se a chave pública é apenas resultado de um cálculo, como comprovar que esse cálculo está única e exclusivamente associado a alguém que conheça o valor secreto s que o gerou? Em criptosistemas de chave pública, se não houver uma garantia de legitimidade da chave pública, não haverá segurança alguma.

Uma solução para a legitimação de chaves públicas que nos interessa revisar aqui é a baseada em certificados digitais. Para se garantir que (s, P) pertence a um certo usuário com identidade ID , introduzimos uma entidade confiável que chamaremos de Autoridade de Confiança (AC), ou autoridade certificadora. AC tem seu próprio par de chaves (s_{AC}, P_{AC}) e emite certificados digitais que atestam que um certo valor P pertence a uma entidade identificada por ID . Todos que confiarem na autoridade, usarão P certos de que estarão se comunicando com ID .

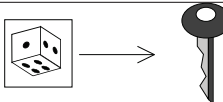
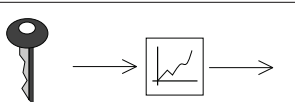
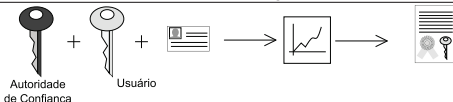
Um **certificado digital** (no modelo convencional) nada mais é do que um documento que contém, na essência, dados sobre ID , sua chave pública P , dados sobre a autoridade AC , além de uma assinatura digital de AC que assegura a validade dos dados do certificado. Portanto, a assinatura no certificado é uma **garantia** de que a chave P está associada com ID . Podemos interpretar essa assinatura como uma função que tem na entrada a chave pública P , a identidade ID e a chave secreta da autoridade s_{AC} .

Uma **ICP** nada mais é do que um agregado de hardware, software, pessoal especializado e procedimentos, para gerenciar todo o ciclo de vida dos certificados: da sua

criação, distribuição até sua renovação ou revogação. Alguns aspectos do funcionamento dessa infraestrutura serão revistos ao longo do texto, conforme se fizer necessário.

Os principais atributos da criptografia de chave pública, no modelo convencional sobre ICP, são sintetizados na tabela 2.1. Durante a análise dos modelos alternativos, veremos que a variação desses atributos modifica sensivelmente as propriedades do modelo.

Tabela 2.1. Atributos do modelo convencional de criptografia de chave pública.

Chave secreta	Chave pública	Garantia
s	$P = f(s)$	$f(ID, P, s_{AC})$
		
escolhida pelo usuário (ou pela autoridade)	calculada pelo usuário (ou pela autoridade)	calculada pela autoridade, exclusivamente

Para cifrar uma mensagem para o dono de P ou para verificar uma assinatura dele, os usuários usam o valor da chave P , obtido do certificado (ver a figura 2.1).

Para decifrar uma mensagem destinada ao dono de P ou para que este crie uma assinatura, é necessária a chave secreta s correspondente.

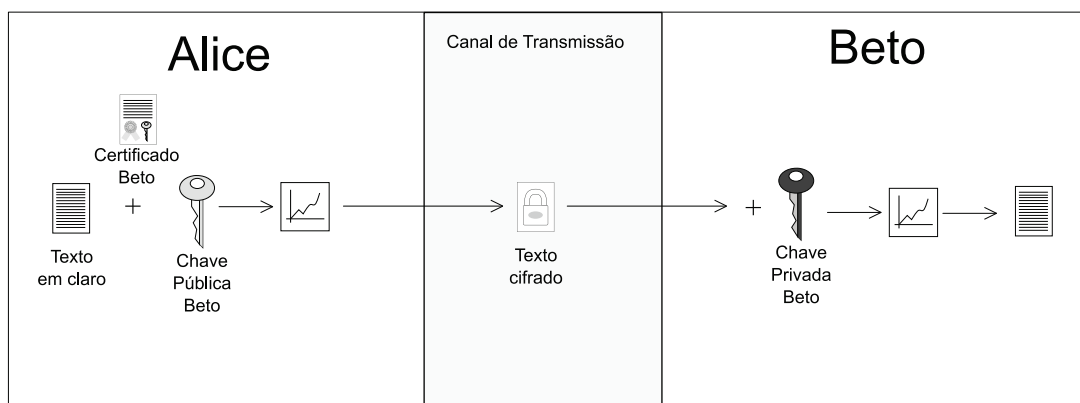


Figura 2.1. Cifrando no modelo convencional com ICP

Quando dissemos que a autoridade do sistema deve ser uma entidade de confiança, não entramos no mérito sobre o que significa confiança. Em [Girault 1991], são definidos os três níveis abaixo, que indicam o grau de credibilidade que devemos depositar na autoridade.

Nível 1: a autoridade conhece (ou calcula facilmente) chaves secretas dos usuários; pode personificar qualquer entidade sem ser detectada;

Nível 2: a autoridade desconhece (ou dificilmente calcula) chaves secretas dos usuários; pode personificar qualquer entidade, gerando falsas chaves públicas, sem ser detectada;

Nível 3: a autoridade desconhece (ou dificilmente calcula) chaves secretas dos usuários; pode personificar qualquer entidade, porém é detectada.

No modelo com ICP e certificados de chave pública X.509, a autoridade alcança nível 3, o mais desejável. Os modelos alternativos, muitas vezes caem em níveis mais baixos e, conseqüentemente, a autoridade tem mais poder e os usuários precisam confiar mais nela.

2.2. Modelos Alternativos: Conceitos e Propriedades

Nas próximas quatro subseções, descreveremos os princípios de funcionamento dos modelos alternativos, as premissas, propriedades, pontos fortes e fracos de cada um, de forma conceitual, sem citarmos algoritmos nem protocolos, por enquanto.

2.2.1. Criptografia de Chave Pública Baseada em Identidade

A ideia central da criptografia de chave pública baseada em identidade é muito simples. Se é um problema o fato da chave pública ser um valor numérico sem sentido explícito, por que não calcular a chave secreta a partir da pública, que passa a ser uma cadeia de caracteres com algum significado? Em [Shamir 1984], foi proposto que a chave pública fosse a própria identidade do usuário, como nome, endereço de email, CPF, número de telefone celular, endereço IP, número serial de dispositivos eletrônicos, etc.

Se a chave pública é predeterminada (igual à identidade), como, então, calcular a chave secreta? A resposta a essa questão vem junto com as primeiras premissas de segurança do modelo: existe uma *AC*, com as seguintes atribuições principais:

- Criar e manter a guarda segura de uma chave mestra secreta s_{AC} ;
- Identificar e registrar todos usuários do sistema;
- Calcular as chaves secretas dos usuários;
- Entregar as chaves secretas de forma segura (com sigilo e autenticidade).

Em 1984, Shamir descreveu o modelo e algoritmos para assinatura digital. Foram necessárias quase duas décadas até que fossem descobertos algoritmos de cifragem eficientes e demonstrados seguros, para que o modelo baseado em identidade despertasse um interesse renovado nos pesquisadores e na indústria.

Para fins de comparação, na tabela 2.2, vê-se que a chave secreta é calculada em função do segredo da autoridade do sistema e da identidade do usuário. Para uma *f* conveniente, é inviável recuperar a chave mestra a partir dos valores de *ID*. E somente a autoridade é capaz de gerar as chaves secretas, de modo que a própria secreta *s* é a garantia de que o uso de *ID* funcionará nas operações criptográficas envolvendo o dono dessa identidade.

Para cifrar uma mensagem para o dono de *ID* ou para verificar uma assinatura de *ID*, os usuários usam a identidade *ID* mais os parâmetros públicos do sistema, que incluem a **chave pública da autoridade** (ver a figura 2.2).

Tabela 2.2. Atributos do modelo de criptografia de chave pública baseada em identidade.

Chave secreta	Chave pública	Garantia
$s = f(ID, s_{AC})$	ID	s
calculada pela autoridade e compartilhada com o usuário	escolhida pelo usuário ou formatada pela autoridade	

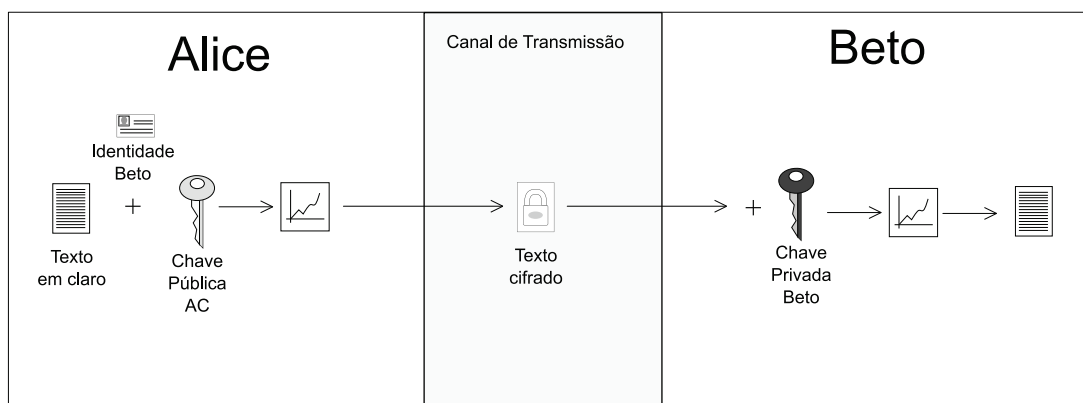
Para decifrar uma mensagem destinada a ID ou para que este crie uma assinatura, é necessária a chave secreta de ID .

Vantagens

O modelo baseado em identidade é atraente por apresentar várias vantagens interessantes. A primeira é que a chave pública pode, em grande parte dos casos, ser memorizável por humanos. Muito diferente da chave pública convencional, que costuma ser uma cadeia binária com centenas ou milhares de bits. A identidade pode ser informada pelo próprio usuário aos seus parceiros e não há a obrigatoriedade de manutenção de diretórios de chaves.

Se houver credibilidade na autoridade do sistema e no conteúdo identificador da identidade, a certificação ocorre implicitamente. Tornam-se dispensáveis os certificados digitais e, mais ainda, a infraestrutura que os gerencie. Como consequência, muitos procedimentos existentes sobre uma ICP deixam de existir nos sistemas baseados em identidade.

Para que seja possível visualizar a economia de tempo de processamento, de custos de armazenamento e de transmissões de dados, vamos relembrar, por exemplo, como é, de maneira geral, uma operação de cifragem com ICP. Para Beto cifrar uma mensagem para Alice, antes de tudo, ele deve obter o certificado que fora emitido para a Alice (con-

**Figura 2.2. Cifrando no modelo baseado na identidade**

sultando um diretório público ou a própria Alice). De posse do certificado, Beto precisa verificar o período de validade e a assinatura contida no certificado. A verificação da assinatura é um processo que, às vezes, percorre todo o caminho de certificação das autoridades certificadoras envolvidas na hierarquia, até se chegar à autoridade certificadora raiz. Se até este ponto, nada falhou, Beto pode guardar o certificado da Alice para futuros usos. Entretanto, antes de cada uso, Beto precisa consultar uma autoridade de validação, para verificar se o certificado não foi revogado (quase sempre, um procedimento de consulta a um servidor que esteja on-line). Estando o certificado válido e não revogado, Beto extrai a chave pública da Alice, cifra a mensagem e transmite.

No modelo baseado em identidade, basta que os parâmetros do sistema sejam autênticos (no modelo com ICP também é preciso que os parâmetros do sistema sejam autênticos). Satisfeita essa condição, as operações de Beto para cifrar com base na identidade se resumem a obter o identificador da Alice, cifrar e enviar (considerando-se que revogação de identidade é tratada como explicado adiante).

Uma peculiaridade do modelo baseado em identidade é o fato da chave pública poder ser usada antes do cálculo da chave secreta. Assim, é possível cifrar uma mensagem para quem ainda não se registrou junto à autoridade do sistema nem possui chave secreta de decifragem. Por contraste, no modelo com certificados, o usuário deve primeiramente se registrar e obter seu certificado, para então poder receber uma mensagem cifrada sob sua chave pública.

Desvantagens

O preço a pagar por todas essas vantagens é uma série de desvantagens, que, em alguns contextos, podem ser muito críticas.

A primeira desvantagem, que é característica de sistemas baseados em identidade “puros” (isto é, sem modificações que tentam minimizar ou eliminar os efeitos dessa característica) é a **custódia de chaves**. Conforme explicado anteriormente, a autoridade do sistema tem a capacidade de gerar as chaves secretas de todos os usuários sob sua responsabilidade. Isso implica que a autoridade alcança nível 1 de confiança, na definição de [Girault 1991]. Consequentemente, pode decifrar quaisquer textos cifrados a que tiver acesso (se puder identificar a identidade do destinatário). Também pode assinar em nome de qualquer usuário e não há como garantir irretratabilidade. Portanto, é fundamental que a autoridade do sistema seja confiável o bastante para que ações de bisbilhotagem ou de falsificação como essas sejam controláveis.

A propriedade de custódia de chaves, referenciada por *key escrow* nos textos em inglês, nem sempre é indesejável. Dentro de uma empresa, por exemplo, se todos os documentos e dados sensíveis são cifrados pelo funcionário que o gerou, a diretoria pode ter acesso à decifragem em caso de morte ou desligamento do funcionário. Quando houver necessidade de monitoria do conteúdo de emails cifrados, também pode ser justificável a custódia de chaves. No entanto, para a maioria das aplicações, custódia de chaves é uma desvantagem.

Outro ponto desfavorável ao modelo baseado em identidade é a necessidade de

um **canal seguro** para distribuição das chaves secretas. Se a entrega ocorrer num ambiente em rede e remotamente, há que se garantir autenticação mútua e entrega com sigilo. Para podermos comparar, no modelo sobre ICP, a autenticação frequentemente acontece durante o registro do usuário junto à autoridade de registro (muitas vezes presencial), mas não há transmissão nem compartilhamento de segredo.

Do lado do usuário, a guarda da chave secreta deve ser à prova de perdas e roubos, o que sugere o uso de mecanismos adicionais para proteção da chave, como dispositivos de hardware, senha ou desafio/resposta. E sempre que for necessário renovar a chave secreta, é obrigatória a interação do usuário com a autoridade do sistema. Esses aspectos de guarda e renovação parecem ser equivalentemente implementados nos modelos com ICP e baseado em identidade.

O **risco de comprometimento da chave mestra** no sistema baseado em identidade é muito alto, maior que no modelo convencional. Sobre a ICP, o comprometimento da chave secreta da autoridade potencialmente leva à criação de certificados falsos de chaves públicas. Portanto, são afetadas as operações que ocorrerem num momento posterior ao comprometimento da chave mestra. Já no modelo baseado em identidade, quando a chave mestra é roubada ou perdida, operações criptográficas do passado e do futuro estão comprometidas, para todos os usuários sob a autoridade em questão. Qualquer texto cifrado capturado poderá ser decifrado, se o intruso souber as identidades, e independentemente se a cifra foi gerada antes ou depois do vazamento da chave mestra.

Outra preocupação que se deve ter no modelo baseado em identidade é a possibilidade de **revogação de identidades**. Caso a chave secreta de um usuário seja comprometida, sua identidade deve ser revogada. Portanto, não é recomendável usar simplesmente o número do CPF ou do telefone celular, por exemplo, como identificador de usuário. Nem sempre é possível cancelá-los. Uma alternativa é concatenar ao identificador principal um período de validade, como em `JoaoSilva-deJan09aDez09`. Caso esse usuário tenha seu segredo comprometido, nova chave secreta poderia ser gerada para uma nova identidade, como `JoaoSilva-deAgo09aDez09`. No entanto, pode não ser trivial garantir que ninguém use a antiga identidade comprometida. Diminuir a granularidade do período, por um lado pode ajudar, por outro, sobrecarregará a autoridade, que terá que renovar as chaves com maior frequência (e vale lembrar que a cada renovação o canal seguro deve ser restabelecido).

Características Adicionais

Como observado por [Shamir 1984], o modelo baseado em identidade é ideal para grupos fechados de usuários, como executivos de uma multinacional ou filiais de um banco, uma vez que a sede dessas corporações podem servir como autoridade do sistema, em que todos confiam. Aplicações de pequena escala, em que os custos com implantação e manutenção de uma ICP sejam proibitivos, são candidatas ao uso do modelo baseado em identidade. Quando as desvantagens citadas anteriormente não forem críticas, as características do modelo possibilitam implementações interessantes.

Vale evidenciarmos a flexibilidade de definição da chave pública. A cadeia de caracteres associada com a identidade pode, em princípio, ser qualquer coisa. A con-

catenação de indicadores de tempo e atributos possibilitam aplicações em serviços com disponibilidade temporal ou sigilo no envio de mensagens com base em papéis, como descrito em [Misaghi 2008].

Alguns exemplos de serviços com disponibilidade temporal: documento confidencial que possa ser revelado à imprensa ou a um grupo particular, somente a partir de determinada data e hora; lances de um leilão que devem ser mantidos em segredo até o fim das negociações; ou exibição de um filme que deve ser habilitada somente dentro do período de locação contratado.

