Acordo de Chave Seguro contra Autoridade Mal Intencionada

Denise Goya, Dionathan Nakamura e Routo Terada

Depto Ciência da Computação – IME – USP

SBSeg 2011

Objetivos

- Apresentar uma extensão de modelo para acordo de chave com autenticação sem certificado
- Apresentar protocolo seguro nesse modelo estendido

Roteiro

Roteiro

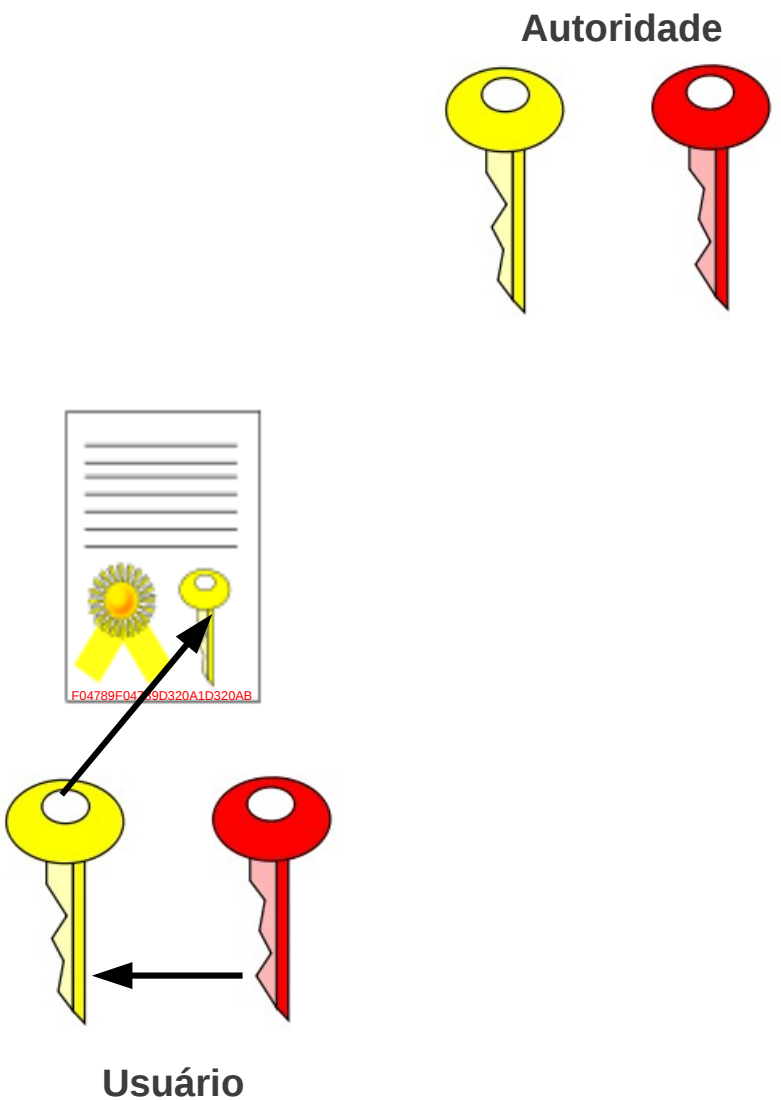
Criptografia de Chave Pública

- Chave pública sob infraestrutura de chaves públicas (ICP)

