Criptografia de chave pública: novos modelos baseados em emparelhamentos bilineares sobre pontos de curvas elípticas

Danilo Prates de Oliveira e Marco Aurélio Amaral Henriques

***Resumo*— Em 2001, Boneh e Franklin apresentaram o primeiro esquema completo e funcional de *criptografia de chave pública baseada em identidade* (*CCP-ID*), utilizando emparelhamentos bilineares sobre pontos de curvas elípticas. A partir de então, passaram a surgir novos modelos para a criptografia de chave pública, como a *criptografia de chave pública sem certificado* (*CCP-SC*). Este tutorial tem como objetivo descrever de maneira simples os modelos de *CCP-ID* e de *CCP-SC*, bem como apontar suas vantagens e desvantagens em relação à infra-estrutura de chaves públicas tradicional.**

***Palavras-Chave*— criptografia de chave pública, infra-estrutura de chaves públicas, criptografia baseada em identidade, criptografia de chave pública sem certificado.**

***Abstract*— In 2001, Boneh and Franklin presented the first fully practical and secure *identity-based public key cryptography* scheme (*PKC-ID*), using bilinear pairings over elliptic curve points. Since then, new pubic key cryptography models were created, such as *the certificateless public key cryptography* (*PKC-CL*). This tutorial describes *PKC-ID* and *PKC-CL* models in a simple way, presenting their benefits and drawbacks when compared to the traditional public key infrastructure.**

***Keywords*— public key cryptography, public key infrastructure, *PKI*, identity-based encryption, *IBE*, certificateless encryption.**

I. Introdução

A principal dificuldade no desenvolvimento de sistemas seguros de criptografia de chave pública está no gerenciamento da infra-estrutura que garante a autenticidade das chaves criptográficas. Em uma *infra-estutura de chaves públicas* tradicional (*ICP*), esta garantia é fornecida através de certificados emitidos por uma *autoridade certificadora*. A tecnologia da *ICP* encontra-se bem documentada [1] e os seus problemas estão associados com o gerenciamento e o custo computacional da verificação dos certificados.

A *criptografia de chave pública baseada em identidade* (*CCP-ID*), primeiramente proposta por A. Shamir [2] em 1984, se propôs a resolver o problema da autenticidade das chaves de uma maneira diferente: a chave pública de uma entidade seria derivada diretamente de um identificador aleatório como, por exemplo, o endereço de e-mail de um usuário. Neste modelo, a autoridade de confiança é um *gerador de chaves privadas* (*GCPr*), que, em posse de uma *chave-mestra*, gera as chaves privadas do sistema correspondentes às chaves públicas escolhidas. Entretanto, o primeiro esquema prático de *CCP-ID* foi apresentado por Boneh e Franklin em 2001 [3]. A escolha arbitrária das chaves públicas na *CCP-ID* elimina a necessidade de certificados e alguns dos problemas a eles associados. Por outro lado, por conhecer todas as chaves privadas do sistema, o *GCPr* pode forjar assinaturas e decifrar qualquer texto cifrado e, por isso, a *CCP-ID* não evita um eventual repúdio de autoria de mensagens por parte de seus usuários.

Em 2003, foi proposta a *criptografia de chave pública sem certificado* (*CCP-SC*) [4], um modelo intermediário entre *ICP* e a *CCP-ID*, pois não requer o uso de certificados e não permite que a autoridade de confiança, o gerador de chaves privadas parciais (*GCPrP*), tenha acesso às chaves privadas dos usuários.

Neste tutorial, apresentaremos os três modelos e faremos comparações a respeito das vantagens e desvantagens de cada um sem, no entanto, entrar em detalhes de implementação por questões de espaço. Em prol de uma maior simplicidade da apresentação, serão deixadas de lado algumas notações que seriam exigidas por um maior rigor matemático. A seção II mostrará a evolução da criptografia das chaves simétricas até as chaves públicas. Na seção III, será visto o modelo de uma infra-estrutura de chaves públicas tradicional. A seção IV apresentará a criptografia de chave pública baseada em identidade, um esquema de ciframento utilizando este modelo e as suas principais características. Na seção V, será apresentada a criptografia de chave pública sem certificado e o seu respectivo esquema básico de ciframento. Por fim, na seção VI, faremos comparações entre os modelos apresentados e breves recomendações sobre o tipo de ambiente mais adequado a cada um deles.

II. Histórico

A. Criptografia de chave secreta

Até meados da década de 1970, existiam somente as técnicas de criptografia de chave secreta. Elas são baseadas em substituições e permutações de dados controladas por um segredo que o remetente e o destinatário das mensagens compartilham. Por exemplo, em um sistema *GSM (Global System for Mobile)*, existe uma chave secreta que é compartilhada entre o assinante e a operadora do serviço. Esta chave é instalada em um módulo *SIM (Subscriber Identity Module)* e utilizada para prover comunicações seguras entre as partes.