Vários trabalhos foram desenvolvidos para adaptar o modelo baseado em identidade a hierarquias de autoridades, dentre os quais, o de [Gentry e Silverberg 2002] é um dos pioneiros. Uma das vantagens em trabalhar de forma hierárquica é divisão da responsabilidade da autoridade do sistema com autoridades subordinadas. E isso traz consequências importantes. A criação das chaves secretas dos usuários pode ser delegada a níveis inferiores da hierarquia, distribuindo a carga de trabalho da entidade geradora. Além disso, o segredo de cada usuário passa a depender das chaves mestras de mais de uma autoridade e, portanto, diminui o risco do comprometimento de uma chave mestra e há maior controle sobre o uso indevido de uma chave secreta por conta da custódia.

A organização hierárquica de autoridades também viabiliza a cifragem para grupos de usuários. Mas uma generalização desse tema se deu com o trabalho de [Abdalla et al. 2006], a partir do qual a identidade pode conter caracteres curingas. Num serviço de correio eletrônico cifrado, por exemplo, `*@*.usp.br` atinge todos usuários de todos departamentos da USP; `admin@*.usp.br` diz respeito a todos administradores de sistema dentro daquela universidade.

O modelo baseado em identidade também tem sido alvo de estudos na busca por alternativas ao SSL/TLS, para aplicações na Web, como se pode ver em [Crampton et al. 2007]. Com a eliminação de certificados, simplificam-se o processo de distribuição de chaves públicas e o controle de acesso. De forma semelhante, o modelo tem sido explorado para prover segurança em várias outras áreas de aplicação, como computação em grade e redes de sensores (ver por exemplo [Lim e Paterson 2005] e [Szczechowiak et al. 2008]) e outras aplicações na subseção 2.3.2.

2.2.2. Criptografia de Chave Pública Autocertificada

O modelo de criptografia de chave pública autocertificada foi proposto inicialmente em [Girault 1991]. Aqui, a ideia é que a própria chave pública contenha uma garantia de que seu valor está associado com a identidade. Isso é possível se existir uma autoridade confiável que auxilia na geração da chave pública e se o cálculo dessa chave depender:

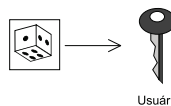
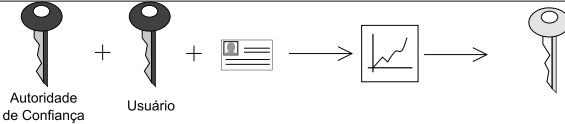
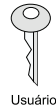
- da identidade do usuário;
- do segredo da autoridade;
- e do segredo do usuário.

A criação da chave pública autocertificada sempre ocorre a partir de um protocolo interativo entre o usuário e a autoridade. Analogamente ao modelo convencional de

criptografia de chave pública, na proposta de Girault o usuário escolhe sua própria chave secreta e a mantém em sigilo. Durante a interação, tanto o usuário quanto a autoridade precisam comprovar que conhecem um segredo (suas respectivas chaves secretas), sem revelá-lo ao outro. Desse modo, a interação pode embutir um protocolo de identificação, uma assinatura ou uma forma qualquer de conhecimento-zero.

Dependendo da construção do protocolo, a chave pública é calcula pelo usuário ou pela autoridade. Porém, em todos os casos, só o usuário conhece sua chave secreta, o que significa que não há custódia de chaves, um dos pontos fracos do modelo baseado em identidade. Os atributos desse modelo são resumidos na tabela 2.3.

Tabela 2.3. Atributos do modelo de criptografia de chave pública autocertificada.

Chave secreta	Chave pública	Garantia
s	$P = f(ID, s, s_{AC})$	P
		
escolhida pelo usuário	calculada pela autoridade ou pelo usuário, em um protocolo de prova interativa	

O uso de uma chave pública falsa impede a correta inversão da operação criptográfica. Em outras palavras, se Beto cifrar um texto com uma chave pública falsificada para Alice, ela não será capaz de decifrá-lo. Da mesma forma, se Alice assinar um documento, todos que forem enganados sobre sua verdadeira chave pública não poderão verificar essa assinatura. Isso caracteriza o que chamamos de certificação implícita.

No modelo tradicional com certificados, a validade da chave pública é explicitamente verificada antes de seu uso. Já no modelo autocertificado, isso não ocorre. A legitimidade de uma chave pública autocertificada é confirmada implicitamente quando a inversão da operação criptográfica for bem sucedida. É claro que essa propriedade nem sempre é conveniente. Em [Kim et al. 1999], por exemplo, pode ser conferida uma solução em que é possível verificar previamente a chave certificada. Na figura 2.3, vemos o uso da chave autocertificada num processo de cifragem e decifragem.

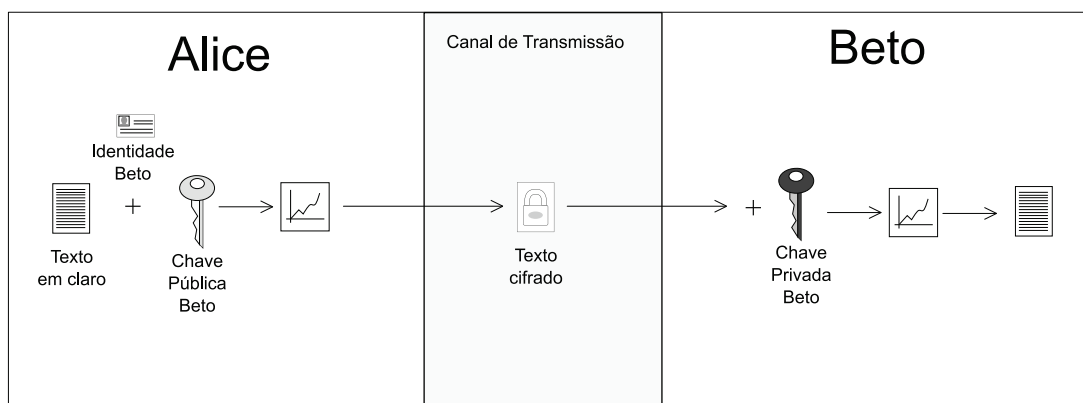


Figura 2.3. Cifrando no modelo autocertificado

Em seu trabalho, Girault mostrou duas formas de se gerar chave pública autocertificada e indicou como realizar um acordo de chaves com autenticação mútua. Em outro trabalho independente, [Günther 1989], também foi apresentado um protocolo de acordo de chaves com autenticação em que as identidades dos usuários eram usadas nos cálculos da chave negociada. A ideia de explorar a identidade para indiretamente validar a chave pública é retomada em [Al-Riyami e Paterson 2003], com outro modelo alternativo, que estudaremos na próxima subseção.

Em síntese, a **vantagem** mais importante da criptografia de chave pública autocertificada é a inexistência de certificados digitais para as chaves públicas (uma vez que a certificação acontece implicitamente) e sem recair em custódia de chaves.

Em contrapartida, surgem **desvantagens**. Relativamente ao modelo baseado em identidade, o de chave pública autocertificada é menos atrativo por requerer um repositório de chaves públicas (ou cada usuário deve informar sua chave pública aos outros).

Entretanto, o ponto mais crítico é a segurança do modelo originalmente definido em [Girault 1991]. Na época daquela publicação, as formalizações de demonstração de segurança para criptografia de chave pública começavam a ser construídas. Até então, os protocolos não possuíam demonstrações matemáticas de sua segurança; continham simplesmente alguns argumentos intuitivos sobre a dificuldade de sua quebra. Os protocolos de Girault e a maioria dos trabalhos subsequentes não apresentavam modelos formais de segurança e, portanto, não eram demonstrados seguros. Ao contrário, muitos deles foram quebrados em algum momento posterior.

Até mesmo os esquemas de [Girault 1991] sofreram abalo. No texto original, o autor alegou ter encontrado uma forma de manter o nível 3 de confiança na autoridade, sem ter que distribuir certificados digitais. Em [Saeednia 2003], no entanto, os esquemas de Girault foram rebaixados ao nível 1 (o mesmo do modelo baseado em identidade). Uma correção foi sugerida nesse mesmo artigo, porém o preço para se recuperar a classificação de nível 3 foi um alto custo computacional.

Vários pesquisadores estudaram o problema de construir esquemas baseados na chave pública autocertificada que pudessem ser demonstrados seguros. Os resultados positivos quase sempre surgiram quando foram introduzidas variações sobre a concepção original de Girault.

Essas variações, que ganharam nomes como certificado implícito ou assinatura autocertificada, são mais adequadamente enquadrados dentro de outros modelos. Breves descrições sobre as variantes mais interessantes serão dadas na subseção 2.3.3.

A chave pública autocertificada, como garantia de autenticação do usuário, não evoluiu como modelo independente. Isto é, não se chegou, pelo menos até o momento, a um conjunto de primitivas básicas que garantam confidencialidade, autenticidade e integridade, todas demonstravelmente seguras.

2.2.3. Criptografia de Chave Pública sem Certificados

O modelo de criptografia de chave pública sem certificados foi apresentado originalmente em [Al-Riyami e Paterson 2003]. Os autores buscavam uma forma de eliminar a custódia de chaves do modelo baseado em identidade, mais especificamente do protocolo de

cifragem de [Boneh e Franklin 2001]. Combinando ideias deste último com o modelo de chave pública autocertificada de [Girault 1991], chegou-se a um modelo intermediário entre o baseado em identidade e o convencional com certificados.

O modelo resultante deixa de ser baseado em identidade, pois há uma chave pública diferente do identificador do usuário. Porém, por haver uso da identidade, ocorre a certificação implícita, sem que haja necessidade de distribuição e armazenamento de certificados digitais. Por esses motivos, costuma-se dizer que o modelo sem certificados de [Al-Riyami e Paterson 2003] reúne o melhor dos dois paradigmas: não há certificados e não há custódia de chaves.

Como nos modelos anteriores, o sem certificados também depende da existência de uma autoridade de confiança *AC*. O papel dela é, além de identificar e registrar todos os usuários, calcular parte das chaves secretas dos usuários.

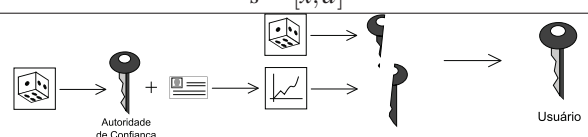
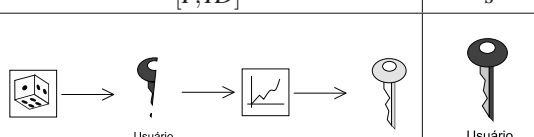
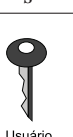
O cálculo das chaves parciais envolve a chave mestra secreta da autoridade, e a identidade do usuário. A entrega dessas chaves parciais deve acontecer de forma segura, com autenticidade e sigilo. Porém, só com o conhecimento da chave secreta parcial, não é possível decifrar nem assinar documentos. Logo não há custódia de chaves.

A chave pública de cada usuário é calculada de modo semelhante ao que ocorre no modelo convencional. Cada usuário escolhe seu segredo e gera a chave pública a partir desse segredo. A chave pública pode ser divulgada pelo próprio usuário ou é colocada em um diretório público.

A chave secreta completa é composta por duas partes: a parcial fornecida pela autoridade e o segredo do usuário. Somente com essas duas partes é possível assinar mensagens ou realizar decifragens. E o usuário é o único que conhece o segredo completo.

Na tabela 2.4, podemos visualizar os principais parâmetros envolvidos no modelo sem certificados.

Tabela 2.4. Atributos do modelo de criptografia de chave pública sem certificado.

Chave secreta	Chave pública	Garantia
$s = [x, d]$	$[P, ID]$	s
		
x é segredo escolhido pelo usuário $d = f(ID, s_{AC})$ é segredo parcial calculado pela autoridade e compartilhado com o usuário	$P = f(x)$ é calculada pelo usuário	

Para cifrar uma mensagem para *A* ou para verificar uma assinatura de *A*, outros usuários precisam da chave pública e da identidade de *A*. Desse modo, podemos dizer que a identidade é parte da chave pública. Dentre os parâmetros do sistema, a chave pública da autoridade é fundamental nos cálculos e é um dos dados de entrada (ver a figura 2.4).

Para criar uma assinatura de *A* é necessária a chave secreta completa de *A*, mais a

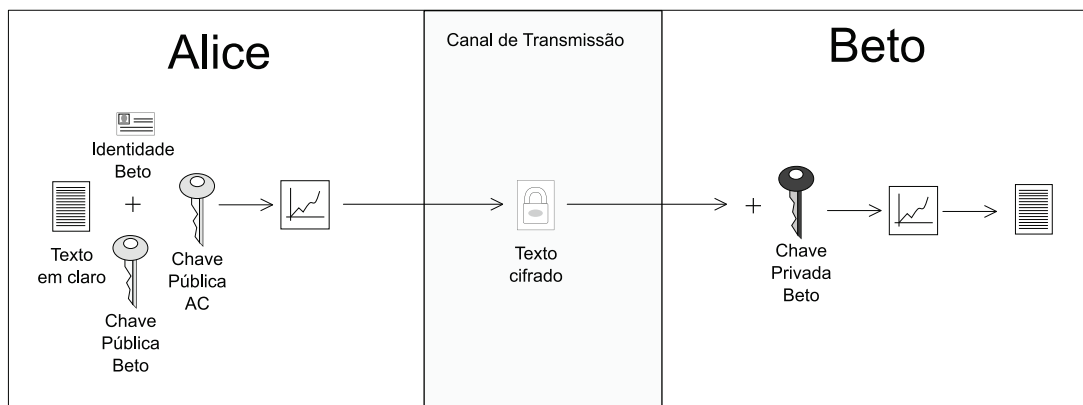


Figura 2.4. Cifrando no modelo sem certificado

identidade de A, a fim de se confirmar a certificação implícita. Para decifrar uma mensagem emitida para A, basta a chave secreta completa de A.

Na **análise de segurança** do modelo sem certificado, os autores modelaram dois tipos de adversários. O primeiro cumpre o papel de um usuário do sistema que tem o poder de substituir a chave pública de qualquer outro usuário; ele no entanto desconhece a chave mestra da autoridade. Esse tipo de adversário foi definido, pois, como não há certificados para validarem explicitamente as chaves públicas antes de seu uso, não há garantia alguma da veracidade delas.

O segundo tipo de adversário representa a própria autoridade do sistema que, embora honesta, pode tentar bisbilhotar as mensagens cifradas.

Um esquema é considerado seguro no modelo sem certificados quando é acompanhado de uma prova de que adversários, de qualquer dos dois tipos, alcançam probabilidade desprezível de sucesso em diferenciar entre duas cifras de duas mensagens conhecidas.

O esquema de cifragem apresentado em [Al-Riyami e Paterson 2003] foi demonstrado seguro sobre essa modelagem de dois adversários. Posteriormente, surgiram propostas de assinatura e outros protocolos variantes, com demonstrações sob os mesmos princípios desse modelo de segurança.

A criptografia de chave pública sem certificados apresenta outras **vantagens**, além da já citada desnecessidade de distribuição de certificados e eliminação da custódia de chaves.

Primeiramente, o risco do comprometimento da chave mestra é menor do que no modelo baseado em identidade. Se houver perda ou violação da chave mestra num sistema sem certificado, serão comprometidas apenas as chaves secretas parciais. Para conseguir decifrar mensagens cifradas anteriormente ao comprometimento da chave mestra, o impostor deverá também obter o segredo do usuário. E para personificar usuários ou decifrar novas mensagens, antes terá que substituir chaves públicas por valores cujo segredo associado seja escolhido pelo adversário.

Outra propriedade, que é exclusiva do modelo sem certificado, é que o processo

de renovação da chave pública pode ser totalmente controlado pelo usuário. Isto é, o usuário pode atualizar sua chave pública e até mesmo possuir várias delas, sem ter que se comunicar com a autoridade do sistema. A chave secreta parcial será única e gerada uma só vez para cada identificador.

E, assim como no modelo baseado em identidade, é possível criar a chave pública e usá-la (para cifrar) antes mesmo que o usuário tenha entrado em contato com a autoridade para seu registro e obtenção da chave parcial (que habilitará a decifragem).

Uma das **desvantagens** do modelo que ainda não tem uma solução simples é o que foi chamado de ataque *Denial of Decryption* (DoD), em referência aos ataques de negação de serviço [Liu et al. 2007]. O DoD se dá quando alguém divulga falsamente uma chave pública ou a substitui em um repositório público, com o objetivo de prejudicar um usuário legítimo do sistema. Este, por sua vez, será incapaz de decifrar mensagens que lhe tenham sido enviadas, se tiverem sido cifradas com a chave falsa. Analogamente, assinaturas válidas deixam de ser verificáveis se for divulgada uma chave pública ilegítima. E, num primeiro instante, não será possível detectar se o que está errado é a chave ou a assinatura.

Outro preço a se pagar pela vantagem de não haver certificados é uma consequência da autenticação implícita. Quem envia uma mensagem cifrada só terá condições de verificar a autenticidade da chave pública depois que o destinatário conseguir decifrar. Um remetente terá ciência de um eventual ataque DoD somente depois de ter consumido considerável tempo de processamento e de comunicação.

Embora não seja necessário armazenar nem distribuir certificados, o modelo ainda requer um repositório de chaves públicas (ou uma forma de distribuição dessas chaves). Porém, mais crítico do que manter esse repositório, é gerenciar um **canal autêntico e confidencial** para entrega da chave secreta; no modelo convencional com ICP, o canal deve ser autêntico, mas o certificado é público.