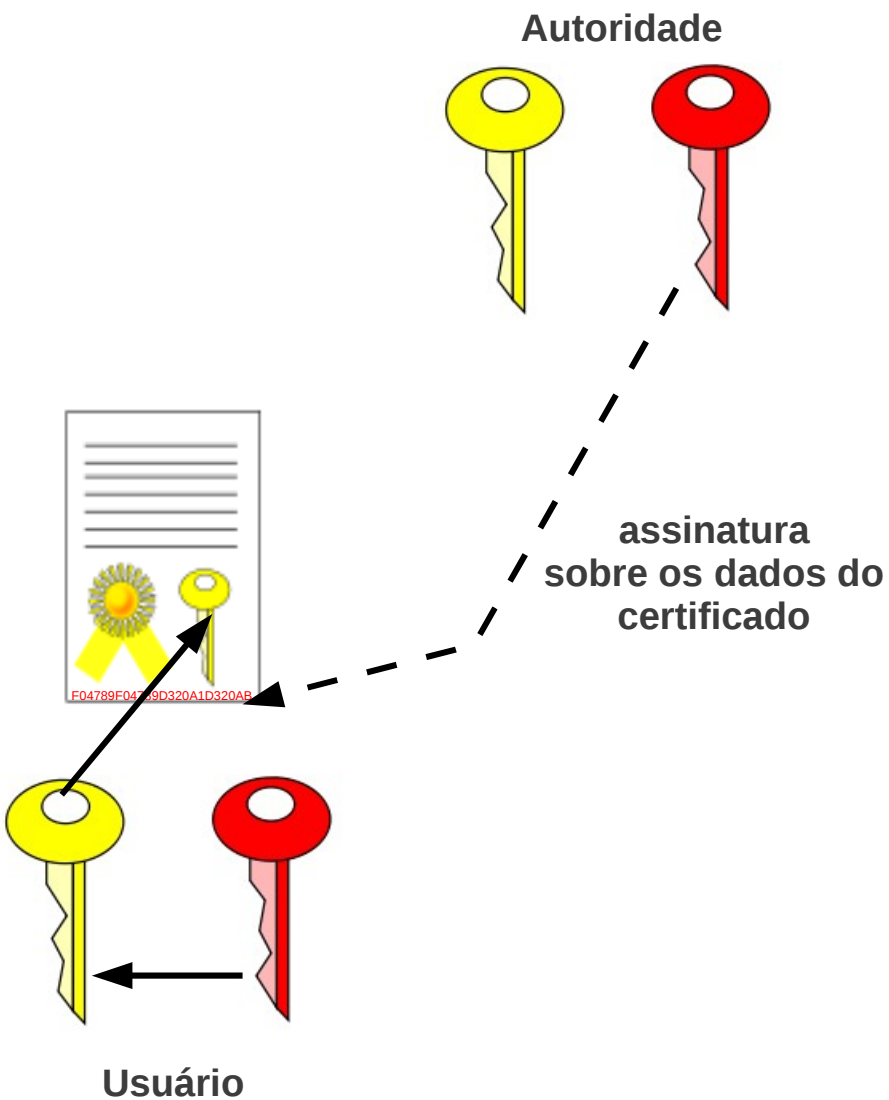
versus

- Modelo sem certificados (*certificateless*): variante de *id-based*