Danilo P. de Oliveira e Marco A. A. Henriques, Faculdade de Engenharia Elétrica e de Computação (FEEC), Universidade Estadual de Campinas (UNICAMP), Campinas, SP, Brasil. E-mails: dprates@dca.fee.unicamp.br, marco@dca.fee.unicamp.br. Trabalho parcialmente financiado pelo CNPq (Proc. 133623/2004-1).

Os três principais problemas da criptografia de chave secreta são:

1) dificuldade de distribuição de chaves secretas de forma segura entre as partes;
2) possibilidade de repúdio de autoria de mensagens enviadas, já que não é possível determinar qual dos detentores da chave secreta cifrou e enviou uma mensagem;
3) baixa escalabilidade, já que, em uma comunicação com n participantes, são necessárias $(n^2 - n)/2$ chaves secretas.

A grande vantagem da criptografia de chave secreta é que ela é computacionalmente mais rápida quando comparada às técnicas de chave pública. Por isso, a maioria dos protocolos criptográficos de hoje ainda utiliza mecanismos de chave secreta. Exemplos de algoritmos de chave secreta são o *DES (Data Encryption Standard)* e o *AES (Advanced Encryption Standard)*.

B. Criptografia de Chave Pública

Em 1976, a fim de solucionar os problemas da criptografia de chave secreta descritos, W. Diffie e M. Hellman [5] propuseram o conceito de criptografia de chave pública. O seu desenvolvimento é considerado uma revolução na história da criptografia, pois ela é assimétrica e envolve o uso de duas chaves diferentes: uma *chave pública*, de domínio público, e uma *chave privada*, de conhecimento exclusivo do seu detentor. Assim, a criptografia de chave pública possibilitou realizar com mais segurança: a distribuição de chaves, a criação de assinaturas digitais (obtidas pelo ciframento da mensagem com a chave privada, o que evita o repúdio da autoria da mesma) e um gerenciamento mais escalável (para uma comunicação segura entre n participantes necessita-se de apenas n pares de chaves). A criptografia de chave pública, diferentemente das ferramentas de substituição e permutação utilizadas anteriormente, é baseada em funções matemáticas conhecidas como funções de mão-única, as quais são simples de se calcular, mas têm funções inversas que são intratáveis do ponto de vista computacional. Exemplos de algoritmos de chave pública utilizados na prática são o *RSA (Rivest-Shamir-Adleman)* e o *ECC (Elliptic Curve Cryptography)*.

III. Infra-estrutura de chaves públicas tradicional

Uma infra-estrutura de chaves públicas tradicional (*ICP*) tem como objetivo garantir a ligação entre uma chave pública e a identidade do detentor da respectiva chave privada. Esta garantia é fornecida através de certificados emitidos por uma terceira parte confiável, cujo papel em uma *ICP* é desempenhado por uma *autoridade certificadora (AC)*. Cabe, portanto, à *AC* emitir certificados, agendar sua data de expiração e publicar a lista dos certificados revogados.

A Figura 1 apresenta o ciclo de vida de um certificado. Basicamente, existem as seguintes etapas:

1) *geração e registro do par de chaves*: primeiramente, uma entidade A gera um par de chaves. Em seguida, a fim de obter um certificado público que associe o seu nome à chave pública gerada, ela precisa se registrar perante a autoridade certificadora, comprovando a sua identidade. Descrito de uma forma simples, um certificado contém o nome da entidade, a sua chave púlica e uma data, todos cifrados com a chave privada da *AC*;
2) *geração e distribuição dos certificados*: uma vez que a identidade de A tenha sido verificada, a *AC* emite e distribui cópias do certificado para a entidade A e para um repositório público de certificados. Além disso, a *AC* também entrega a A uma cópia do seu próprio certificado, pois assim ela poderá verificar a validade dos certificados que receber das outras entidades;
3) *revogação de certificados*: se a chave privada da entidade A for comprometida, a *AC* deve revogar o certificado anteriormente emitido. Neste caso, a *AC* atualiza a lista de certificados revogados (*LCR*);
4) *aquisição de certificados de terceiros*: a entidade B que deseja se comunicar com A obtém o certificado desta entidade a partir dela mesma ou do repositório de certificados, dependendo do protocolo de segurança adotado;
5) *validação dos certificados*: quando a entidade B recebe o certificado de A, ela precisa validá-lo. Para isto, ela verifica se o certificado não se encontra na lista de certificados revogados e se é possível decifrar informações contidas no mesmo com a chave pública da *AC*.