Comparando com o modelo baseado em identidade, aqui esse canal seguro é usado com frequência um pouco menor: basta uma troca segura para cada identificador de usuário. No modelo baseado em identidade, se forem usados períodos de validade no identificador, o canal seguro terá que ser estabelecido a cada renovação.

Da forma como foi proposto originalmente, o modelo sem certificado situa-se no **nível 2 de confiança** na autoridade, pois uma autoridade desonesta pode divulgar chaves públicas falsas. E não é possível comprovar se quem gerou a chave falsa foi a autoridade ou um usuário qualquer.

Uma autoridade desonesta pode proceder da seguinte forma: ela escolhe um segredo, calcula a chave pública correspondente e a divulga como se fosse de um usuário A; se a autoridade capturar textos cifrados para A sob essa falsa chave pública, terá condições de decifrá-los, pois conhece o segredo e a chave secreta. Em seguida, para tentar ocultar a fraude, cifra novamente com a chave pública verdadeira e repassa para o destinatário, fingindo ser o remetente original. Raciocínio semelhante vale para falsificação de assinatura por parte da autoridade.

Portanto, os usuários de um sistema no modelo sem certificados precisam confiar que a autoridade do sistema não propaga ativamente chaves públicas falsas.

Conforme discutido em [Al-Riyami 2005], é possível elevar o nível de confiança dentro do modelo sem certificado, se o valor da chave pública for incluído no cálculo da chave parcial secreta. Desse modo, é possível detectar uma fraude da autoridade, sob algumas condições, explicadas a seguir.

Vamos primeiro considerar o caso de um esquema de assinatura, projetado para garantir irretratabilidade. No modelo sem certificados, para assegurar que uma assinatura não tenha posteriormente a autoria negada, o identificador de cada usuário deve incluir o valor da chave pública, por exemplo, concatenando ID com P ($ID||P$). E no cálculo da chave parcial secreta, o usuário precisa provar o conhecimento do valor secreto correspondente; ao mesmo tempo, a autoridade também deve demonstrar que conhece a chave mestra. Em outras palavras, a geração da chave parcial deve consistir de um protocolo interativo com provas de posse de segredo. Nessas condições, o esquema de assinatura tem a propriedade de **irretratabilidade** e a autoridade alcança **nível 3 de confiança**.

Para esquemas de cifragem, a análise é um pouco mais trabalhosa. Quando o identificador do usuário embute o valor da chave pública (como em $ID||P$), não há obrigatoriedade de que a chave parcial seja mantida em segredo. Isso é verdade, pois, a parcial sozinha não é a chave secreta completa e, para que a autoridade consiga bisbilhotar mensagens de algum usuário, antes terá que substituir falsamente a chave pública e o identificador, além de gerar nova parcial.

Considere, então, a seguinte fraude: a autoridade substitui a identidade original da Alice $ID||P$ por $ID||P'$; Beto, ao tentar enviar uma mensagem com sigilo para Alice, é enganado sobre os dados dela e cifra usando $ID||P'$; a autoridade captura tal mensagem e a decifra, descobrindo o conteúdo; a autoridade cifra novamente sob o identificador real e submete para Alice como se nada anormal tivesse ocorrido.

Nem sempre essa fraude será detectável. Primeiro, Beto e Alice terão que desconfiar de tal possibilidade e perceberem que a Alice possui dois identificadores distintos: um foi usado por Beto para cifrar, e outro pela Alice para decifrar.

Se a chave parcial for considerada pública, a evidência dessa fraude será a existência das duas chaves parciais, que só a autoridade consegue emitir. No entanto, a autoridade tentará ocultar a chave parcial derivada da chave pública falsa, para não se autoincriminar.

Por outro lado, se a chave parcial for considerada secreta, compartilhada apenas entre a autoridade e o usuário em questão, a evidência criptográfica dessa fraude será a existência de duas cifras para uma mesma mensagem, sob duas chaves públicas distintas de Alice. Entretanto essa evidência pode ser questionada pela autoridade: quem garante que Beto não inventou a chave pública falsa, criou as duas cifras e está tentando levantar suspeitas sobre a autoridade?

Desse modo, nem sempre a fraude será detectável para o caso de cifragem. E o nível de confiança não chega a 3.

Usos e aplicações do modelo sem certificado são em grande parte relacionados às aplicações do modelo baseado em identidade. Quando uma aplicação deste último tiver como requisito a eliminação da custódia de chaves, o modelo sem certificado pode ser empregado. Isso pode ser afirmado, pois protocolos com base na identidade quase

sempre podem ser modificados para o modelo de Al-Riyami e Paterson.

O modelo sem certificado também é indicado para uso em grupos fechados ou parceiros de negócios, em contextos em que a autoridade de confiança pode ser representada por alguém do próprio grupo.

2.2.4. Criptografia de Chave Pública Baseada em Certificado

O modelo de criptografia de chave pública baseada em certificado, proposto por [Gentry 2003] é uma variação do modelo convencional que mantém a infraestrutura de chaves públicas.

O objetivo inicial do autor era resolver o problema de revogação de chaves públicas e eliminar o tráfego de validação de certificados. Na solução proposta, os certificados são usados apenas pelo próprio usuário dono (e não por todos que precisem se comunicar com segurança com o proprietário do certificado, como é no modelo tradicional).

No modelo de Gentry, a hierarquia de autoridades certificadoras existe da mesma forma que nas ICPs sob padrão X.509. Cada usuário cria seu par de chaves, ou, alternativamente, a autoridade cria e entrega para o usuário, dependendo dos requisitos da aplicação.



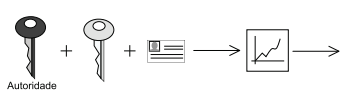
O que muda para a autoridade certificadora é que, para cada período i predeterminado, novo certificado deve ser gerado para todos os usuários (e esse período é menor que na ICP comum). Esse certificado funciona como se fosse uma parte adicional à chave secreta do usuário, pois sem certificado válido para um período, não é possível assinar nem decifrar. Portanto, do ponto de vista do usuário, o que muda em relação ao modelo convencional é a forma como é usado o certificado.

Ao proprietário do certificado, cabe a responsabilidade de obter e armazenar localmente o certificado válido para o período. Ele será usado como se fosse um componente da chave secreta, embora possa ser distribuído em canal público.

Aos usuários terceiros, o certificado não tem serventia alguma, pois para cifrar para A ou para verificar uma assinatura de A , bastam a chave pública e a identidade de A , mais o período i . A certificação da chave pública ocorre implicitamente.

Na tabela 2.5, podemos visualizar os principais parâmetros envolvidos no modelo baseado em certificado. E, na figura 2.5, é ilustrado o processo de cifragem e decifragem.

Tabela 2.5. Atributos do modelo de criptografia de chave pública baseado em certificado.

Chave secreta	Chave pública	Garantia
$[s, c]$	$[P = f(s), ID]$	$c = f(ID, P, s_{AC}, i)$
 Usuário	 Usuário	 Autoridade de Confiança
s é segredo escolhido pelo usuário (ou pela autoridade)	calculada pelo usuário (ou pela autoridade)	c é o certificado calculado pela autoridade (exclusivamente), para período i

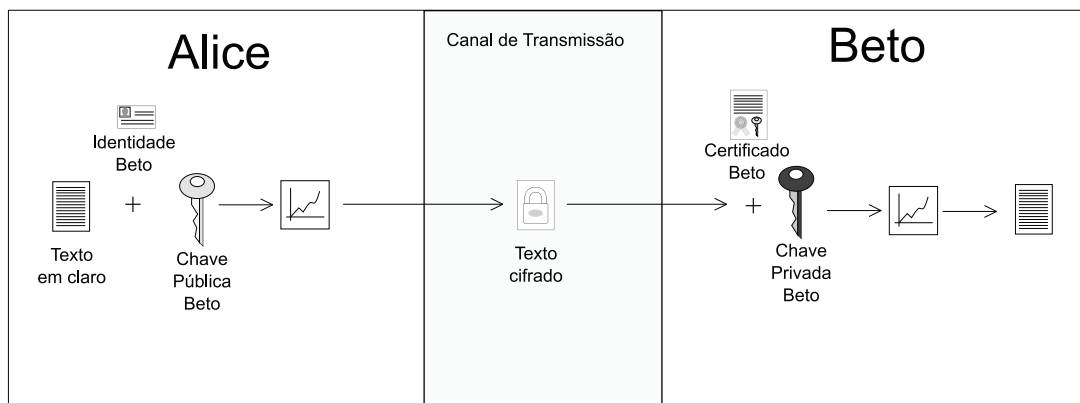


Figura 2.5. Cifrando no modelo baseado em certificado

Se compararmos com o modelo convencional de ICP, o modelo de Gentry tem como principal **vantagem** a eliminação do tráfego de validação dos certificados, ou seja, não há consultas para verificação do estado do certificado. O autor estabelece que o certificado emitido para o período i é automaticamente invalidado quando expira o prazo para o qual foi emitido. Uma premissa fundamental é que a chave pública sempre é considerada válida por quem a usa. Portanto, o período deve ser curto o bastante para que o comprometimento de uma chave secreta cause o menor dano possível até o fim daquele período.

Se uma chave secreta é comprometida, o usuário notifica a autoridade e gera novo par de chaves. A autoridade, por sua vez, passa a emitir certificados somente para a nova chave.

O trabalho que um usuário tem antes de cifrar para alguém (ou verificar uma assinatura) é, portanto, o de obter a identidade e a chave pública e de usá-las diretamente. Numa ICP convencional, como já dissemos anteriormente, primeiro é necessário obter o certificado, verificá-lo (prazo de validade e assinatura), validá-lo (consultar uma lista de certificados revogados, por exemplo), para então extrair e usar a chave pública.

Comparando com o modelo baseado em identidade, o baseado em certificado tem a vantagem de eliminar a custódia de chaves, pois a criação do par de chaves acontece exatamente como no sistema tradicional, em que a chave secreta pode ser de conhecimento exclusivo do usuário.

Em relação ao modelo sem certificado, o de Gentry apresenta como vantagens o **nível 3 de confiança** na autoridade e a não necessidade de canal sigiloso para troca de segredos. Em contrapartida, o modelo baseado em certificado ainda requer uma infraestrutura para emissão e distribuição de certificados e pode ser considerado simplesmente uma variação do modelo convencional de criptografia de chave pública sobre ICP.

A principal **desvantagem** do modelo de Gentry é a potencial sobrecarga sobre a autoridade, devido à reemissão de certificados. Isso vai depender de como for definido, no sistema, a frequência com que os certificados devem ser atualizados, combinado com o número de usuários. Para quando for grande o número de usuários, o autor descreve uma técnica auxiliar, que foi denominada *subset covers*, em que os certificados têm sobrevida

maior e são “reconfirmados” a cada período de tempo; como consequência, a carga de emissões diminui.

Um ponto bastante crítico associado ao modelo baseado em certificado situa-se nos detalhes de projeto e implementação, pois deles dependem o quão bem será resolvido o problema de revogação de certificados.

Dentro do modelo baseado em certificado, revogação significa: a autoridade para de emitir certificados para a chave pública cuja secreta foi comprometida. Na realidade, nem chave pública e nem certificado é revogado.

Suponha que Alice teve seu segredo violado e imediatamente ela entrou em contato com a autoridade para gerar novo par de chaves. Se os demais usuários não forem informados de que a chave pública foi alterada e a continuarem usando, aquele que violou a antiga chave da Alice potencialmente poderá usar o certificado “revogado” até o fim do período de validade dele, seja na tentativa de falsificar assinaturas ou para ler documentos cifrados, capturados de alguma forma.

Desse modo, o ideal é que, antes de usar uma chave pública, todo usuário a obtenha de algum repositório atualizado, ou a pegue diretamente do usuário que a gerou. E ambas possibilidades requerem sistemas online, aproximando-se mais do que costuma ser implementado tradicionalmente, com listas de certificados revogados (CLR) ou verificação online do estado do certificado (OCSP). Outra alternativa é reduzir o período i de emissão de certificados, o que sobrecarrega a autoridade.

Se a decisão de projeto for para que seja mantido um repositório de chaves públicas válidas, consultado por todos usuários antes de uma operação criptográfica, surge nova preocupação no gerenciamento de chaves: a segurança desse repositório deve ser adequada o bastante para que ele nunca seja atualizado falsamente. E esse repositório de chaves públicas substitui o que seria um repositório de certificados digitais, na implementação convencional.

Por outro lado, cada usuário talvez tenha que ter um repositório particular de certificados emitidos para vários períodos. Isso pode ocorrer se i for relativamente pequeno e existirem, por exemplo, certificados diferentes para cada hora ou menos. Suponha um sistema de email baseado em certificado e o seguinte cenário. Alice fica fora da empresa, em visita a algum cliente, passa um período sem consultar o sistema, e ao mesmo tempo recebe grande quantidade de emails cifrados. Ao retornar, Alice terá que obter todos os certificados necessários para ler as mensagens, alguns referentes a um mesmo período, outros não.

Cabe um comentário sobre o tamanho do certificado de Gentry. A rigor, o autor separa as informações do certificado em duas partes. Os dados do usuário, que são essencialmente textuais, fazem parte de seu ID ; os usuários podem manter uma cópia local ou, por praticidade, guardar apenas um hash desses dados, pois eles serão usados nos protocolos. O equivalente à assinatura da autoridade sobre os dados do certificado, fica fisicamente separado dos dados. O que o autor chama de certificado, e que é usado como parte da chave secreta, é apenas esse valor de assinatura. E somente esse valor é atualizado e transmitido constantemente.

Portanto, embora o tráfego para distribuição de certificados possa ser mais contínuo, em cada conexão, o tamanho dos dados é menor. Ainda assim, não é trivial comparar o tráfego neste modelo com o padrão X.509.

O tráfego para distribuição de certificados pode ser maior ou menor no modelo baseado em certificado. Isso depende das características do sistema que o implementar. Digamos que Alice se comunica com sigilo com uma quantidade n de outros usuários do sistema. Nenhum desses n usuários terá que obter certificado da Alice (como ocorre no modelo da ICP tradicional), mas Alice tem que obter seu próprio certificado, um número x de vezes. Durante a fase de projeto de um sistema baseado em certificado, é prudente avaliar como x se relaciona com n e averiguar se o valor de x pode ser um problema ou não.

Em [Gentry 2003], é afirmado que o modelo baseado em certificado começa a se tornar ineficiente atualizando, a cada hora, mais de 225 milhões de certificados, aproximadamente. Esses dados foram obtidos pelo autor, analisando a capacidade de autoridades certificadoras em sistemas na época.

Para a **análise de segurança** do modelo, similarmente ao modelo sem certificados, são definidos dois tipos de adversários: um representa o usuário comum, porém não certificado; outro adversário representa a própria autoridade, que não conhece o segredo dos usuários e é considerada honesta em não divulgar falsas chaves públicas, mas tenta bisbilhotar mensagens cifradas.

Os esquemas demonstrados seguros no modelo baseado em certificado preveem os dois tipos de usuário. Isto é, um esquema de cifragem é demonstrado seguro se, sob a condição de que o adversário, seja de qual tipo, não conhece o segredo associado a uma chave pública, alvo de um ataque, não tem condições de diferenciar entre dois textos cifrados de duas mensagens distintas. E isso acontece mesmo que o adversário tiver a capacidade de decifrar mensagens para outras chaves públicas. E num esquema de assinatura, apenas com o conhecimento do certificado e da chave pública, os adversários de ambos os tipos não são capazes de gerar uma assinatura válida.

Conforme detalharemos na subseção 2.3.5, os esquemas de assinatura mais recentes são demonstrados seguros mesmo que o adversário tenha a capacidade de substituir chaves públicas por valores à sua escolha, porém sob a condição de que não há emissão de certificado válido para a falsa chave.

Para se compreender o **nível de confiança** que a autoridade alcança sob o modelo baseado em certificado, é necessária uma análise semelhante à que foi apresentada no estudo do modelo sem certificado. Aqui, o certificado é uma assinatura da autoridade sobre as informações de identificação do usuário, incluindo a chave pública e um período.

Para que num esquema de assinatura haja garantia de irretratabilidade, o registro da chave pública deve ocorrer sob canal autêntico e com prova de posse de segredo. Se a autoridade divulgar uma chave pública ilegítima, ela terá condições de criar um certificado para essa chave e assinar falsamente. A fraude será detectada se toda chave pública (usada na verificação de assinaturas) for divulgada obrigatoriamente junto do certificado. Dois certificados e duas chaves comprometem a idoneidade da autoridade. Nessas condições, o nível de confiança é 3 e há irretratabilidade de assinatura.

Nos esquemas de cifração, também será necessária a publicação do certificado junto de cada chave pública. Isso se deve ao fato da autoridade pretender decifrar mensagens, cifradas sob falsas chaves públicas, e reenviar ao usuário final recifradas com a verdadeira chave. A evidência da fraude se dará com a existência de um certificado emitido para a chave falsa. Como somente a autoridade tem acesso à chave mestra, só ela emite certificados. Se a substituição de chave pública ocorrer em canal autêntico, a autoridade é responsabilizada pela fraude.