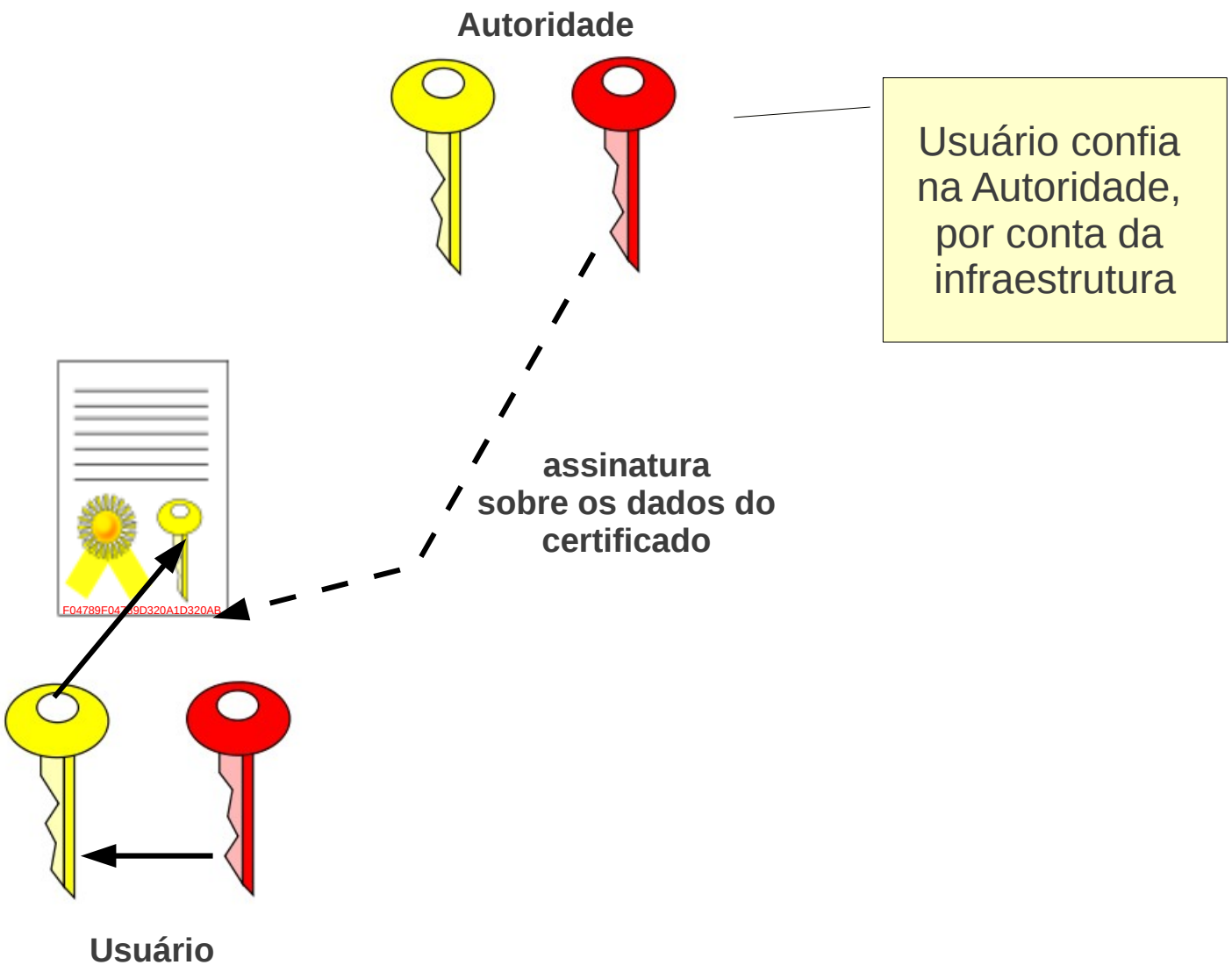
Usuário e Autoridade em uma ICP



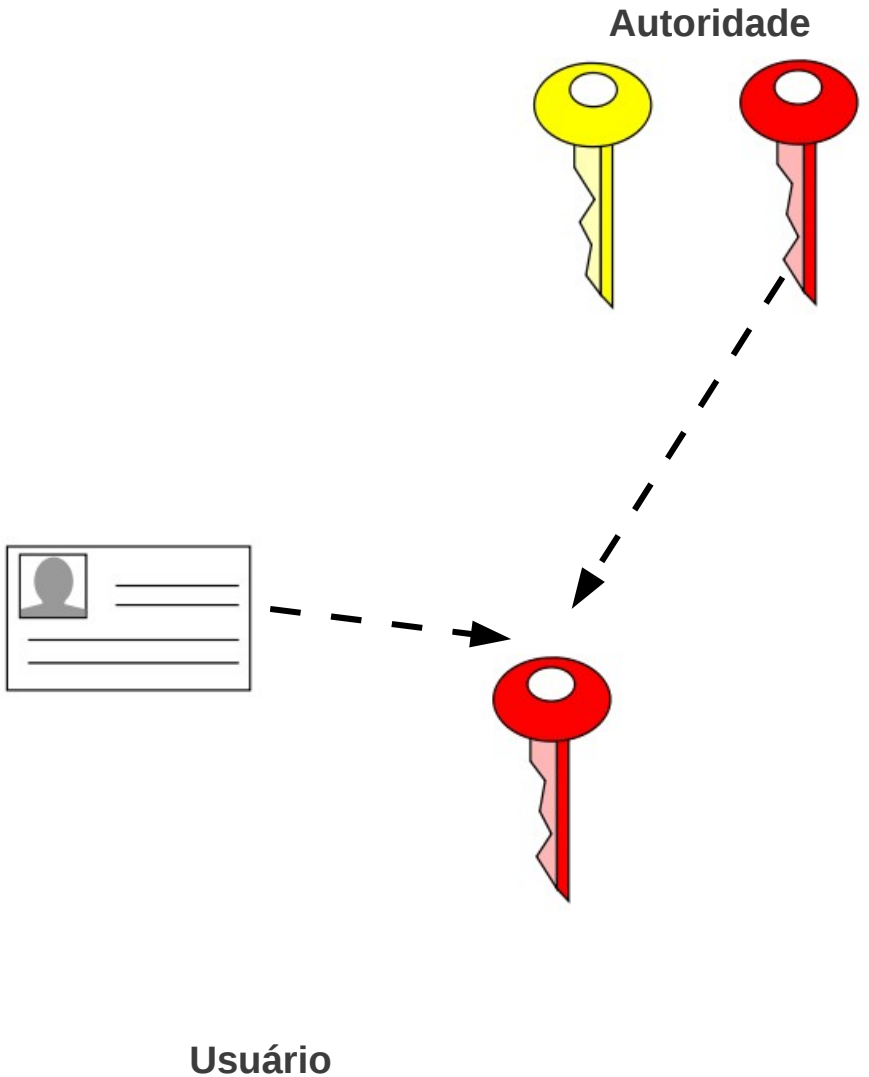