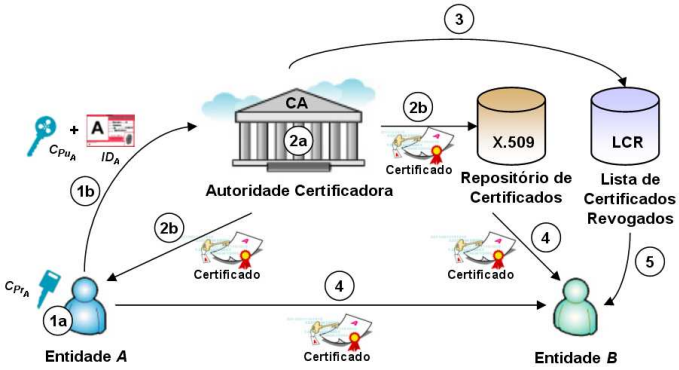

Fig. 1. Modelo básico de uma *ICP*

Em uma *ICP*, pode-se escolher onde o par de chaves será gerado. Assim, as chaves podem ser geradas tanto pela *AC*, quanto pelo próprio usuário. A escolha do mecanismo de geração das chaves deve ser ditada pelas políticas de segurança do sistema. Por exemplo, caso se deseje que uma assinatura suporte não-repúdio, então é melhor que a chave seja gerada pelo usuário. Por outro lado, se a chave será utilizada para manter confidenciais as informações de uma empresa, então pode ser recomendável que a *AC* (que pode ser a própria empresa) gere a chave privada, pois existirá então uma forma de recuperar as informações cifradas pelos funcionários. Quando alguma entidade consegue ter acesso à chave privada de outra entidade, diz-se que houve *key escrow*.

Os problemas relacionados com a *ICP* estão associados com o gerenciamento dos certificados, incluindo revogação, armazenamento, distribuição e o custo computacional da sua

verificação. Estes problemas são ainda mais graves em ambientes restritos, tais como telefones celulares, onde a capacidade de processamento e a largura de banda disponível são limitadas [6].

IV. Criptografia de chave pública baseada em identidade

Em 1984, A. Shamir [2] lançou a idéia de um sistema de criptografia em que a chave pública de uma entidade seria um identificador aleatório que a caracterizasse de forma única, como um número de *CPF*, um endereço eletrônico ou um endereço *IP (Internet Protocol)*. Somente em 2001, Boneh e Franklin [3] propuseram o primeiro sistema completo e funcional de criptografia de chave pública baseada em identidade (*CCP-ID*), utilizando emparelhamentos bilineares de pontos sobre curvas elípticas. Tal fato possibilitou o estabelecimento de comunicações seguras sem a necessidade de haver troca de certificados, nem de se manter um diretório público de chaves.

A principal diferença entre a *CCP-ID* e *ICP* tradicional está na maneira como as chaves são geradas. Na *CCP-ID*, a autoridade de confiança é chamada de gerador de chaves privadas (*GCPr*) e o seu papel é gerar uma chave privada correspondente a uma chave pública previamente escolhida. Para isto, o *GCPr* possui uma chave-mestra que deve ser mantida sempre em segredo. Sendo assim, na *CCP-ID* a antiga *AC* é substituída pelo *GCPr* e a chave privada da *AC* pela chave-mestra do *GCPr*.

Antes de apresentarmos este modelo, precisaremos de algumas definições básicas.

A. Definições básicas

Seja P um ponto sobre uma curva elíptica com a propriedade de gerar um grupo $\mathbb{G}_1$ de q elementos que são múltiplos de P. O emparelhamento $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ é o mapeamento de elementos de $\mathbb{G}_1$ em elementos de um grupo $\mathbb{G}_2$, que possui características distintas de $\mathbb{G}_1$. O emparelhamento e deve possuir as seguintes propriedades:

1) *bilinearidade*: dados os pontos $Q, W \in \mathbb{G}_1$, para quaisquer a e b inteiros pertencentes ao intervalo $[0, q-1]$, onde q é um número primo, temos que

$$\begin{aligned} & e(aQ, bW) = e(abQ, W) = e(Q, abW) \\ = \;& e(bQ, aW) = e(Q, bW)^a = e(aQ, W)^b \\ = \;& e(bQ, W)^a = e(Q, aW)^b = e(Q, W)^{ab}; \end{aligned}$$

2) *não degeneração*: $e(P, P)$ é diferente do elemento identidade de $\mathbb{G}_2$, isto é, diferente do elemento de $\mathbb{G}_2$ que, multiplicado por outro elemento qualquer deste grupo, não o altera;
3) *computabilidade*: o emparelhamento e pode ser calculado em um tempo que viabilize a sua utilização em aplicações práticas.