Entretanto, gerenciar o armazenamento de todos os certificados, para cada chave pública, passa a ser uma preocupação adicional. E é maior que na ICP tradicional, pois a tendência é que o volume de certificados seja maior.

2.3. Construções e Aplicações

Para as próximas subseções, selecionamos alguns protocolos que ilustram o funcionamento de modelos alternativos. Também citaremos os trabalhos que marcaram a evolução de cada paradigma e descreveremos aplicações que podem ter solução mais elegante ou eficiente em um modelo em particular. Primeiramente, descreveremos alguns conceitos preliminares, necessários para a compreensão dos algoritmos.

2.3.1. Conceitos Fundamentais

Grupos

Um grupo é um conjunto não vazio dotado de uma operação \circ , que satisfaz as seguintes propriedades [Koblitz 1994]:

- Possui um elemento identidade: quando aplicada a operação \circ sobre um elemento Q qualquer do grupo e a identidade, o resultado é o próprio elemento Q ;
- Possui o elemento inverso: quando aplicada a operação \circ sobre um elemento Q qualquer do grupo e seu elemento inverso de Q , o resultado é o elemento identidade;
- Associativo: se Q, R, S pertencem ao grupo, então $(Q \circ R) \circ S = Q \circ (R \circ S)$;
- Fechado: a operação \circ sobre elementos do grupo sempre tem como resultado um elemento do grupo.

Na verdade não definimos o que é a operação \circ , nós a usamos para genericamente definir operações de adição ou multiplicação. Quando um grupo é definido para operações de adição podemos dizer que é um grupo aditivo, neste texto iremos representá-lo por \mathbb{G}_1 ; quando um grupo é definido para operações de multiplicação dizemos que é um grupo multiplicativo e iremos representá-lo por \mathbb{G}_2 .

Quando o número de elementos de um grupo é finito, este número é chamado de ordem do grupo.

Podemos aplicar a operação \circ sobre o mesmo elemento Q do grupo n vezes, com n um número natural: $Q \circ Q \circ Q \circ \dots \circ Q$. Para $Q \in \mathbb{G}_1$ representamos por nQ , no caso de $Q \in \mathbb{G}_2$ representamos por Q^n . Dizemos que Q é um elemento gerador do grupo

quando podemos representar todos os elementos deste grupo através de Q operado sobre ele mesmo. Se o grupo possui um elemento gerador é chamado de grupo cíclico.

Em implementações práticas, os grupos de interesse são formados por pontos sobre alguma curva elíptica.

Emparelhamento Bilinear

Sejam \mathbb{G}_1 e \mathbb{G}_2 grupos cíclico de ordem prima q . Um emparelhamento bilinear admissível, de acordo com a definição em [Boneh e Franklin 2001], é um mapeamento $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, que satisfaz as seguintes condições:

1. **Bilinear:** para qualquer $P, Q \in \mathbb{G}_1$ e $a, b \in \mathbb{Z}_q$, temos:
$$e(aP, bQ) = e(P, Q)^{ab} = e(abP, Q) = e(P, abQ)$$
2. **Não Degenerado:** não leva todos os pares $\mathbb{G}_1 \times \mathbb{G}_1$ à identidade em \mathbb{G}_2
3. **Computável:** existe algoritmo eficiente que calcula $e(P, Q)$ para todos $P, Q \in \mathbb{G}_1$

Modelos de Segurança

Os modelos de segurança são a base da demonstração formal de que um algoritmo é seguro e indicam quais são as condições para que ele opere sem riscos. Os protocolos demonstravelmente seguros sob o modelo padrão tendem a ser mais robustos e os demonstrados sob o modelo do oráculo aleatório tendem a ser mais eficientes.

Ferramentas de Apoio

Para implementação e testes de criptosistemas é possível usar ferramentas matemáticas ou bibliotecas que implementam total ou parcialmente funções úteis, tais como: hash, manipulação de números grandes, algoritmos criptográficos padrão, emparelhamento bilinear, entre outras.

Na subseção 2.3.4 utilizamos uma modificação do exemplo `dl2.cpp` da biblioteca MIRACL (<http://www.shamus.ie/>) e conseguimos em poucas linhas de código descrever um esquema baseado em emparelhamento, mostrando ser possível comprovar rapidamente a adequação de um algoritmo para determinada aplicação. Essa biblioteca é de uso livre para fins educacionais, possui uma extensa quantidade de funções para manipulação de números grandes, corpos $GF(p)$ e $GF(2^m)$, curvas elípticas, funções hash e criptografia simétrica; possui interfaces em C e C++; ainda se destaca por seu alto desempenho em testes de “*Benchmarks*”.

Outras bibliotecas não proprietárias que podem auxiliar as implementações são OpenSSL disponível em <http://www.openssl.org/>, Cripto++ disponível em <http://www.cryptopp.com/>, SECCURE disponível em <http://point-at-infinity.org/seccure/>, SKS disponível em

merseine.nu/, LibECC disponível em <http://libecc.sourceforge.net/>; e outras bibliotecas proprietárias tal como Java SE 6 e a Microsoft Cryptography API.

Além disso existem ferramentas matemáticas que auxiliam a criação e demonstração de diversos modelos criptográficos, tais como as ferramentas livres SAGE, disponível em <http://www.sagemath.org>, e PARI-GP, disponível em <http://pari.math.u-bordeaux.fr/>, e outras proprietárias tais como MAGMA <http://magma.maths.usyd.edu.au/magma/>, Mathematica <http://www.wolfram.com/>, Maple <http://www.maplesoft.com/> e Matlab <http://www.mathworks.com/>. O livro [Trappe e Washington 2005] apresenta vários exemplos utilizando as três últimas ferramentas.

2.3.2. Criptografia de Chave Pública Baseada em Identidade

Quando Shamir descreveu o modelo baseado em identidade, em 1984, apenas assinatura digital ganhou algoritmos concretos. Por muitos anos sem sucesso, pesquisadores trabalharam na busca por algoritmos eficientes e seguros para cifragem e decifragem, que compõem o chamado esquema IBE (*Identity Based Encryption*).

Em 2001, dois criptossistemas baseados em identidade, mudaram este panorama. Um foi proposto por [Cocks 2001], fundamentado em resíduos quadráticos, e outro por [Boneh e Franklin 2001], baseado em emparelhamentos bilineares. Este último ganhou notoriedade não apenas pela eficiência, bem como pela formalização completa de IBE e pelas demonstrações de segurança que se tornaram referência para outros protocolos de cifragem sobre emparelhamentos. Vale mencionarmos que o uso de emparelhamentos bilineares em esquemas baseados em identidade havia sido anteriormente estudado por [Sakai et al. 2000] e que [Joux 2000] também já tinha sinalizado que emparelhamentos apresentavam grande atrativo para a criptografia, ao propor o primeiro protocolo de acordo de chaves de três participantes com uma só interação.

Os criptossistemas hierárquicos baseados em identidades surgiram na sequência, conforme detalhados em [Gentry e Silverberg 2002] e, posteriormente, em [Boneh e Boyen 2004, Yao et al. 2004, Chatterjee e Sarkar 2007]. As implementações com hierarquia de autoridades se justificam não apenas porque reduzem a responsabilidade e a sobrecarga sobre uma autoridade centralizada, mas também porque são uma abordagem para eliminar a característica de custódia de chaves.

Desde então, o modelo recebeu cada vez mais atenção dos pesquisadores, que passaram a acrescentar melhorias em várias frentes: aumento da velocidade do cálculo de emparelhamento em [Fan et al. 2008], redução do tamanho da chave em [Naccache 2007], e aumento do nível de segurança com demonstrações sem a hipótese de oráculos aleatórios, que é o caso do esquema proposto por [Waters 2005]. Paralelamente, surgiram inúmeros protocolos e aplicações interessantes; só para citar alguns:

- Assinatura em anel;
- Assinatura curtas;
- Assinatura em grupo;

- Cifassinatura;
- Acordo de chaves autenticado;
- Acordo de chaves com vários participantes;
- Implementação de disponibilidade condicional.

Passemos, então, à descrição genérica de esquema de cifragem e à respectiva concretização dessa definição genérica, apresentada em [Boneh e Franklin 2001].

Esquema de cifragem baseado na identidade

Um esquema IBE é composto por quatro fases: **inicializa**, **extrai**, **cifra** e **decifra**. Normalmente qualquer usuário pode cifrar uma mensagem usando um *ID* na fase de **cifra**. O destinatário, proprietário do *ID*, poderá decifrar a mensagem, por meio de **decifra**, usando uma chave privada correspondente a *ID*, obtida da autoridade de confiança *AC*. Conforme [Baek et al. 2004], essas fases são definidas da seguinte forma:

inicializa Os parâmetros do sistema são gerados, junto com o par de chaves da *AC*, com a privada *s* e pública *P*.

extrai Beto se autentica junto à *AC* e obtém sua chave privada s_{Beto} , que é associada à identidade ID_{Beto} .

cifra Usando a identidade do Beto, ID_{Beto} , e *P*, Alice cifra a sua mensagem *m* em texto claro e obtém o texto cifrado *C*.

decifra Ao receber o texto cifrado *C* de Alice, Beto decifra a mensagem através da sua chave privada s_{Beto} .

Segue, abaixo, o esquema IBE de [Boneh e Franklin 2001]:

inicializa Dado um parâmetro de segurança *k*, gera a chave mestra $s \in \mathbb{Z}_q^*$ e os parâmetros públicos do sistema $\text{params} = \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_{pub}, H_1, H_2, H_3, H_4 \rangle$, onde \mathbb{G}_1 e \mathbb{G}_2 são grupos de ordem prima *q*, $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ é um emparelhamento bilinear admissível, *P* é gerador de \mathbb{G}_1 , $P_{pub} = sP$ e H_i são funções de hash, tais que:

$$H_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1^*$$

$$H_2 : \mathbb{G}_2 \rightarrow \{0, 1\}^n$$

$$H_3 : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \mathbb{Z}_q^*$$

$$H_4 : \{0, 1\}^n \rightarrow \{0, 1\}^n$$

O espaço de mensagens é $\mathcal{M} = \{0, 1\}^n$. O espaço de textos cifrados é $\mathcal{C} = \mathbb{G}_1^* \times \{0, 1\}^n \times \{0, 1\}^n$.

extrai Dados um identificador $ID_A \in \{0, 1\}^*$, params e a chave mestra *s*, calcular $Q_A = H_1(ID_A)$ e a chave secreta $d_A = sQ_A$.

cifra Dados um texto $m \in \mathcal{M}$, uma identidade ID_A e **params**, calcular o texto cifrado $\langle rP, \sigma \oplus H_2(e(Q_A, P_{pub})^r), m \oplus H_4(\sigma) \rangle$, onde $r = H_3(m, \sigma)$, para $\sigma \in \{0, 1\}^n$, escolhido aleatoriamente.

decifra Dados $C = \langle U, V, W \rangle \in \mathcal{C}$, a chave secreta d_A e **params**:

Se $U \notin \mathbb{G}_1^*$, C é rejeitado. Caso contrário, calcular $V \oplus H_2(e(d_A, U)) = \sigma$ e $W \oplus H_4(\sigma) = m$. Com $r = H_3(\sigma, m)$, verificar se $U = rP$. Se for, m é a resposta, senão C é rejeitado.

Vale salientarmos que a proposta de [Boneh e Franklin 2001] foi o primeiro a ser implementado na prática e foi o ponto de partida para produtos comercializados nesta área. O esquema de [Cocks 2001] não usa emparelhamentos; despertou menor interesse devido ao alto grau de expansão do texto cifrado, porém ressurgiu melhorado em [Boneh et al. 2007], quando o problema de ineficiência em espaço computacional foi resolvido.

Adiante, descrevemos a definição de esquema de assinatura baseada na identidade, conhecido por IBS (*Identity Based Signature*).

Esquema genérico de assinatura baseada na identidade

Um esquema IBS genérico consiste em quatro fases: **inicializa**, **extrai**, **assina** e **verifica**. Normalmente, nesse esquema, Alice pretende assinar um documento, obtém da AC a sua chave de assinatura que está associada à informação do seu identificador. Ela assina a mensagem com a chave obtida. Para Beto verificar a assinatura de Alice, precisa do identificador dela (e de nenhum certificado). As fases de um esquema genérico podem ser detalhadas conforme [Baek et al. 2004]:

inicializa Os parâmetros do sistema são gerados, junto com o par de chaves da AC, com a privada s e pública P .

extrai Alice se autentica junto à AC e obtém a sua chave privada, s_{Alice} , associada à sua identidade ID_{Alice} .

assina Com a sua chave privada s_{Alice} , Alice cria a assinatura σ sobre a mensagem m .

verifica Beto verifica se σ é assinatura genuína sobre a mensagem m , usando a identidade de Alice e a chave pública da AC, P , aceitando-a ou a rejeitando.

O esquema IBS de [Shamir 1984] é considerado o ponto inicial para outros esquemas que surgiram. Por exemplo, o esquema proposto em [Hess 2003] permite a pré-computação na fase de assinatura, o que é bastante útil no caso de um assinante ter muitos documentos para assinar. A pré-computação auxilia na eliminação de uma das operações de emparelhamento na fase de verificação de assinatura, contribuindo para desempenho do sistema. Um IBS poderá ser implementado a partir de um esquema hierárquico de cifragem baseado na identidade (HIBE - *Hierarchical Identity Based Encryption*), como

comentado em [Misaghi 2008, Joye e Neven 2009]. Neste último, são dados os detalhes de construção e transformação de diversos modelos de assinatura baseada na identidade com as suas principais características.

Enquanto os esquemas de cifragem fornecem o serviço de confidencialidade, os de assinatura conferem autenticidade e integridade. Em algumas situações, esses três requisitos devem ser assegurados nas mensagens trafegadas. [Zheng 1997] mostrou que mais eficiente do que encadear os algoritmos desses dois esquemas, é compô-los em um único esquema, que foi chamado de cifrassinatura. [Malone-Lee 2002] elaborou um modelo de segurança para esquema de cifrassinatura baseado na identidade e apresentou um protocolo, que foi aprimorado por vários outros trabalhos que vieram na sequência.

Para finalizarmos as construções do modelo, comentaremos a seguir a próxima primitiva importante em criptografia de chave pública, que também ganhou versão neste modelo: acordo de chaves baseado em identidade, IBKA (*Identity Based Key Agreement*).

Esquema de acordo de chaves baseado em identidade

O protocolo de acordo de chave é um dos primitivos fundamentais em criptografia, pois por meio dele é possível usar um canal inseguro para combinar um segredo em comum entre dois ou mais participantes, sem que a chave secreta de cada um seja revelada. O segredo negociado pode dar origem a uma chave de sessão e ser usado em algoritmos de criptografia simétrica, que são bem mais eficientes que os do modelo de chave pública.

Um protocolo de acordo de chaves que ofereça autenticação mútua é chamado de protocolo de acordo de chaves autenticado. No caso do modelo baseado em identidade, a autenticação ocorre implicitamente.

Um esquema IBKA genérico consiste em três fases: **inicializa**, **extrai**, **acordo**. [McCullagh e Barreto 2004] propõem um esquema eficiente de acordo da chave com autenticação implícita. O esquema se destaca pela possibilidade de instanciamento com ou sem custódia de chaves, sem a necessidade de ter mais passos no seu esquema. O esquema apresentado a seguir é caracterizado pela custódia da chave:

inicializa Dado um parâmetro de segurança, são obtidos dois grupos \mathbb{G}_1 e \mathbb{G}_2 de ordem prima q , e um emparelhamento bilinear $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. É escolhida uma função hash $H_1 : \{0, 1\}^* \rightarrow \mathbb{Z}_q^*$. A AC seleciona um gerador P de \mathbb{G}_1 , escolhe uma chave secreta $s \in \mathbb{Z}_q^*$ e calcula a chave pública como $P_{pub} = sP$. A chave secreta é mantida sob sigilo e os demais parâmetros são distribuídos através de canais autenticados.

extrai A AC calcula a chave privada da Alice como $s_{Alice} = (a + s)^{-1}P$, onde $a = H_1(ID_{Alice})$. O valor público da Alice é $Q_{Alice} = (a + s)P$, que pode ser calculado por qualquer usuário como $aP + P_{pub}$. Analogamente, Beto possui valor público $Q_{Beto} = (b + s)P$ e seu segredo é $s_{Beto} = (b + s)^{-1}P$.

acordo O acordo da chave será feito da seguinte forma:

1. Alice escolhe um valor aleatório $x_a \in \mathbb{Z}_q^*$, calcula $T_a = x_a Q_{Beto} = x_a (b + s)P$ e envia T_A para Beto.