Usuário e Autoridade em uma ICP



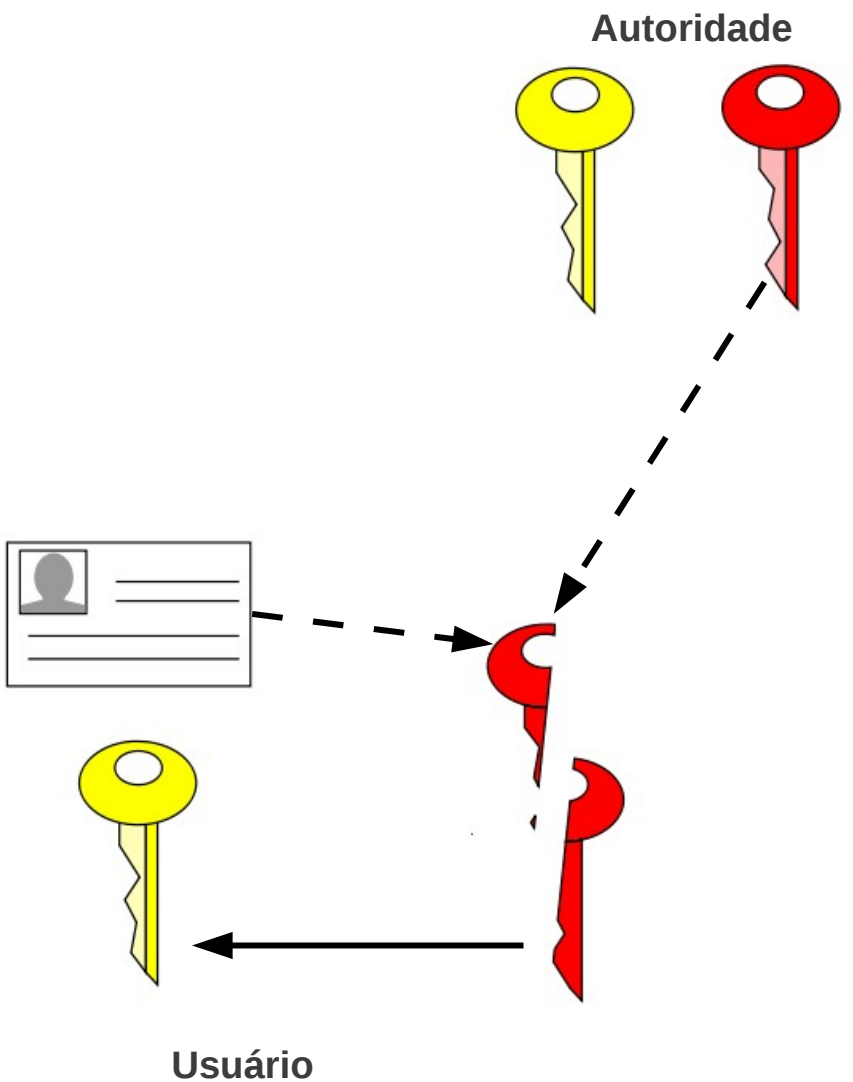
Usuário e Autoridade em uma ICP