Os emparelhamentos bilineares utilizados são os emparelhamentos de Weil e de Tate. Ambos são definidos sobre grupos formados por pontos de curvas elípticas, mas o emparelhamento de Tate é considerado mais eficiente que o de Weil. Nos criptossistemas baseados em identidade, o cálculo do emparelhamento em curvas elípticas é a operação de mais alto custo computacional e, portanto, fazê-lo com eficiência é essencial para um bom desempenho. A referência [7] apresenta uma descrição mais completa a respeito dos parâmetros que devem ser selecionados na prática para oferecer eficiência e segurança no cálculo do emparelhamento de Tate.

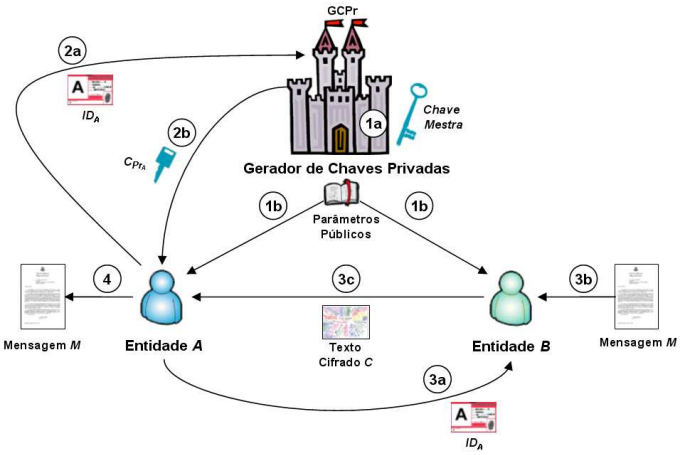

Fig. 2. Esquema de ciframento baseado em identidade.

A segurança de um criptossistema de chave pública baseado em identidade consiste na intratabilidade do problema de Diffie-Hellman Bilinear descrito a seguir.

Problema de Diffie-Hellman Bilinear (BDHP): é computacionalmente inviável calcular $e(P, P)^{abc} \in \mathbb{G}_2$, dados P, aP, bP e cP, onde a, b e c são inteiros, escolhidos aleatoriamente no intervalo $[1, q-1]$.

B. Ciframento baseado em identidade

Um esquema de ciframento baseado em identidade (Figura 2) pode ser resumido nas seguintes etapas.

1) *Setup*. Nesta etapa, o *GCPr* estabelece os parâmetros do sistema, $params = \langle \mathbb{G}_1, \mathbb{G}_2, e, P, P_0, H_1, H_2 \rangle$, e uma chave-mestra s inteira pertencente ao intervalo $[1, q-1]$. Os parâmetros do sistema são públicos e a chave-mestra é mantida em segredo pelo *GCPr*. Descrevendo em mais detalhes, o *GCPr* realiza os seguintes passos:
   - seleciona os grupos $\mathbb{G}_1$ e $\mathbb{G}_2$;
   - escolhe um ponto P que gere $\mathbb{G}_1$;
   - seleciona aleatoriamente uma chave-mestra s e calcula $P_0 = sP$. Observe que dados P_0 e P é inviável na prática obter s (problema do logaritmo discreto sobre curvas elípticas);
   - escolhe as funções de *hash* H_1, que mapeia uma cadeia de bits de comprimento arbitrário em um elemento de $\mathbb{G}_1$, e H_2, que mapeia um elemento de $\mathbb{G}_2$ em uma cadeia de bits de mesmo comprimento da mensagem a ser cifrada.
2) *Criação da chave privada*. A partir dos parâmetros do sistema, $params$, da chave-mestra, s, e de um identificador para a entidade A, ID_A, o *GCPr* gera a chave privada C_{Pr_A}, da seguinte forma:
   - calcula $Q_A = H_1(ID_A)$;

- calcula a chave privada $C_{Pr_A} = sQ_A$.

A chave privada é entregue pelo *GCPr* à entidade A, pessoalmente ou utilizando um canal seguro já existente. A entidade A pode verificar a validade da chave privada, checando se $e(C_{Pr_A}, P) = e(Q_A, P_0)$.