2. Beto escolhe um valor aleatório $x_b \in \mathbb{Z}_q^*$, calcula $T_b = x_b Q_{Alice} = x_b(a+s)P$ e envia T_B para Alice.
3. Alice calcula $K_{AB} = e(T_b, s_{Alice})^{x_a}$; de forma semelhante, Beto calcula $K_{BA} = e(T_a, s_{Beto})^{x_b}$. Se ambos seguirem esse protocolo, calcularão o mesmo segredo compartilhado, pois:

$$\begin{aligned}
 K_{AB} &= e(T_b, s_{Alice})^{x_a} \\
 &= e(x_b(a+s)P, (a+s)^{-1}P)^{x_a} \\
 &= e(P, P)^{x_a x_b} \\
 &= e(x_a(b+s)P, (b+s)^{-1}P)^{x_b} \\
 &= e(T_a, s_{Beto})^{x_b} \\
 &= K_{BA}
 \end{aligned}$$

Avanços importantes

Não é tarefa fácil enumerar todos os avanços relevantes que foram feitos sobre o modelo baseado em identidade, pois muitos trabalhos merecem menção. Entretanto, para finalizarmos esta subseção, pontuamos dois bastante recentes, que nos dão uma noção razoável sobre o atual estágio das pesquisas que tratam dois pontos críticos do modelo baseado em identidade: custódia de chaves e revogação da identidade.

Custódia de chaves é provavelmente uma das mais indesejáveis características no modelo baseado em identidade. Na tentativa de removê-la, [Al-Riyami e Paterson 2003] e [Gentry 2003], acabaram por criar modelos alternativos, que não são baseados em identidade por possuírem outra chave pública. Entretanto, mantendo-se a identidade como único valor de chave pública, a solução mais comum para eliminar a custódia de chaves gira em torno de uma hierarquia de autoridades e de adaptações sobre os esquemas existentes.

Em [Chow 2009], no entanto, é proposta uma nova abordagem para impedir que a autoridade decifre mensagens de seus usuários. Se o texto cifrado embutir uma garantia sobre o anonimato do destinatário, passa a ser proibitivo o custo computacional para que a autoridade tente decifrar as mensagens trafegadas. Chow estendeu o modelo de segurança para IBE de modo a contemplar o anonimato, descreveu um esquema concreto seguro sob esse modelo e propôs uma nova arquitetura para permitir que a emissão da chave secreta seja feita de forma anônima perante a autoridade.

A solução convencional que existe para o problema de revogação de identidade é a concatenação de períodos de validade junto do identificador do usuário, criando identidades como `JoaoSilva-Setembro09`. O problema dessa técnica é a alta carga de trabalho sobre a autoridade, na medida em que aumenta a frequência de renovação de chaves. Em termos computacionais, dizemos que a complexidade da renovação é linear no número de usuários.

No trabalho de [Boldyreva et al. 2008], a identidade é preservada (sem concatenação) e a complexidade de renovação é logarítmica no número de usuários, o que é muito eficiente quando a quantidade de usuários envolvidos é bastante grande.

Aplicações do Modelo Baseado em Identidade

Em [Appenzeller e Lynn 2002] foi proposto um novo protocolo de segurança na camada de rede que permite comunicação autenticada e cifrada entre os nós da rede. Os autores usaram IBE na formulação do protocolo, tornando-o uma **alternativa ao IPSec**. Dentre as vantagens em se usar IBE, podem ser citados o fato de ser desnecessário o processo de *handshaking* inicial e não há troca de certificados para se enviar mensagens cifradas. Neste caso, o remetente simplesmente envia um pacote cifrado com o endereço IP do destinatário.

A importância da **computação em grade** se deve ao aumento expressivo de aplicações que requerem poder computacional e capacidade de armazenamento cada vez maiores. A comunicação segura sobre uma infraestrutura de computação em grade normalmente ocorre sobre uma ICP. Uma forma de aliviar a carga de trabalho, reduzir o tráfego e evitar o armazenamento de certificados digitais é empregar o modelo baseado em identidade, como uma abordagem alternativa em tais ambientes [Lim 2006]. Nesse trabalho, o autor propõe um protocolo de acordo de chaves com autenticação mútua baseado em identidade e indica como viabilizar serviços de delegação e *single sing-on*.

Redes tolerantes a atrasos e desconexões (DTNs) são redes caracterizadas pela conectividade intermitente e que, por algum motivo, são desconectadas, interrompidas ou apresentam um certo atraso na entrega de pacotes. Podem estar em uma área de grande extensão ou até debaixo da água. Nesses tipos de redes, o modelo baseado na identidade pode contribuir na implantação de confidencialidade, conforme [Asokan et al. 2007]. Segundo os autores, a adoção de IBE em DTNs diminui a carga sobre o servidor e os receptores são menos exigidos com relação à conectividade.

Outra aplicação de interesse em que o modelo baseado em identidade tem participação é a **busca em dados cifrados**. Considere sistemas de emails cifrados, em que palavras-chave podem ser pesquisadas por um redirecionador sem que o conteúdo seja revelado. Por exemplo, uma secretária pré-organizando emails de seu diretor por palavras-chave como “urgente” ou “projetoX”. Sem que ela consiga ler o conteúdo dos emails, será capaz de redirecionar as mensagens com tais palavras-chave e priorizá-las. Naturalmente, essa secretária pode ser substituída por um sistema automatizado de filtragem, junto do servidor de emails.

Analogamente, *logs* cifrados podem ser pesquisados por um auditor pré-habilitado a localizar ocorrências relacionadas a palavras específicas. Em [Abdalla et al. 2008] é apresentado um importante trabalho nessa área; uma das contribuições refere-se a uma transformação de um esquema IBE com garantia de anonimato para um esquema seguro de criptografia de chave pública com busca de palavras-chave.

Uma das formas de se implementar autenticação mútua é por meio do uso de **cartões inteligentes**. [Scott et al. 2006] descrevem a implementação de três emparelhamentos bilineares sobre um *smartcard* de 32 bits. Neste artigo, os autores demonstram que os emparelhamentos podem ser calculados com eficiência equivalente à alcançada pelas primitivas criptográficas clássicas.

2.3.3. Criptografia de Chave Pública Autocertificada

Com o objetivo de exemplificarmos o conceito de chave pública autocertificada, vamos descrever um dos protocolos de geração de chaves de [Girault 1991], um que é baseado na assinatura RSA.

Inicialmente, a autoridade seleciona parâmetros RSA, isto é, escolhe n , produto de dois primos p, q . Calcula seu par de chaves: escolhe e relativamente primo a $(p-1)$ e $(q-1)$, e calcula d como inversa de e no módulo $(p-1)(q-1)$. Escolhe g de ordem maximal no grupo multiplicativo $(\mathbb{Z}/n\mathbb{Z})^*$. Os parâmetros n, e, g são tornados públicos e os demais, mantidos em segredo.

Um usuário com identidade I escolhe sua chave secreta s , calcula $v = g^{-s} \bmod n$ e entrega v para a autoridade. Por meio de um protocolo de identificação, o usuário deve provar à autoridade que conhece s , sem revelá-lo.

A autoridade, então, calcula e entrega a chave pública:

$$P = (g^{-s} - I)^d \bmod n$$

Conforme dissemos na seção 2.2.2, a chave pública autocertificada é calculada em função da identidade do usuário (I) e dos segredos da autoridade e do usuário (no caso acima, respectivamente s e d).

A autoridade alcançará nível 3 de confiança se demonstrar que gera os parâmetros de forma honesta, conforme detalhado em [Saeednia 2003].

Um exemplo de uso da chave autocertificada acima é dado por com um protocolo de acordo de chaves autenticado, proposto em [Girault 1991], que, embora seja vulnerável ao ataque do intermediário, ilustra o princípio de funcionamento da chave autocertificada de forma simples. Considere que os atributos de Alice são (I_A, s_A, P_A) e de Beto, (I_B, s_B, P_B) . Ambos negociam uma chave secreta calculando:

$$\text{Alice calcula } (P_B^e + I_B)^{s_A}$$

$$\text{Beto calcula } (P_A^e + I_A)^{s_B}$$

Como $d \cdot e = 1 \bmod (p-1)(q-1)$, pode-se verificar que $P^e + I = g^{-s} \bmod n$. Desse modo, os dois cálculos acima são iguais a $g^{-s_A s_B} \bmod n$. O protocolo é autenticado, pois Alice tem certeza de que conversa com Beto e vice-versa, sem a necessidade de se conferir certificados.

Uma quantidade considerável de publicações na Ásia, durante as duas últimas décadas, tem como tema a chave pública autocertificada. É possível encontrar propostas de cifragem, cifra autenticada, acordo de chaves, assinatura e assinatura em grupo, só para citar algumas. Praticamente todas são desprovidas de demonstrações formais e, portanto, não devemos considerá-las para uso prático.

Talvez boa parte desses trabalhos tenha sido motivada pelo forte potencial de **aplicação** do conceito. Em [Petersen e Horster 1997], são relatados os seguintes usos para chave pública autocertificada, com vantagens sobre o modelo convencional de certificados digitais:

- Delegação do poder de decifrar ou assinar;
- Delegação de direitos;
- Votação eletrônica;
- Dinheiro eletrônico;
- Acordo não-interativo de chaves de sessão, com autenticação.

Na ocasião da publicação de [Petersen e Horster 1997] ainda não eram conhecidas as aplicações de emparelhamentos bilineares, que hoje são base de soluções mais eficientes (e demonstravelmente seguras) para os usos acima. Entretanto, vale citarmos as ideias principais que esses autores relatam para aplicação da chave pública autocertificada, pois essas ideias são de alguma forma retomadas ou reaproveitadas em trabalhos posteriores.

Para os casos de delegação de assinatura ou de decifragem, os protocolos de geração de chaves autocertificadas são adaptados para serem executados entre o usuário principal e aquele para quem é delegado algum poder. O “procurador” calcula seu par de chaves por meio do protocolo interativo com o emissor da “procuração”. A delegação de direitos ocorre dentro de um contexto de hierarquia de autoridades; cada nó da hierarquia pode ser associado a um privilégio, que é concedido àqueles que forem previamente autorizados, por meio da emissão de chaves autocertificadas.

As aplicações de votação e dinheiro eletrônicos citadas em [Petersen e Horster 1997], se baseiam num protocolo de geração de chave pública autocertificada para um pseudônimo (em vez da identidade). O pseudônimo garante o anonimato e o sigilo do voto ou do uso de um dinheiro emitido, mas, ao mesmo tempo, oferece algum nível de rastreabilidade, por exemplo para assegurar que cada eleitor vote uma única vez, ou para que num caso de extorsão o banco e uma entidade de confiança dentro do sistema possam, em conjunto, rastrear as movimentações fraudulentas.

O acordo de chaves com autenticação de [Petersen e Horster 1997] permite que as chaves públicas autocertificadas deem origem a novos pares de chaves, recalculados pelos próprios usuários sem necessidade de interação com a autoridade. Esses pares de chaves são usados no cálculo de chaves de sessão, em protocolo com autenticação implicitamente verificada. Mais precisamente falando, cada usuário divulga um valor de testemunho (em vez da própria chave pública); a partir de um testemunho, da identidade e dos parâmetros públicos, qualquer usuário pode recalculá-la chave pública autocertificada. Quando o valor de testemunho é combinado com um período de tempo ou a um número de sessão, por exemplo, é possível realizar uma negociação não interativa de chaves de sessão.

Variações e Mais Aplicações

Quando tentamos isolar os trabalhos relacionados ao de [Girault 1991] que apresentam formalizações conceituais e/ou demonstrações de segurança, encontramos variantes do conceito original que não podem ser enquadrados dentro do modelo de chave pública autocertificada, porém mantêm algum aspecto da autocertificação.

Uma primeira variante é a assinatura autocertificada, que pressupõe a existência de uma ICP convencional. Ou seja, todo usuário tem um par de chaves calculado da forma tradicional e obtém um certificado digital para a chave pública. Esse certificado, entretanto, é distribuído de uma forma diferenciada, para otimizar as operações de verificação de assinatura.

No modelo convencional com ICP, a verificação de uma única assinatura sobre um documento acaba levando a dois procedimentos de verificação: primeiro é necessário verificar a assinatura da autoridade sobre o certificado; sendo esta válida, é extraída a chave pública do usuário para se conferir a assinatura do documento em questão. O esquema de assinatura proposto em [Lee e Kim 2002] evita essa verificação dupla. A partir do valor de assinatura do certificado obtido da autoridade, o usuário calcula um par de chaves para assinatura (pode opcionalmente ser diferente para cada documento assinado). A nova chave pública calculada é autocertificada. A assinatura a ser transmitida consiste dos dados do certificado e de um testemunho, a partir do qual é recalculada a chave pública para verificação da assinatura.

Em [Shao 2007], outro esquema de assinatura autocertificada é proposto. O autor estabelece que parte do valor da assinatura do certificado deve ser secreto e obtém uma variante mais próxima do trabalho de [Al-Riyami e Paterson 2003], de criptografia de chave pública sem certificado.

Também com o objetivo principal de eliminar a dupla verificação de assinaturas em sistemas baseados na infraestrutura de chaves públicas, existe o que a empresa Certicom chama de certificado implícito. Estruturalmente, em nada difere um certificado implícito de um padrão X.509 para chaves públicas; a infraestrutura necessária para ambos é a mesma. No entanto, o valor de assinatura no certificado é usado para qualquer usuário recalculando a chave pública autocertificada. E essa operação de extração do valor da chave pública é mais barato computacionalmente que uma verificação de assinatura (para validar um certificado).

Um exemplo de geração de certificado implícito é descrito em [Brown et al. 2002]. Todos algoritmos associados a essa tecnologia e que são desenvolvidos pela Certicom estão protegidos por uma série de patentes (por exemplo, *US Patent 20090041238*). A linha de produtos *ZigBee Smart Energy* inclui um dispositivo que realiza as funções de uma autoridade certificadora, emitindo certificados implícitos sobre curvas elípticas ECQV (Qu-Vanstone) para dispositivos com limitação de recursos (de armazenamento, de banda ou computacionais).

Outro uso menos esperado da chave autocertificada também se deu dentro do modelo com certificados. Uma dificuldade existente em sistemas com criptografia de chave pública é a validação da segurança de esquemas demonstravelmente seguros, quando implementados em conjunto com outras operações. Pode acontecer, por exemplo, um protocolo de assinatura com segurança demonstrável não ter garantia de segurança quando usado dentro de uma ICP. No trabalho de [Boldyreva et al. 2007], os modelos de segurança para esquemas de cifragem e de assinatura incluem as várias operações que acontecem dentro de uma infraestrutura de chaves públicas. Os autores propuseram dois protocolos demonstrados seguros nesse modelo. Na prática, esses protocolos oferecem maior garantia de segurança em implementações reais, pois o modelo de adversários é mais

amplo. E um dos pontos chave desse trabalho é o emprego da certificação implícita.

2.3.4. Criptografia de Chave Pública sem Certificados

Quando o modelo sem certificados foi proposto em [Al-Riyami e Paterson 2003], os autores apresentaram definições formais de um esquema de cifragem CLE (*Certificateless Encryption*), definiram um modelo de segurança extremamente forte e construíram um esquema concreto que cumpre todos os requisitos definidos.

Na sequência, surgiram várias outras propostas, muitas das quais demonstradas seguras sob um modelo de segurança aparentemente mais realista, embora mais fraco que o original. Até hoje os pesquisadores questionam se o poder dado ao adversário no modelo de segurança proposto inicialmente é demasiado forte ou não. Ninguém conseguiu apontar uma aplicação real em que o modelo se aplica na totalidade, mas, por outro lado, como já existem protocolos que atendem um grau de segurança maior, há quem os prefira, apesar deles serem um pouco menos eficientes.

Não replicaremos aqui as formalizações sobre o modelo sem certificados e referenciamos o leitor à excelente pesquisa feita por [Dent 2008], onde há também um levantamento de vários protocolos de cifragem e uma discussão sobre os modelos de segurança.

Para ilustrar o funcionamento da cifragem no modelo sem certificados, listamos o esquema de [Goya 2006] que é simples o bastante, pois envolve menos parâmetros e cifra mais curta que o de [Al-Riyami e Paterson 2003]. O esquema foi demonstrado seguro sob o modelo enfraquecido. Os códigos 1 a 3 referem-se a trechos do esquema a seguir, implementados em MIRACL.

inicializa Dado um parâmetro de segurança k , inteiros n e k_0 , com $0 < k_0 < n$, AC:

1. Gera dois grupos \mathbb{G}_1 e \mathbb{G}_2 de ordem prima $q > 2^k$ e um emparelhamento bilinear $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. Escolhe aleatoriamente um gerador $Q \in \mathbb{G}_1^*$.
2. Escolhe aleatoriamente $s \in \mathbb{Z}_q^*$ e calcular $Q_o := sQ$.
3. Escolhe três funções de *hash*

$$H_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1^*$$

$$H_2 : \mathbb{G}_1 \times \mathbb{G}_2 \times \mathbb{G}_1 \rightarrow \{0, 1\}^n$$

$$H_3 : \{0, 1\}^{n-k_0} \times \{0, 1\}^{k_0} \rightarrow \mathbb{Z}_q^*$$
4. Define:
 - $\mathcal{M} = \{0, 1\}^{n-k_0}$, como espaço de mensagens;
 - $\mathcal{C} = \mathbb{G}_1^* \times \{0, 1\}^n$, como espaço de textos cifrados;
 - s , como chave-mestra do sistema;
 - params** := $\langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, k_0, Q, Q_o, H_1, H_2, H_3 \rangle$, como parâmetros do sistema.

extrai Dados um identificador $ID_A \in \{0, 1\}^*$, **params** e a chave-mestra s :

1. Calcular $Q_A := H_1(ID_A)$.

- Entrega para entidade A a chave secreta parcial $d_A := sQ_A$.

publica Dado params , a entidade A escolhe um valor aleatório $s_A \in \mathbb{Z}_q^*$ outra chave secreta parcial e calcula sua chave pública $P_A := s_A Q$. A .

cifrar Dados um texto $m \in \mathcal{M}$, uma identidade ID_A , params e a chave pública P_A :

- Escolher aleatoriamente $\sigma \in \{0, 1\}^{k_0}$
- Calcular

$$r := H_3(m, \sigma)$$

$$Q_A := H_1(ID_A)$$

$$g^r := e(Q_o, Q_A)^r$$

$$f := rP_A$$
- Devolver o texto cifrado $C = \langle rQ, (m \parallel \sigma) \oplus H_2(rQ, g^r, f) \rangle$.

decifra Dados $C = \langle U, V \rangle \in \mathcal{C}$ e os valores secretos d_A e s_A :

- Calcular

$$g' := e(U, d_A)$$

$$f' := s_A U$$

$$(m \parallel \sigma) := V \oplus H_2(U, g', f')$$
- Desmembrar $(m \parallel \sigma)$ e calcular $r := H_3(m, \sigma)$
- Se $U := rQ$, devolver a mensagem m , senão C é rejeitado.

Código 1 - Gerar chaves para a entidade A

```
// Executado pelo usuario:
// sA (gerar valor secreto)
sA = rand(BITS_RANDOM, 2);

// PA (calcular chave publica)
PA = Q;
PA *= sA;

// Executado pela autoridade:
QA = H1("A");

// dA (chave secreta parcial)
dA = QA;
dA *= s;
```

Evolução do Modelo

Vamos pontuar os trabalhos mais significativos que marcaram a evolução do modelo de criptografia de chave pública sem certificados. Citaremos primeiramente os esquemas que contemplam o sigilo; os de assinatura são descritos posteriormente.

Código 2 - Cifrar para a entidade A

```

// m pertencente a M
str = MENSAGEM;
m = (char *) str.c_str();

// Escolher aleatorio o (sigma)
o = rand(TAM_k0_BITS, 2);

r = H3(m, o);

// g = e(Q0, QA)^r
g = tate(Q0, QA);
g = pow(g, r);

// f = r PA
f = PA;
f *= r;

// Obter texto cifrado c=<U,V>
// u = rQ;
u = Q;
u *= r;

// v = m.o XOR H2(g, U, f)
v1 = concatena(m, o);
v = H2(g, u, f);
v = XOR(v, v1);

```

Código 3 - Decifrar pela entidade A

```

// Obter g' e f'
gl = tate(u, dA);
fl = u;
fl *= sA;

r1 = H3(m, o);

P1_aux = Q;
P1_aux *= r1;

// (m.o) = v XOR H2(gl, u, fl)
mo = H2(gl, u, fl);
mo = XOR(mo, v);

// Separar m e o
m = Get_m(mo);
o = Get_o(mo);

if (P1_aux == u)
{
    cout << "Texto: " << m;
} else
{
    cout << "Erro rQ != U \n";
}

```

Dentre os protocolos de maior eficiência computacional para cifragem, podemos citar os de [Libert e Quisquater 2006] e de [Cheng et al. 2007]. Ambos usam o modelo de oráculos aleatórios para demonstrar a segurança de construções genéricas de CLE e dos esquemas concretos propostos. O primeiro desses usa como hipótese um problema computacional pior que o segundo, porém adota o modelo de adversários mais forte (ao contrário do último).

No contraponto da maior eficiência computacional, encontram-se os esquemas de maior segurança, demonstrados no modelo padrão (sem oráculos aleatórios), e sob o modelo mais forte de adversários. Dentro desse nicho, o trabalho [Dent et al. 2008] é a melhor referência que se tem até o momento.

Embora a esmagadora maioria das propostas envolva emparelhamentos bilineares, não é obrigatória tal técnica para que seja viável o modelo sem certificados. Em [Sun et al. 2007], há um exemplo de esquema de cifragem sem emparelhamentos. Os autores melhoram uma versão anterior, [Baek et al. 2005], entretanto preservam uma grande

quantidade de exponenciações. Em termos de eficiência, esse esquema é superado por muitos outros que são baseados em emparelhamentos. Só para se ter uma ideia, a versão mais antiga é comparada com esquemas de cifrassinatura em [Barreto et al. 2008] e perde em várias ocasiões (isto é, mesmo apenas cifrando, o esquema sem emparelhamento é mais lento que alguns esquemas que cifram e assinam simultaneamente).

Talvez não seja exagero dizer que o calcanhar de Aquiles do modelo sem certificados é o que se convencionou chamar de *Denial of Decryption* (DoD). É razoável pensar em uma tal forma de ataque, pois as chaves públicas são distribuídas e não certificadas explicitamente. Se o destinatário (dono da verdadeira chave pública) não conseguir decifrar ou obter uma mensagem diferente da original, o ataque é bem sucedido. No contexto de assinatura, usuários não conseguem validar assinaturas legítimas e sequer podem identificar que o erro está na chave pública e não na assinatura. No caso de implementações que envolvam um repositório centralizado para armazenar as chaves públicas, o ataque DoD terá este nome mais que justificado, se o impostor substituir várias (ou todas) chaves.

A solução delineada em [Liu et al. 2007] para se evitar o DoD envolve o cálculo da chave pública dependentemente da chave parcial secreta. Isso, entretanto, elimina uma das características peculiares do modelo sem certificado, que é a possibilidade de se gerar chaves públicas antes da interação com a autoridade do sistema. A proposta combina CLE e CLS (respectivamente cifragem e assinatura no modelo sem certificado), criando **duas** chaves parciais secretas, uma para CLE e outra para CLS. Portanto, há também duas chaves secretas completas, uma para decifrar e outra para assinar. A chave pública passa a ser um par $\langle pk, \sigma \rangle$ onde σ é a assinatura do usuário sobre sua própria chave pública pk . Um remetente deve verificar σ antes de cifrar com pk e, assim, DoD é evitado.

Os autores chamaram essa solução de *self-generated certificate PKC*; ela se assemelha a certificados autoassinados em ICPs. Um questionamento que ainda está em aberto é se não é possível outro tipo de solução, pois, na verdade, os autores inseriram uma espécie de certificado (a assinatura, que deve ser verificada e que só pode existir depois de uma interação com a autoridade) em um modelo que, em princípio, é sem certificados.

Um outro aspecto problemático no modelo sem certificados, ainda que em menor escala que o DoD, é o ataque da **autoridade mal intencionada**. O modelo de segurança definido originalmente em [Al-Riyami e Paterson 2003], presume que a autoridade é honesta e segue os protocolos conforme especificados. Em [Au et al. 2007b], no entanto, é apresentada a possibilidade de que a autoridade gere parâmetros desonestamente, de modo a conseguir um atalho para decifrar textos, sem o conhecimento dos usuários.

À exceção de [Libert e Quisquater 2006, Hu et al. 2006], todos os esquemas propostos até o trabalho de [Au et al. 2007b], são vulneráveis a essa autoridade mal intencionada e mesmo os trabalhos mais recentes também o são em grande parte. O esquema de [Dent et al. 2008], por exemplo, falha nesse quesito. [Dent 2008] provou que um esquema CLE construído sob a técnica de [Dodis e Katz 2005] evita esse tipo de ação; uma implementação concreta de esquema de cifragem sob essa técnica pode ser vista em [Chow et al. 2006], cuja proposta tem por objetivo principal o de resolver o problema de **revogação de chaves públicas** no modelo sem certificados. Outro trabalho que trata o problema da autoridade mal intencionada é o de [Hwang et al. 2008], que apresenta um

CLE demonstrado seguro sem oráculos aleatórios.

A propósito do problema de revogação, a solução de [Chow et al. 2006] requer um mediador confiável, a quem a autoridade entrega as chaves parciais secretas, e que se torna capaz de iniciar o processo de decifragem. A cifra parcialmente decifrada é submetida ao destinatário final, que completa a operação com o secreto aleatório. Com esta solução, o mediador deve estar online e disponível sempre que alguém precisar cifrar ou decifrar.

Com relação a **assinaturas** no modelo sem certificados, existem inúmeros trabalhos publicados, muitos porém demonstrados posteriormente inseguros. Um esquema que ainda se mantém seguro é apresentado em [Zhang et al. 2006], que é melhorado em [Hu et al. 2007]. Este último aperfeiçoa alguns aspectos do modelo de segurança para assinatura, mas regride em outro (o oráculo de assinatura sempre responde corretamente, mesmo que o adversário forneça valor inválido de chave secreta). Todas essas propostas se baseiam no modelo do oráculo aleatório.

Em [Liu et al. 2007], há um esquema de assinatura demonstrado seguro no modelo padrão, no entanto requer parâmetros bastante longos.

Vale citar o trabalho de [Zhang e Wang 2008] que, embora tenha usado um modelo mais fraco de adversários, apresentou CLS seguro contra autoridade mal intencionada, não vulnerável ao ataque DoD, que alcança nível 3 de Girault e todas as demonstrações são feitas sob o modelo padrão, sem oráculos aleatórios. Os autores se valem de uma técnica menos usual em CLS: é adotado um protocolo de conhecimento-zero na geração da chave parcial secreta.

Aplicações do Modelo sem Certificado

Um **fluxo criptográfico**, na denominação de [Al-Riyami 2005], é uma sequência de operações criptográficas, como cifrar, autenticar e decifrar, que precisa ser executada numa determinada ordem. No modelo baseado em identidade e no sem certificado, é possível emitir e usar a chave pública antes que a chave secreta completa esteja disponível. O exemplo abaixo ilustra uma aplicação que faz uso dessa característica.

Na maneira tradicional, quando a emissão de um documento depende da aprovação de muitos órgãos, o usuário solicita a aprovação de cada órgão e, com essas autorizações em mãos, é feita uma solicitação ao órgão emissor do documento. Este último verificará a validade de cada autorização para então emitir o documento solicitado.

Com o uso de sistemas sem certificado, é possível ao órgão emissor entregar o documento eletrônico cifrado com a chave pública e identidade do usuário. Cada órgão que deve aprovar o documento entrega ao usuário uma chave parcial privada. Após recolher todas as chaves parciais que compõe a chave secreta completa, o usuário passa a ter acesso ao documento, sem a necessidade de retornar no órgão emissor. Como não há custódia de chaves, o conluio de um ou mais órgãos não permitirá acesso ao documento.

Esse mesmo exemplo poderia ser adotado com software em ambientes sigilosos, onde o acesso ao software só é liberado após conclusão de etapas de autenticação.

As construções genéricas de esquemas sem certificado se valem de uma compo-

sição de esquemas seguros de criptografia de chave pública convencional e baseada em identidade. Isso indica que as aplicações baseadas em identidades podem ser convertidas para o modelo sem certificado, ao custo da necessidade de divulgação das chaves públicas (ou da manutenção de um diretório de chaves) e de um canal seguro para distribuição dos segredos parciais. O modelo sem certificado pode vir a ser uma **ponte** entre sistemas baseados em identidade com os que requerem ICP, em particular, a **ICP-Brasil**.

A possibilidade de renovação da chave pública pelo usuário, sem interação com a autoridade, é uma característica exclusiva do modelo de Al-Riyami e Paterson. Isso pode vir a ser vantagem e ser explorado em alguma aplicação específica. Alguns protocolos com base na chave pública autocertificada apresentam tal propriedade, porém sem garantia de segurança.

2.3.5. Criptografia de Chave Pública Baseada em Certificado

Na ocasião da publicação do trabalho de [Gentry 2003], o autor havia definido apenas a cifragem sob o modelo e estudou como seria a construção num ambiente com várias autoridades em hierarquia. Desde então, surgiram novas propostas de esquemas para cifragem e também apareceram modelos para assinatura no paradigma de Gentry.

Na medida em que novos trabalhos foram apresentados, as formalizações do modelo evoluíram e os algoritmos foram aprimorados, seja com relação ao desempenho computacional, seja na melhoria de algum aspecto de segurança. Vamos citar alguns trabalhos de maior interesse.

Cifragem no Modelo Baseado em Certificado

Para que o leitor possa assimilar a conceituação do modelo discutida na seção 2.2.4, vamos revisar o esquema cifragem baseado em certificado (CBE) de [Gentry 2003], que possui demonstração de segurança contra ataques de texto cifrado escolhido. O esquema é composto de cinco algoritmos, fundamentados em emparelhamento bilinear:

inicializa Dado um parâmetro de segurança k , gera a **chave mestra** $s \in \mathbb{Z}/q\mathbb{Z}$ e os parâmetros públicos do sistema $\text{params} = \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_{pub}, H_1, H_2, H_3, H_4 \rangle$, onde \mathbb{G}_1 e \mathbb{G}_2 são grupos de ordem prima q , $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ é um emparelhamento bilinear admissível, P é gerador de \mathbb{G}_1 , $P_{pub} = sP$ e H_i são funções de hash, tais que:

$$H_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1^*$$

$$H_2 : \mathbb{G}_2 \rightarrow \{0, 1\}^n$$

$$H_3 : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \mathbb{Z}_q^*$$

$$H_4 : \{0, 1\}^n \rightarrow \{0, 1\}^n$$

O espaço de mensagens é $\mathcal{M} = \{0, 1\}^n$. O espaço de textos cifrados é $\mathcal{C} = \mathbb{G}_1^* \times \{0, 1\}^n \times \{0, 1\}^n$.

A entidade A escolhe aleatoriamente sua chave secreta $t_A \in \mathbb{Z}_q^*$ e calcula sua chave pública $N_A = t_A P$, como no modelo convencional.

certifica A identificação do usuário A é A_{info} , que combina um identificador $ID_A \in \{0, 1\}^*$, com N_A . A autoridade calcula $P_A = H_1(P_{pub}, i, A_{info})$, para um período i . O certificado $Cert_A = sP_A$ é enviado à entidade A , que calcula sua chave de decifragem (válida para o período i): $S_A = Cert_A + t_A P'_A$, onde $P'_A = H_1(A_{info})$.

cifra Dados um texto $m \in \mathcal{M}$, um identificador A_{info} , um período de tempo i e params:
Calcular o texto cifrado $\langle rP, \sigma \oplus H_2((e(P_{pub}, P_A)e(N_A, P'_A))^r), m \oplus H_4(\sigma) \rangle$, onde $r = H_3(m, \sigma)$, para $\sigma \in \{0, 1\}^n$, escolhido aleatoriamente, e P_A e P'_A são calculados como em **certifica**.

decifra Dados $C = \langle U, V, W \rangle \in \mathcal{C}$, a chave secreta S_A e params:

Calcular $V \oplus H_2(e(S_A, U)) = \sigma$ e $W \oplus H_4(\sigma) = m$. Com $r = H_3(\sigma, m)$, verificar se $U = rP$. Se for, m é a resposta, senão C é rejeitado.

Gentry adotou a assinatura agregada do esquema baseado em identidade de [Boneh et al. 2003], para usar como chave de decifragem: o valor de assinatura S_A agrega os segredos da autoridade e do usuário, permitindo a correta inversão da operação criptográfica.

A definição formal de cifragem no modelo baseado em certificado foi revista em [Al-Riyami e Paterson 2005]; modificações sutis também foram acrescentadas no modelo de adversários. A partir de então, todos os esquemas de cifragem baseados em certificado seguem as definições e modelo de segurança desse trabalho.

Ainda em [Al-Riyami e Paterson 2005], é dada uma prova de que, dado um esquema de cifragem sem certificado seguro, é possível construir um esquema seguro baseado em certificado. Contudo, em [Kang e Park 2005] é apontada uma falha na demonstração que invalida o resultado. Os autores lembram que também em [Yum e Lee 2004] foi apresentada uma demonstração de equivalência entre os modelos, posteriormente invalidada, e sugerem que essa sequência de insucessos pode ser um indício de que cada um dos conceitos tenha vantagens próprias, apesar dos vários aspectos em comum.

O trabalho de [Dodis e Katz 2005] apresenta uma série de resultados importantes para a criptografia de chave pública. Além de formalizar segurança de cifragem múltipla (encadeada), os autores propõem construções genéricas para composição segura de esquemas de cifragem (PKE, no modelo convencional) e estudam aplicações decorrentes. Uma das aplicações apontadas resulta na primeira construção genérica de um CBE seguro, a partir de um IBE e um PKE seguros. Todas as demonstrações de segurança são feitas no modelo padrão, sem oráculos aleatórios.

Uma descrição simplificada da construção de Dodis e Katz é dada em [Galindo et al. 2008]: para cifrar uma mensagem m para um usuário com identidade ID , para um período i :

- Gerar um par de chaves (vk, sk) de um esquema de assinatura de uso único (OTS, one-time signature);
- Quebrar m em duas partes m_1 e m_2 tais que $m = m_1 \oplus m_2$;

- Cifrar m_1 usando o esquema IBE, com identidade $id||i$ e label vk , gerando C_1 ;
- Cifrar m_2 usando o esquema PKE, com label vk , gerando C_2 ;
- Assinar (C_1, C_2) com a chave sk , gerando σ ;
- Texto cifrado de saída é $C = (vk, C_1, C_2, \sigma)$.