3) *Ciframento*. A entidade B cifra a mensagem M utilizando o identificador da entidade A, ID_A. Mais detalhadamente, B realiza o seguinte:
   - calcula $Q_A = H_1(ID_A)$;
   - escolhe um inteiro r aletoriamente no intervalo $[1, q-1]$;
   - calcula $U = rP$;
   - calcula $V = M \oplus H_2(e(Q_A, P_0)^r)$, onde $\oplus$ denota a operação de ou-exclusivo bit a bit;
   - envia o texto cifrado resultante $C = (U, V)$ para a entidade A.
4) *Deciframento*. Após o recebimento do texto cifrado C, vindo da entidade B, a entidade A o decifra, a partir da sua chave privada C_{Pr_A}, utilizando a seguinte expressão:

$$M = V \oplus H_2(e(C_{Pr_A}, U))$$

Observe que:

$$\begin{aligned} & V \oplus H_2(e(C_{Pr_A}, U)) \\ = \; & V \oplus H_2(e(sQ_A, rP)) \\ = \; & V \oplus H_2(e(Q_A, sP)^r) \\ = \; & V \oplus H_2(e(Q_A, P_0)^r) \\ = \; & M \oplus H_2(e(Q_A, P_0)^r) \oplus H_2(e(Q_A, P_0)^r) = M. \end{aligned}$$

C. Características gerais da CCP-ID

1) *Key escrow.* O *GCPr* inevitavelmente tem conhecimento de todas as chaves privadas geradas por ele. Sendo assim, o *GCPr* pode decifrar qualquer texto cifrado ou forjar assinaturas digitais. Esta característica, como foi dito na seção III, pode ser ou não desejável, dependendo do ambiente, mas, de qualquer forma, fica evidente que a *CCP-ID* não é capaz de oferecer a propriedade de não-repúdio de autoria da mesma maneira que uma *ICP*.
2) *Revogação.* Seja a seguinte situação: uma chave pública é gerada, utilizada, e depois removida ou revogada. Em algum momento futuro, é possível que um outro cliente solicite uma chave com o mesmo identificador por uma razão legítima (por exemplo, os usuários possuem o mesmo nome). Neste caso, os dois usuários irão compartilhar o mesmo par de chaves pública e privada, pois a chave pública está intrinsecamente relacionada com a identidade. Dessa maneira, como não se pode revogar a identidade de uma pessoa, torna-se necessário adicionar entradas adicionais ao processo de geração de chaves. Por isso a identidade não pode ser considerada a única determinante da chave pública de um usuário. Por exemplo, informações como o cargo do indivíduo em uma organização e o período de validade para as chaves podem ser incluídos nos dados que serão utilizados na chave pública.
3) *Geração das chaves em instantes diferentes.* Na *CCP-ID*, as chaves pública e privada podem ser geradas em instantes diferentes. Isto pode ser bom, pois uma entidade B pode cifrar uma mensagem para A utilizando um identificador de A, ID_A, e mais alguma condição que A deva satisfazer antes de o *GCPr* entregar a respectiva chave privada para A. Por exemplo, B pode cifrar uma mensagem utilizando a seguinte chave pública: *"Esta mensagem somente poderá ser decifrada por ID_A quando A tiver mais de 18 anos"*. Sendo assim, para que A obtenha a respectiva chave privada do *GCPr*, ela terá que se autenticar e provar que a condição imposta por B foi satisfeita. Este tipo de aplicação não pode ser suportada por uma *ICP* tradicional. Por outro lado, esta geração em instantes diferentes possui o seguinte problema: antes do uso da chave pública, o *GCPr* não necessariamente validou a ligação entre a chave pública e a entidade A. Por exemplo, B pode utilizar uma informação que ela pensa que é válida como chave pública de A, mas esta informação pode levar a uma outra entidade ou ser completamente inválida.
4) *Efeitos desastrosos em caso de comprometimento da chave-mestra.* Caso a chave-mestra do *GCPr* seja comprometida, os efeitos seriam absolutamente desastrosos em um sistema de *CCP-ID* e também mais severos do que o comprometimento da chave privada de uma *AC* em uma *ICP*. Isto ocorre, porque a chave-mestra da *GCPr* pode ser utilizada para computar todas as chaves privadas do sistema. Sendo assim, um adversário que possua a chave-mestra da *GCPr* poderá ler todas as comunicações cifradas anteriormente e produzir assinaturas válidas para qualquer entidade sem a necessidade de utilizar novos identificadores. Observe que em uma *ICP* o comprometimento da chave privada de uma *AC* causaria a revogação imediata de todos os certificados emitidos por ela. Entretanto, isto afetaria somente a confiança nos certificados a serem emitidos futuramente e todo o tráfego cifrado ou assinado anteriormente ao comprometimento continuaria válido e protegido.