Na tentativa de concretizar um esquema de cifragem mais eficiente, os autores em [Galindo et al. 2008] adotaram uma estratégia diferente e conseguiram reduzir a cifra para $C = (vk, C_1, \sigma)$, embutindo C_2 em C_1 , e a demonstraram segura sem oráculos aleatórios.

Um terceiro esquema CBE demonstrado seguro no modelo padrão foi dado em [Liu e Zhou 2008]. Este, entretanto, usa como hipótese a dificuldade de um problema computacional pouco conhecido.

Outra construção genérica de CBE a partir de um PKE e de um IBE seguros é descrita em [Lu e Li 2008], que usa a transformação de [Fujisaki e Okamoto 1999] para alcançar segurança contra ataques de texto cifrado escolhido (mesma estratégia usada em [Boneh e Franklin 2001] e [Gentry 2003] e tantos outros). As demonstrações acontecem sob oráculos aleatórios.

Posteriormente, em [Lu et al. 2009], é apresentado um esquema concreto eficiente, baseado nessa última construção genérica e na geração de chaves de [Sakai e Kasahara 2003]. Como resultado é obtido um esquema de cifragem mais eficiente que o de original de [Gentry 2003], ao custo da hipótese de dificuldade de um problema computacional menos estudado.

Assinatura no Modelo Baseado em Certificado

A primeira formalização de assinatura para o modelo baseado em certificado foi feita em [Kang et al. 2004]. Nesse mesmo artigo, foi proposto um esquema concreto de assinatura, que posteriormente foi demonstrado inseguro a um ataque de substituição de chave pública, em [Li et al. 2007].

Esses últimos autores fortaleceram o modelo de adversário, fornecendo a ele o poder de substituição de chaves públicas. No jogo com o adversário, o simulador do sistema permite que as chaves públicas sejam substituídas por valores à escolha do adversário, para quaisquer usuários. Se for garantido que o impostor não obtém certificado válido para uma falsa chave, ele não terá condições de forjar assinatura em um esquema demonstrado seguro.

Na prática, o novo modelo de segurança para assinatura baseada em certificado se tornou mais próximo do proposto por [Al-Riyami e Paterson 2003], para o modelo sem certificado. Os trabalhos subsequentes relacionados a assinaturas seguem esse modelo de adversário fortalecido. Curiosamente, os mais recentes esquemas de cifragem baseados em certificado ainda não pressupõem substituição de chaves.

Além de melhorar os requisitos de segurança para assinatura, [Li et al. 2007] apresentaram esquema concreto com emparelhamentos bilineares, com demonstração de segurança no novo modelo, sob oráculo aleatório.

Para completar um leque de possibilidades, em [Liu et al. 2008] foram apresentadas mais duas assinaturas: uma sem emparelhamentos, com o objetivo de maximizar desempenho computacional; e outra, sem oráculos aleatórios, para fornecer uma solução no mais alto nível de segurança. O primeiro esquema de assinatura, sem emparelhamentos, foi comparado a outros e os autores afirmaram superioridade em desempenho; a demonstração de segurança foi baseada no modelo de oráculos aleatórios. O segundo esquema, demonstrado seguro no modelo padrão, faz uso de emparelhamentos bilineares; é teoricamente mais seguro e perde em desempenho, quando comparado aos demais, apenas por conta da função de hash concretizada como em [Waters 2005].

Variações sobre assinatura também foram estudadas para o modelo: em [Au et al. 2007a] há a proposta de assinatura em anel; [Shao 2008] propõe um esquema de assinatura cifrada verificável; e [Liu et al. 2009] elaboram esquema de assinatura agregada baseada em certificado, em que n mensagens podem ser assinadas por n participantes, com comprimento fixo, independente de n .

Tanto a assinatura agregada de [Liu et al. 2009] quanto a assinatura sem emparelhamentos de [Liu et al. 2008] são sugeridas para uso em ambientes com restrição de recursos computacionais ou de banda, como redes sem fio (de sensores ou dispositivos móveis).

Todos os esquemas citados pressupõem que a autoridade do sistema gera honestamente os parâmetros do sistema. Isto é, até a elaboração deste texto, nenhum trabalho levou em consideração a ação de uma autoridade que gera os parâmetros públicos de modo a obter vantagens em conseguir, sem ser detectada, falsificar assinaturas ou decifrar mensagens de usuários. [Au et al. 2007b] fazem tal alerta no contexto do modelo sem certificado, mas também se aplica ao modelo baseado em certificado.

Aplicações do Modelo Baseado em Certificado

Uma aplicação interessante que o modelo de Gentry possibilita é uma construção mais elegante de um sistema de proxy em que é possível revogar o poder de decifragem de um “procurador”, ainda durante o período de validade da chave dada a ele. O modelo formal e a segurança de um esquema concreto foram analisados em [Wang et al. 2007].

Basicamente, uma aplicação que possa ser realizada em uma ICP convencional, pode ser implementada no modelo de Gentry. A escolha entre um modelo ou outro depende essencialmente da conveniência ou não da característica de renovação de certificados para tratar revogação.

O modelo baseado em certificado requer implementações mais semelhantes às de um sistema convencional de chave pública. São poucas as diferenças existentes entre uma ICP e a infraestrutura de gerenciamento de certificados do modelo de Gentry. Por esse motivo, é mais fácil o reaproveitamento de módulos prontos ou de bibliotecas.

2.4. Considerações e Conclusões

Apresentaremos, nas próximas subseções, comparações gerais entre os modelos de criptografia de chave pública e algumas considerações sobre implementação. Por fim, finalizamos com sugestões de trabalhos que podem ser desenvolvidos nesses temas, seguidas de conclusões.

2.4.1. Comparações Gerais

Conforme já indicamos na seção anterior, alguns pesquisadores exploraram as semelhanças existentes entre os modelos aqui estudados, para tentar construir conversões genéricas de um para o outro. Todas as tentativas foram posteriormente invalidadas, devido a alguma falha nas demonstrações de conversão. Exemplos disso podem ser vistos em [Yum e Lee 2004, Al-Riyami e Paterson 2005], contestados respectivamente em [Libert e Quisquater 2006, Kang e Park 2005]. No primeiro desses trabalhos, chegou-se a mostrar (falsamente) que cifragem nos modelos baseado em identidade, sem certificado e baseado em certificado (IBE, CLE e CBE) são essencialmente equivalentes entre si, isto é, dado um deles se constrói os outros dois.

Portanto, pelo menos até o momento, cada um dos paradigmas aparentemente possui vida própria, com propriedades, prós e contras próprios.

Abstraindo-se as diferenças relacionadas com os detalhes de gerenciamento de chaves, podemos classificar e ordenar os modelos de chave pública da seguinte forma:

- num extremo, fica o modelo baseado em identidade, com nível de confiança 1;
- no outro extremo, fica o modelo convencional sobre ICP, com nível de confiança 3;
- no meio ficam os demais, híbridos dos dois primeiros, com níveis de 1 a 3.

Os atributos de criptografia de chave pública, discutidos ao longo do texto, encontram-se resumidos na tabela 2.6. Tais atributos são compostos pelo par de chaves (pública e secreta) e pela garantia de que o par se relaciona com seu dono, identificado por ID . A função f em cada ocorrência na tabela é uma representação genérica de função matemática; possui propriedades específicas em cada caso.

Tabela 2.6. Atributos dos modelos de criptografia de chave pública.

Atributos	Modelos				
	Com ICP	Baseado em Identidade	Auto-certificado	Sem Certificado	Baseado em Certificado
Chave secreta	s	$s = f(ID, s_{AC})$	s	$s = [x, f(ID, s_{AC})]$	$[s, c]$
Chave pública	$P = f(s)$	ID	$P = f(ID, s, s_{AC})$	$[f(x), ID]$	$[P = f(s), ID]$
Garantia	$f(ID, P, s_{AC})$	s	P	s	$c = f(ID, P, s_{AC}, i)$

Na tabela 2.7, algumas das propriedades dos modelos são colocadas lado a lado para comparação. Nela, a segunda coluna refere-se ao modelo convencional sobre ICP. Em geral, um “sim” é considerado um ponto positivo; um “não”, ponto negativo.

A primeira propriedade, “Dispensa ICP”, diz respeito à infraestrutura tradicional de gerenciamento de chaves públicas. Naturalmente, todos os modelos necessitam de algum tipo de infraestrutura, cujos requisitos e características são retratados pelas demais linhas da tabela. A propriedade “Fluxo criptográfico” se refere ao uso da chave pública antes da emissão da chave secreta.

Tabela 2.7. Comparação entre os modelos de criptografia de chave pública.

Propriedades	Modelos				
	Com ICP	Baseado em Identidade	Auto-certificado	Sem Certificado	Baseado em Certificado
Dispensa ICP	não	sim	sim	sim	não
Dispensa certificados	não	sim	sim	sim	não
Dispensa diretório de chaves ou certificados	não	sim	não	não	não
Certificação explícita	sim	não	não	não	não
Nível de confiança	3	1	3(*)	2	3
Sem custódia de chaves	sim	não	sim	sim	sim
Chave secreta criada integralmente pelo usuário	sim	não	sim	não	não
Irretratibilidade	sim	não	sim	sim(*)	sim(*)
Risco da chave mestra	alto	altíssimo	alto	alto	alto
Dispensa canal seguro para distribuir chaves	sim	não	sim	não	sim
Renovação de chaves controlada pelo usuário	não	não	não	sim	não
Fluxo criptográfico	não	sim	não	sim	não
Principais protocolos demonstrados seguros	sim	sim	não	sim	sim

(*) Sob as condições discutidas no texto

2.4.2. Considerações sobre Implementação

Achamos prudente frisar alguns detalhes relacionados à implementação de criptosistemas baseados nos modelos alternativos, embora muitos deles se aplicam também ao caso do modelo tradicional. Pelo fato dessas alternativas serem relativamente novas, não há ainda padronizações internacionais generalizadas que guiem o desenvolvedor.

Em primeiro lugar, esquemas e protocolos desacompanhados de demonstração de segurança não devem ser considerados para implementações práticas. Caso a demonstração de segurança seja baseada no modelo do oráculo aleatório, é necessário ter um cuidado especial com a escolha e implementação das funções hash, pois elas podem vir a ser ponto vulnerável no sistema. Também é preciso que se tenha atenção ao modelo de adversário usado na demonstração de segurança; as hipóteses lá consideradas devem ser contempladas na implementação da aplicação.

A escolha adequada de curvas elípticas e do emparelhamento bilinear, quando for o caso, exercem influência fundamental no desempenho final dos algoritmos escolhidos, pois há várias otimizações já desenvolvidas. Um texto que é boa referência para

quem deseja criar aplicações baseadas em emparelhamentos é o capítulo de implementações escrito por Hankerson, Menezes e Scott em [Joye e Neven 2009]. Ademais, existem algumas padronizações para escolha de curvas elípticas, de modo a evitar aquelas já conhecidas como inseguras (ex. FIPS 186-3).

Dentre os alternativos, o modelo baseado em identidade é o que já possui propostas para padronizações. A RFC 5091, de dezembro 2007 (<http://tools.ietf.org/html/rfc5091>), aborda padrões para implementação dos algoritmos de [Boneh e Franklin 2001] e [Boneh e Boyen 2004] a partir de curvas supersingulares. Existem outras padronizações escritas pela Voltage Security, empresa pioneira em comercializar produtos baseados no modelo. A RFC 5408, de 2009, por exemplo, descreve a arquitetura necessária para implementar IBE e define as estruturas de dados suportadas.

2.4.3. Sugestões de Trabalhos Futuros

Os modelos alternativos são relativamente novos e existem poucas implementações em execução. Somente as implantações reais podem confirmar as vantagens e desvantagens levantadas no plano conceitual e revelar dificuldades ainda ocultas. Portanto, uma primeira sugestão de trabalhos está relacionada a implementações nos modelos alternativos, avaliando também aspectos de eficiência computacional.

Como a proposta de Girault não evoluiu para um modelo independente, há espaço para o desenvolvimento de um conjunto de primitivas básicas que garantam confidencialidade, integridade e autenticidade demonstravelmente seguras. Ou, ao contrário, demonstrar que o modelo é impraticável.

É possível melhorar alguns aspectos em cada modelo, eliminando uma característica ruim ou acrescentando uma propriedade atraente. Refinamentos assim tem sido estudados, mas há pontos ainda em aberto, em todos os modelos.

Pudemos observar que a composição dos atributos de criptografia de chave pública comprometem a caracterização de cada modelo. Variações sobre as composições atuais podem induzir novas variantes ou aprimoramentos.

2.4.4. Considerações Finais

Neste texto foram discutidas as propriedades de quatro modelos de criptografia de chave pública que são alternativos ao convencional, por minimizarem algumas das dificuldades impostas pela infraestrutura de chaves públicas. A partir da análise conceitual, foram levantados pontos fortes e fracos de cada modelo, e discutidas algumas possíveis aplicações.

Os quadros comparativos apresentam um sumário dos parâmetros que constroem cada modelo e uma síntese das características relevantes, que são meras consequências da variações exercidas sobre esses parâmetros.

Aos profissionais de desenvolvimento ou de implantação de sistemas de segurança, as revisões conceituais ajudam na escolha do modelo adequado de criptografia de chave pública para a aplicação em particular. Pesquisadores e estudantes podem contribuir com a evolução dos modelos, tendo como ponto de partida a melhor compreensão deles.

Referências

- [Abdalla et al. 2008] Abdalla, M., Bellare, M., Catalano, D., Kiltz, E., Kohno, T., Lange, T., Malone-Lee, J., Neven, G., Paillier, P., e Shi, H. (2008). Searchable encryption revisited: Consistency properties, relation to anonymous ibe, and extensions. *J. Cryptol.*, 21(3):350–391. 27
- [Abdalla et al. 2006] Abdalla, M., Catalano, D., Dent, A., Malone-Lee, J., e Smart, N. (2006). Identity-based encryption gone wild. In *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006*, pages 300–311. Springer-Verlag LNCS 4052. 9
- [Al-Riyami 2005] Al-Riyami, S. S. (2005). *Cryptographic Schemes based on Elliptic Curve Pairings*. Tese de doutorado, Department of Mathematics, Royal Holloway, University of London. 15, 35
- [Al-Riyami e Paterson 2003] Al-Riyami, S. S. e Paterson, K. G. (2003). Certificateless public key cryptography. In *ASIACRYPT 2003*, volume 2894 of *Lecture Notes in Computer Science*. Springer. Cryptology ePrint Archive, Report 2003/126, <http://eprint.iacr.org/>. 11, 12, 13, 26, 30, 31, 34, 38
- [Al-Riyami e Paterson 2005] Al-Riyami, S. S. e Paterson, K. G. (2005). Cbe from cl-pke: A generic construction and efficient schemes. In *Public Key Cryptography - PKC 2005*, volume 3386 of *Lecture Notes in Computer Science*, pages 398–415, Les Diablerets, Switzerland. Springer. 37, 40
- [Appenzeller e Lynn 2002] Appenzeller, G. e Lynn, B. (2002). Minimal-overhead ip security using identity-based encryption. Disponível em: <http://rooster.stanford.edu/~ben/pubs/ipibe.pdf>. 27
- [Asokan et al. 2007] Asokan, N., Kostianen, K., Ginzboorg, P., Ott, J., e Luo, C. (2007). Applicability of identity-based cryptography for disruption-tolerant networking. In *MobiOpp '07: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pages 52–56, New York, NY, USA. ACM. 27
- [Au et al. 2007a] Au, M. H., Liu, J. K., Susilo, W., e Yuen, T. H. (2007a). Certificate based (linkable) ring signature. In *ISPEC*, volume 4464 of *Lecture Notes in Computer Science*, pages 79–92. Springer. 39
- [Au et al. 2007b] Au, M. H., Mu, Y., Chen, J., Wong, D. S., Liu, J. K., e Yang, G. (2007b). Malicious kgc attacks in certificateless cryptography. In *ASIACCS '07: Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security*, pages 302–311, New York, NY, USA. ACM. 34, 39
- [Baek et al. 2004] Baek, J., Newmarch, J., Safavi-Naini, R., e Susilo, W. (2004). A survey of identity-based cryptography. AUUG 2004. Disponível em: <http://jan.netcomp.monash.edu.au/publications/>. 23, 24
- [Baek et al. 2005] Baek, J., Safavi-Naini, R., e Susilo, W. (2005). Certificateless public key encryption without pairing. In *ISC*, volume 3650 of *Lecture Notes in Computer Science*, pages 134–148, Singapore. Springer. 33

- [Barreto et al. 2008] Barreto, P. S. L. M., Deusajute, A. M., de Souza Cruz, E., Pereira, G. C. F., e da Silva, R. R. (2008). Toward efficient certificateless signcryption from (and without) bilinear pairings. In *SBSeg 2008*. 34
- [Boldyreva et al. 2007] Boldyreva, A., Fischlin, M., Palacio, A., e Warinschi, B. (2007). A closer look at pki: Security and efficiency. In *PKC 2007*, volume 4450 of *Lecture Notes in Computer Science*, pages 458–475. Springer. 30
- [Boldyreva et al. 2008] Boldyreva, A., Goyal, V., e Kumar, V. (2008). Identity-based encryption with efficient revocation. In *CCS '08: Proceedings of the 15th ACM conference on Computer and communications security*, pages 417–426, New York, NY, USA. ACM. 26
- [Boneh e Boyen 2004] Boneh, D. e Boyen, X. (2004). Efficient selective-ID secure identity based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238. Berlin: Springer-Verlag. Disponvel em: <http://www.cs.stanford.edu/~xb/eurocrypt04b/>. 22, 42
- [Boneh e Franklin 2001] Boneh, D. e Franklin, M. K. (2001). Identity-based encryption from the weil pairing. In *CRYPTO '01: Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, pages 213–229, London, UK. Springer-Verlag. 12, 21, 22, 23, 24, 38, 42
- [Boneh et al. 2007] Boneh, D., Gentry, C., e Hamburg, M. (2007). Space-efficient identity based encryption without pairings. In *FOCS '07: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 647–657, Washington, DC, USA. IEEE Computer Society. 24
- [Boneh et al. 2003] Boneh, D., Gentry, C., Lynn, B., e Shacham, H. (2003). Aggregate and verifiably encrypted signatures from bilinear maps. In *EUROCRYPT*, pages 416–432. 37
- [Brown et al. 2002] Brown, D. R. L., Gallant, R. P., e Vanstone, S. A. (2002). Provably secure implicit certificate schemes. In *FC '01: Proceedings of the 5th International Conference on Financial Cryptography*, pages 156–165, London, UK. Springer-Verlag. 30
- [Chatterjee e Sarkar 2007] Chatterjee, S. e Sarkar, P. (2007). Constant size ciphertext hibe in the augmented selective-id model and its extensions. *J. UCS*, 13(10):1367–1395. 22
- [Cheng et al. 2007] Cheng, Z., Chen, L., Ling, L., e Comley, R. (2007). General and efficient certificateless public key encryption constructions. In *Pairing*, volume 4575 of *Lecture Notes in Computer Science*, pages 83–107. Springer. 33
- [Chow 2009] Chow, S. (2009). Removing escrow from identity-based encryption - new security notions and key management techniques. In *Public Key Cryptography - PKC 2009*, volume 5443 of *Lecture Notes in Computer Science*, pages 256–276. Springer. 26

- [Chow et al. 2006] Chow, S. S. M., Boyd, C., e Nieto, J. M. G. (2006). Security-mediated certificateless cryptography. In *Public Key Cryptography PKC 2006*, volume 3958 of *Lecture Notes in Computer Science*, pages 508–524, New York, NY, USA. Springer. 34, 35
- [Cocks 2001] Cocks, C. (2001). An identity based encryption scheme based on quadratic residues. In *Proceedings of the 8th IMA International Conference on Cryptography and Coding*, pages 360–363, London, UK. Springer-Verlag. 22, 24
- [Crampton et al. 2007] Crampton, J., Lim, H. W., e Paterson, K. G. (2007). What can identity-based cryptography offer to web services? In *SWS '07: Proceedings of the 2007 ACM workshop on Secure web services*, pages 26–36, New York, NY, USA. ACM. 9
- [Dent 2008] Dent, A. W. (2008). A survey of certificateless encryption schemes and security models. *Int. J. Inf. Secur.*, 7(5):349–377. Cryptology ePrint Archive, Report 2006/211, <http://eprint.iacr.org/>. 31, 34
- [Dent et al. 2008] Dent, A. W., Libert, B., e Paterson, K. G. (2008). Certificateless encryption schemes strongly secure in the standard model. In *Public Key Cryptography - PKC 2008*, volume 4939 of *Lecture Notes in Computer Science*, pages 344–359, Berlin/Heidelberg. Springer. Também disponível em Cryptology ePrint Archive, Report 2007/121. 33, 34
- [Diffie e Hellman 1976] Diffie, P. e Hellman, M. E. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654. 2
- [Dodis e Katz 2005] Dodis, Y. e Katz, J. (2005). Chosen-ciphertext security of multiple encryption. In *TCC*, volume 3378 of *Lecture Notes in Computer Science*, pages 188–209. Springer. 34, 37
- [Fan et al. 2008] Fan, X., Gong, G., e Jao, D. (2008). Speeding up pairing computations on genus 2 hyperelliptic curves with efficiently computable automorphisms. In *Pairing '08: Proceedings of the 2nd international conference on Pairing-Based Cryptography*, pages 243–264, Berlin, Heidelberg. Springer-Verlag. 22
- [Fujisaki e Okamoto 1999] Fujisaki, E. e Okamoto, T. (1999). Secure integration of asymmetric and symmetric encryption schemes. In *CRYPTO '99: Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, pages 537–554, London, UK. Springer-Verlag. 38
- [Galindo et al. 2008] Galindo, D., Morillo, P., e Ràfols, C. (2008). Improved certificate-based encryption in the standard model. *J. Syst. Softw.*, 81(7):1218–1226. 37, 38
- [Gentry 2003] Gentry, C. (2003). Certificate-based encryption and the certificate revocation problem. Cryptology ePrint Archive, Report 2003/183. 16, 19, 26, 36, 38
- [Gentry e Silverberg 2002] Gentry, C. e Silverberg, A. (2002). Hierarchical id-based cryptography. In *ASIACRYPT '02: Proceedings of the 8th International Conference on*

- the Theory and Application of Cryptology and Information Security*, pages 548–566, London, UK. Springer-Verlag. 9, 22
- [Girault 1991] Girault, M. (1991). Self-certified public keys. In *EuroCrypt91*, pages 490–497, Brighton, UK. Springer. LCNS vol.547. 4, 7, 9, 11, 12, 28, 29
- [Goya 2006] Goya, D. H. (2006). Proposta de esquemas de criptografia e de assinatura sob modelo de criptografia de chave pública sem certificado. Dissertação de mestrado, Instituto de Matemática e Estatística, Universidade de São Paulo. Disponível em <http://www.teses.usp.br/teses/disponiveis/45/45134/tde-28072006-142410/>. 31
- [Günther 1989] Günther, C. G. (1989). An identity-based key-exchange protocol. In *EUROCRYPT*, volume 434 of *Lecture Notes in Computer Science*, pages 29–37. Springer. 11
- [Hess 2003] Hess, F. (2003). Efficient identity based signature schemes based on pairings. In *SAC '02: Revised Papers from the 9th Annual International Workshop on Selected Areas in Cryptography*, pages 310–324, London, UK. Springer-Verlag. 24
- [Hu et al. 2007] Hu, B., Wong, D., Zhang, Z., e Deng, X. (2007). Certificateless signature: a new security model and an improved generic construction. *Designs, Codes and Cryptography*, 42(2):109–126. 35
- [Hu et al. 2006] Hu, B. C., Wong, D. S., Zhang, Z., e Deng, X. (2006). Key replacement attack against a generic construction of certificateless signature. In *Information Security and Privacy, 11th Australasian Conference, ACISP 2006*, volume 4058 of *Lecture Notes in Computer Science*, pages 235–246. Springer. 34
- [Hwang et al. 2008] Hwang, Y. H., Liu, J. K., e Chow, S. S. (2008). Certificateless public key encryption secure against malicious kgc attacks in the standard model. *Journal of Universal Computer Science*, 14(3):463–480. 34
- [Joux 2000] Joux, A. (2000). A one round protocol for tripartite diffie-hellman. In *ANTS-IV: Proceedings of the 4th International Symposium on Algorithmic Number Theory*, volume 1838 of *Lecture Notes in Computer Science*, pages 385–394, London, UK. Springer-Verlag. 22
- [Joye e Neven 2009] Joye, M. e Neven, G. (2009). *Identity-based Cryptography*. IOS Press, Amsterdam. 25, 42
- [Kang e Park 2005] Kang, B. G. e Park, J. H. (2005). Is it possible to have cbe from cl-pke? *Cryptology ePrint Archive, Report 2005/431*. 37, 40
- [Kang et al. 2004] Kang, B. G., Park, J. H., e Hahn, S. G. (2004). A certificate-based signature scheme. In *CT-RSA*, volume 2964 of *Lecture Notes in Computer Science*, pages 99–111. Springer. 38
- [Kim et al. 1999] Kim, S., Oh, S., Park, S., e Won, D. (1999). Verifiable self-certified public keys. In *WCC'99 : Workshop on Coding and Cryptography*, pages 139–148, Le Chesnay, França. INRIA. 10

- [Koblitz 1994] Koblitz, N. (1994). *A course in number theory and cryptography*, 2.ed. Springer-Verlag, New York - NY - USA. 20
- [Lee e Kim 2002] Lee, B. e Kim, K. (2002). Self-certified signatures. In *INDOCRYPT '02: Proceedings of the Third International Conference on Cryptology*, pages 199–214, London, UK. Springer-Verlag. 30
- [Li et al. 2007] Li, J., Huang, X., Mu, Y., Susilo, W., e Wu, Q. (2007). Certificate-based signature: Security model and efficient construction. In *EuroPKI*, volume 4582 of *Lecture Notes in Computer Science*, pages 110–125. Springer. 38, 39
- [Libert e Quisquater 2006] Libert, B. e Quisquater, J.-J. (2006). On constructing certificateless cryptosystems from identity based encryption. In *Public Key Cryptography 2006 (PKC'06)*, volume 3958 of *Lecture Notes in Computer Science*, pages 474–490, New York, NY, USA. Springer-Verlag. 33, 34, 40
- [Lim 2006] Lim, H. W. (2006). *On the Application of Identity-Based Cryptography In Grid Security*. Doutorado, University of London. 27
- [Lim e Paterson 2005] Lim, H. W. e Paterson, K. G. (2005). Identity-based cryptography for grid security. In *E-SCIENCE '05: Proceedings of the First International Conference on e-Science and Grid Computing*, pages 395–404, Washington, DC, USA. IEEE Computer Society. 9
- [Liu et al. 2007] Liu, J. K., Au, M. H., e Susilo, W. (2007). Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model. In *ASIACCS '07: Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security*, pages 273–283, New York, NY, USA. ACM. 14, 34, 35
- [Liu et al. 2008] Liu, J. K., Baek, J., Susilo, W., e Zhou, J. (2008). Certificate-based signature schemes without pairings or random oracles. In *ISC '08: Proceedings of the 11th international conference on Information Security*, volume 5222 of *Lecture Notes in Computer Science*, pages 285–297, Berlin, Heidelberg. Springer-Verlag. 39
- [Liu et al. 2009] Liu, J. K., Baek, J., e Zhou, J. (2009). Certificate-based sequential aggregate signature. In *WiSec '09: Proceedings of the second ACM conference on Wireless network security*, pages 21–28, New York, NY, USA. ACM. 39
- [Liu e Zhou 2008] Liu, J. K. e Zhou, J. (2008). Efficient certificate-based encryption in the standard model. In *SCN '08: Proceedings of the 6th international conference on Security and Cryptography for Networks*, pages 144–155, Berlin, Heidelberg. Springer-Verlag. 38
- [Lu e Li 2008] Lu, Y. e Li, J. (2008). A general and secure certification-based encryption construction. In *ChinaGrid'08*, pages 182–189, Los Alamitos, CA. IEEE Computer Society. 38
- [Lu et al. 2009] Lu, Y., Li, J., e Xiao, J. (2009). Constructing efficient certificate-based encryption with pairing. *Journal of Computers*, 4(1):19–26. 38

- [Malone-Lee 2002] Malone-Lee, J. (2002). Identity-based signcryption. *Cryptology ePrint Archive-Report 2002/098*. <http://eprint.iacr.org/2002/098>. 25
- [Mao 2003] Mao, W. (2003). *Modern cryptography : theory and practice*. Prentice Hall. 3
- [McCullagh e Barreto 2004] McCullagh, N. e Barreto, P. S. L. M. (2004). A new two-party identity-based authenticated key agreement. In *In proceedings of CT-RSA 2005, LNCS 3376*, pages 262–274. Springer-Verlag. Também disponível em *Cryptology ePrint Report 2004/122*. 25
- [Misaghi 2008] Misaghi, M. (2008). *Um Ambiente Criptográfico Baseado na Identidade*. Doutorado, Escola Politécnica, Universidade de São Paulo. 9, 25
- [Naccache 2007] Naccache, D. (2007). Secure and practical identity-based encryption. *IET Information Security*, 1(2):59–64. Também disponível em *Cryptology ePrint Report 2005/369*. 22
- [Petersen e Horster 1997] Petersen, H. e Horster, P. (1997). Self-certified keys - concepts and applications. 28, 29
- [Saeednia 2003] Saeednia, S. (2003). A note on girault's self-certified model. *Inf. Process. Lett.*, 86(6):323–327. 11, 28
- [Sakai e Kasahara 2003] Sakai, R. e Kasahara, M. (2003). Id based cryptosystems with pairing on elliptic curve. *Cryptology ePrint Archive, Report 2003/054*. 38
- [Sakai et al. 2000] Sakai, R., Ohgishi, K., e Kasahara, M. (2000). Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security (SCIS2000)*, pages 26–28, Okinawa, Japan. Inst. of Electronics, Information and Communication Engineers. 22
- [Scott et al. 2006] Scott, M., Costigan, N., e Abdulwahab, W. (2006). Implementing cryptographic pairings on smartcards. In *CHES*, volume 4249 of *Lecture Notes in Computer Science*, pages 134–147. Springer. 27
- [Shamir 1984] Shamir, A. (1984). Identity-based cryptosystems and signature schemes. In *Proceedings of CRYPTO 84 on Advances in cryptology*, volume 196/1985 of *Lecture Notes in Computer Science*, pages 47–53, New York, NY, USA. Springer-Verlag New York, Inc. 5, 8, 24
- [Shao 2007] Shao, Z. (2007). Self-certified signatures based on discrete logarithms. In *WAIFI '07: Proceedings of the 1st international workshop on Arithmetic of Finite Fields*, pages 252–263, Berlin, Heidelberg. Springer-Verlag. 30
- [Shao 2008] Shao, Z. (2008). Certificate-based verifiably encrypted signatures from pairings. *Information Sciences*, 178(10):2360–2373. 39
- [Sun et al. 2007] Sun, Y., Zhang, F., e Baek, J. (2007). Strongly secure certificateless public key encryption without pairing. In *CANS*, volume 4856 of *Lecture Notes in Computer Science*, pages 194–208. Springer. 33

- [Szczechowiak et al. 2008] Szczechowiak, P., Oliveira, L. B., Scott, M., Collier, M., e Dahab, R. (2008). Nanoecc: Testing the limits of elliptic curve cryptography in sensor networks. In *European conference on Wireless Sensor Networks, EWSN08*, volume 4913 of *Lecture Notes in Computer Science*, pages 305–320. 9
- [Terada 2008] Terada, R. (2008). *Segurança de Dados - Criptografia em Redes de Computador*. Editora Edgard Blücher, São Paulo, SP, 2 edition. 3
- [Trappe e Washington 2005] Trappe, W. e Washington, L. C. (2005). *Introduction to Cryptography with Coding Theory*. Prentice Hall, 2 edition. 22
- [Wang et al. 2007] Wang, L., Shao, J., Cao, Z., Mambo, M., e Yamamura, A. (2007). A certificate-based proxy cryptosystem with revocable proxy decryption power. In *Progress in Cryptology - INDOCRYPT 2007, 8th International Conference on Cryptology in India, Chennai, India, December 9-13, 2007, Proceedings*, volume 4859 of *Lecture Notes in Computer Science*, pages 297–311. Springer. 39
- [Waters 2005] Waters, B. R. (2005). Efficient identity-based encryption without random oracles. In *EUROCRYPT'05*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer. Também disponível em Cryptology ePrint Report 2004/180. 22, 39
- [Yao et al. 2004] Yao, D., Fazio, N., Dodis, Y., e Lysyanskaya, A. (2004). Id-based encryption for complex hierarchies with applications to forward security and broadcast encryption. In *CCS '04: Proceedings of the 11th ACM conference on Computer and communications security*, pages 354–363, New York - NY - USA. ACM Press. 22
- [Yum e Lee 2004] Yum, D. H. e Lee, P. J. (2004). Identity-based cryptography in public key management. In *EuroPKI 2004*, volume 3093 of *Lecture Notes in Computer Science*, pages 71–84, Samos Island, Greece. Springer-Verlag. 37, 40
- [Zhang e Wang 2008] Zhang, G. e Wang, S. (2008). A certificateless signature and group signature schemes against malicious pkg. In *22nd International Conference on Advanced Information Networking and Applications, AINA 2008*, pages 334–341. IEEE Computer Society. 35
- [Zhang et al. 2006] Zhang, Z., Wong, D. S., XU, J., e FENG, D. (2006). Certificateless public key signature: Security model and efficient construction. In *4th. International Conference on Applied Cryptography and Network Security, ACNS'06*, volume 3989 of *Lecture Notes in Computer Science*, Singapore. Springer. 35
- [Zheng 1997] Zheng, Y. (1997). Digital signcryption or how to achieve cost(signature & encryption) cost(signature) + cost(encryption). In *CRYPTO '97: Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology*, pages 165–179, London, UK. Springer-Verlag. 25