V. Criptografia de chave pública sem certificado

Em 2003, Al-Riyami e Paterson [4] propuseram um novo modelo para suportar o uso de criptografia de chave pública. A principal característica deste modelo é que ele elimina completamente a necessidade de certificados e por isso foi chamado de *criptografia de chave pública sem certificado* (*CCP-SC*). A técnica baseia-se no fato de que a chave privada de uma entidade pode ser formada em dois estágios. No primeiro estágio, um *gerador de chaves privadas parciais* (*GCPrP*), a autoridade de confiança neste modelo, gera uma *chave privada parcial* para uma entidade A, a partir do identificador de A e de uma chave-mestra do sistema. No segundo estágio, a entidade A produz suas chaves pública e privada a partir da combinação entre a chave privada parcial e um segredo que somente ela conhece. Então, é divulgada a chave pública correspondente à chave privada gerada e uma entidade B, usando como entradas o identificador e a chave pública de A, tem, de maneira implícita, a garantia de que uma mensagem cifrada com essas entradas só poderá ser decifrada pela entidade A.

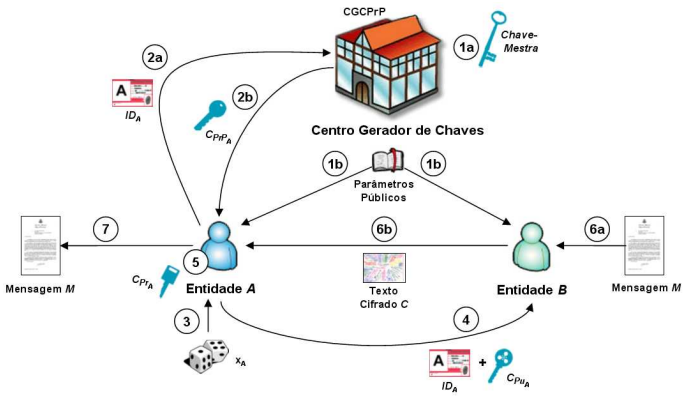

Fig. 3. Esquema de ciframento por chave pública sem certificados

A base da segurança da *CCP-SC* encontra-se no problema computacional descrito a seguir.

Problema de Diffie-Hellman Bilinear Generalizado (GBDHP): é computacionalmente inviável calcular um par $Q \in \mathbb{G}_1^*$ e $e(P,Q)^{abc} \in \mathbb{G}_2$, dados P, aP, bP e cP, onde a, b e c são inteiros escolhidos aleatoriamente no intervalo $[1, q-1]$.

A. Esquema de ciframento por chave pública sem certificados

A seguir é apresentado um esquema básico de ciframento por chave pública sem certificados (Figura 3), que pode ser dividido nas seguintes etapas.

1) *Setup.* Etapa idêntica à anteriormente descrita para o esquema de ciframento baseado em identidade (*CCP-ID*).
2) *Criação da chave privada parcial.* A partir dos parâmetros do sistema, $params$, da chave-mestra, s, e de um identificador para a entidade A, ID_A, o *GCPrP* gera uma chave privada parcial C_{PrP_A} da seguinte forma:
   - calcula $Q_A = H_1(ID_A)$;
   - calcula a chave privada parcial $C_{PrP_A} = sQ_A$.

   A chave privada parcial, C_{PrP_A}, é entregue à entidade A, pessoalmente ou utilizando um canal seguro já existente. Note que a entidade A pode verificar a validade da chave privada parcial checando se $e(C_{PrP_A}, P) = e(Q_A, P_0)$.
3) *Estabelecimento de um valor secreto.* A entidade A, de posse dos parâmetros do sistema e do seu identificador ID_A, gera aleatoriamente um inteiro secreto x_A pertencente ao intervalo $[1, q-1]$, cujo conhecimento deve ficar restrito somente à entidade A.
4) *Estabelecimento da chave pública.* A partir dos parâmetros do sistema e do valor secreto x_A, a entidade A constrói e divulga sua chave pública C_{Pu_A} como sendo $C_{Pu_A} = \langle X_A, Y_A \rangle$, onde $X_A = x_A P$ e $Y_A = x_A P_0 = x_A sP$.
5) *Estabelecimento da chave privada.* Utilizando como entrada os parâmetros do sistema $params$, a chave privada parcial C_{PrP_A} e o valor secreto x_A, a entidade A transforma C_{PrP_A} em uma chave privada completa, denominada C_{Pr_A}, calculando $C_{Pr_A} = x_A C_{PrP_A} = x_A sQ_A$.
6) *Ciframento*. Tendo como entrada os parâmetros do sistema, $params$, a mensagem M, a chave pública C_{Pu_A} e o identificador ID_A, a entidade B obtém o texto cifrado C ou uma indicação de falha no ciframento. Uma falha ocorre toda vez que C_{Pu_A} não está em uma forma coerente. Dessa maneira, para cifrar uma mensagem, B procede da seguinte maneira:
   - checa C_{Pu_A}, verificando se X_A e $Y_A \in \mathbb{G}_1$ e se a igualdade $e(X_A, P_0) = e(Y_A, P)$ é verdadeira. Caso negativo, retorna erro;
   - calcula $Q_A = H_1(ID_A)$;
   - escolhe um valor inteiro aleatório r, pertencente ao intervalo $[1, q-1]$;
   - calcula o texto cifrado $C = (U, V)$, onde $U = rP$ e $V = M \oplus H_2(e(Q_A, Y_A)^r)$.
7) *Decifrамento.* A partir dos parâmetros do sistema, do texto cifrado C e da chave privada C_{Pr_A}, a entidade A obtém a mensagem M calculando:

$$M = V \oplus H_2(e(C_{Pr_A}, U)).$$

Observe que:

$$\begin{aligned} & V \oplus H_2(e(C_{Pr_A}, U)) \\ = \; & V \oplus H_2(e(x_A sQ_A, rP)) \\ = \; & V \oplus H_2(e(Q_A, x_A sP)^r) \\ = \; & V \oplus H_2(e(Q_A, Y_A)^r) \\ = \; & M \oplus H_2(e(Q_A, Y_A)^r) \oplus H_2(e(Q_A, Y_A)^r) = M. \end{aligned}$$

B. Características gerais da CCP-SC

1) *Não é baseada em identidade.* A *CCP-SC* combina elementos da *CCP-ID* e da *ICP*. O sistema não é baseado em identidade, pois a chave pública não é mais diretamente derivada de uma identidade apenas. Por outro lado, como já foi visto na seção IV, na prática, não é recomendável fazer com que um chave pública dependa somente de uma identidade, devido ao problema de multiplicidade de entidades para uma mesma identidade.
2) *Sem key escrow*. A *CCP-SC* não permite o *key escrow* inerente à *CCP-ID*, pois existem informações utilizadas no processo de geração da chave privada que somente a entidade detentora da chave privada conhece. Além disso, a *CCP-SC* não necessita de certificados para garantir a confiança em uma chave pública, já que é possível garantir a autenticidade desta chave de forma implícita.
3) *Revogação*. A revogação em sistemas de *CCP-SC* pode ser realizada de maneira muito semelhante à *CCP-ID*. Neste caso, a idéia é colocar períodos de validade associados aos identificadores das entidades, o que fará com que a chave privada parcial, fornecida pelo *GCPrP*, e conseqüentemente a chave privada completa C_{Pr_A}, possuam vida limitada. Uma alternativa é revogar o identificador ou a chave pública de uma entidade através de uma lista de identificadores revogados, assim como

TABELA I
COMPARAÇÃO ENTRE OS MODELOS APRESENTADOS

Característica	*ICP*	*CCP-ID*	*CCP-SC*
Sem certificados	Não	Sim	Sim
Pode gerar chaves em instantes distintos	Não	Sim	Sim
Provê não-repúdio das assinaturas	Sim	Não	Sim
Chave pública é escolhida ao acaso	Não	Sim	Não
Nível de Confiança necessário na autoridade	Baixo	Alto	Médio

ocorre em uma *ICP*. É importante notar que, mesmo assim, o consumo de banda e de memória será inferior aos métodos que utilizam certificado, porque o volume de informação contida em uma chave pública ou em um identificador é inferior ao contido em um certificado.

4) *Geração de chaves em instantes distintos*. Assim como acontece na *CCP-ID*, na *CCP-SC* as chaves pública e privada de uma entidade podem ser geradas em instantes diferentes, o que permite adicionar aos identificadores condições que devam ser conferidas pelo *GCPrP* antes de entregar a chave privada parcial à entidade.
5) *Interoperabilidade com implementações de CCP-ID*. Os esquemas de *CCP-SC*, assim como os de *CCP-ID*, utilizam-se de emparelhamentos bilineares como infra-estrutura básica para a sua implementação. Portanto, ambos podem coexistir em um sistema sem exigir para isso maiores esforços no desenvolvimento de software.

VI. COMPARAÇÕES

A tabela I resume e compara os modelos apresentados. É possível observar que o nível de confiança que a entidade deve possuir na autoridade varia. Por exemplo, em uma *ICP*, caso o par de chaves tenha sido gerado pelo usuário, o máximo que uma autoridade certificadora maliciosa pode fazer é emitir certificados falsos, ligando a identidade do usuário a uma chave pública diferente daquela gerada por ele. Entretanto, este tipo de ataque da própria parte confiável pode ser facilmente detectado, pois, ao se consultar um repositório de certificados mantido fora da *AC*, será possível verificar que existem dois certificados emitidos para uma única entidade, o que, na prática, só pode acontecer caso um deles já tenha sido revogado. Dessa forma, em uma *ICP*, o nível de confiança que as entidades precisam ter na autoridade certificadora é relativamente baixo.

No caso da *CCP-ID*, como o gerador de chaves privadas possui acesso irrestrito às chaves privadas do sistema, podendo forjar assinaturas e decifrar mensagens, percebe-se que o nível de confiança no gerador de chaves privadas precisa ser alto. Já na *CCP-SC*, o nível de confiança na autoridade é intermediário aos anteriores. Isso acontece porque o centro gerador de chaves gera somente as *chaves privadas parciais* das entidades, não tendo acesso às chaves privadas das entidades. Por outro lado, caso o *GCPrP* divulgue maliciosamente uma chave parcial para um terceiro, este poderá se passar pela entidade real e isto, diferentemente da *ICP*, não pode ser detectado facilmente.

Neste tutorial, apresentamos somente os esquemas de cifra-mentos dos respectivos modelos, mas é importante ressaltar que tanto a *CCP-ID* quanto a *CCP-SC* também comportam esquemas de assinaturas digitais. Além disso, é possível fazer pequenas modificações no esquema de ciframento da *CCP-SC*, a fim de diminuir o nível de confiança necessário na autoridade. Para isto, basta fazer com que a chave privada parcial, gerada pelo *GCPrP*, dependa tanto do identificador da entidade quanto da chave pública gerada por ela. Para maiores informações a respeito desta e outras modificações que podem ser aplicadas ao modelo de *CCP-SC*, consulte a referência [8].

A partir das características de cada modelo, pode-se afirmar que a *CCP-ID* é recomendada para ambientes fechados, como uma empresa, onde possuir um centro gerador de chaves que conhece todas as chaves privadas pode ser necessário. Além disso, como a chave pública está diretamente relacionada ao identificador da entidade, não há necessidade de certificados, o que viabiliza implementações mais leves. Já a *CCP-SC* parece ser ideal para sistemas nos quais o *key escrow* é inaceitável e o peso do gerenciamento de certificados traz problemas de desempenho. Portanto, a *CCP-SC* deve se ajustar bem a uma aplicação de comércio eletrônico para ambientes móveis, onde as assinaturas digitais são necessárias para garantir o não-repúdio da autoria das mensagens enviadas.

VII. CONCLUSÕES

Assim como a criptografia de chave secreta possui diferenças básicas em relação à criptografia de chave pública, existem também diferenças básicas entre os modelos de criptografia de chaves públicas apresentados. Entretanto, não é possível afirmar que uma tecnologia seja sempre melhor do que outra. Na verdade, isto depende do ambiente onde elas serão aplicadas. Dessa maneira, provavelmente estes novos modelos apresentados não irão substituir completamente a infra-estrutura de chaves públicas (*ICP*) tradicional, mas sim levar a criptografia de chave pública para novas e diferentes aplicações.

REFERÊNCIAS

[1] C. Adams, S. Lloyd, *Understanding public-key infrastructure: concepts, standards and deployment considerations*, Macmillan Technical Publishing, Indianapolis, USA, 1999.
[2] A. Shamir, *Identity-based cryptosystems and signature schemes*, Advances in Cryptology - CRYPTO '84, volume 196 of LNCS, pages 47-53. Springer-Verlag, 1984.
[3] D. Boneh, M. Franklin, *Identity-based Encryption from the Weil pairing*, Advances in Cryptology, Asiancrypt'2001, volume 2248, Lecture Notes on Computer Science (LNCS), Springer-Verlag, 2001.
[4] S.S. Al-Riyami, K.G. Paterson, *Certificateless public key cryptography*, Cryptology ePrint Archive, Report 2003/126, 2003. http://eprint.iacr.org/.
[5] W. Diffie, M. Hellman, *New directions in cryptografy*, IEEE Transactions on Information Theory, IT-22(6):644-654, 1976.
[6] J. Dankers, T. Garefalakis, R. Schaffelhofer, T. Wright, *Public key infrastructure in mobile systems*, IEEE Eletronics and Comunication Engineering Journal, 14(5):180-190, 2002.
[7] P.S.L.M. Barreto, B. Lynn, M. Scott, *Efficient algorithms for pairing-based cryptosystems*, volume 2442, Lecture Notes on Computer Science (LNCS), Springer-Verlag, 2002.
[8] S.S. Al-Riyami, *Cryptographic schemes based on elliptic curve pairings*, tese de doutorado, Information Security Group, Department os Mathematics, Royal Holloway, University of London, 2004.