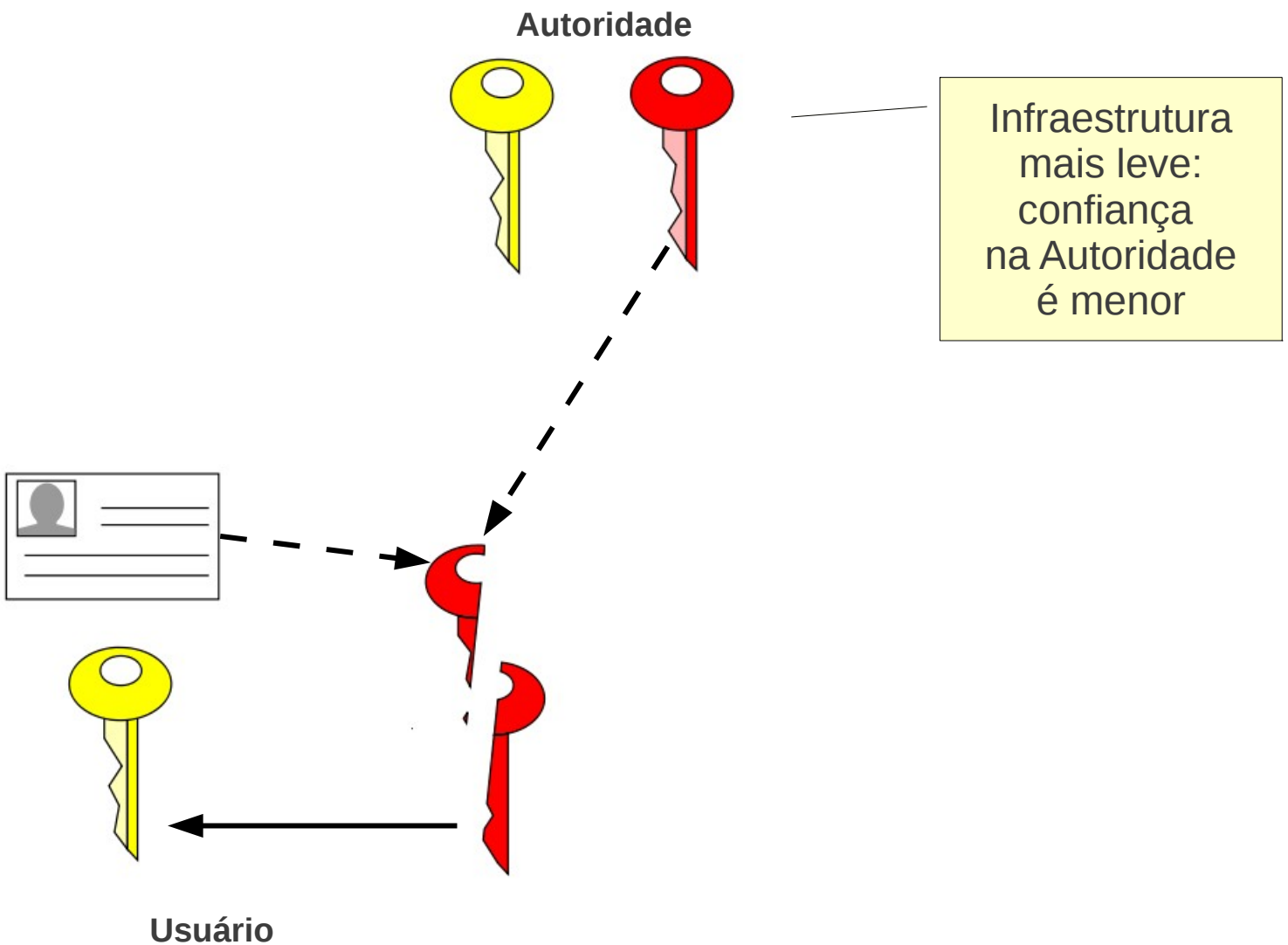
Usuário e Autoridade sob Modelo Baseado em Identidade



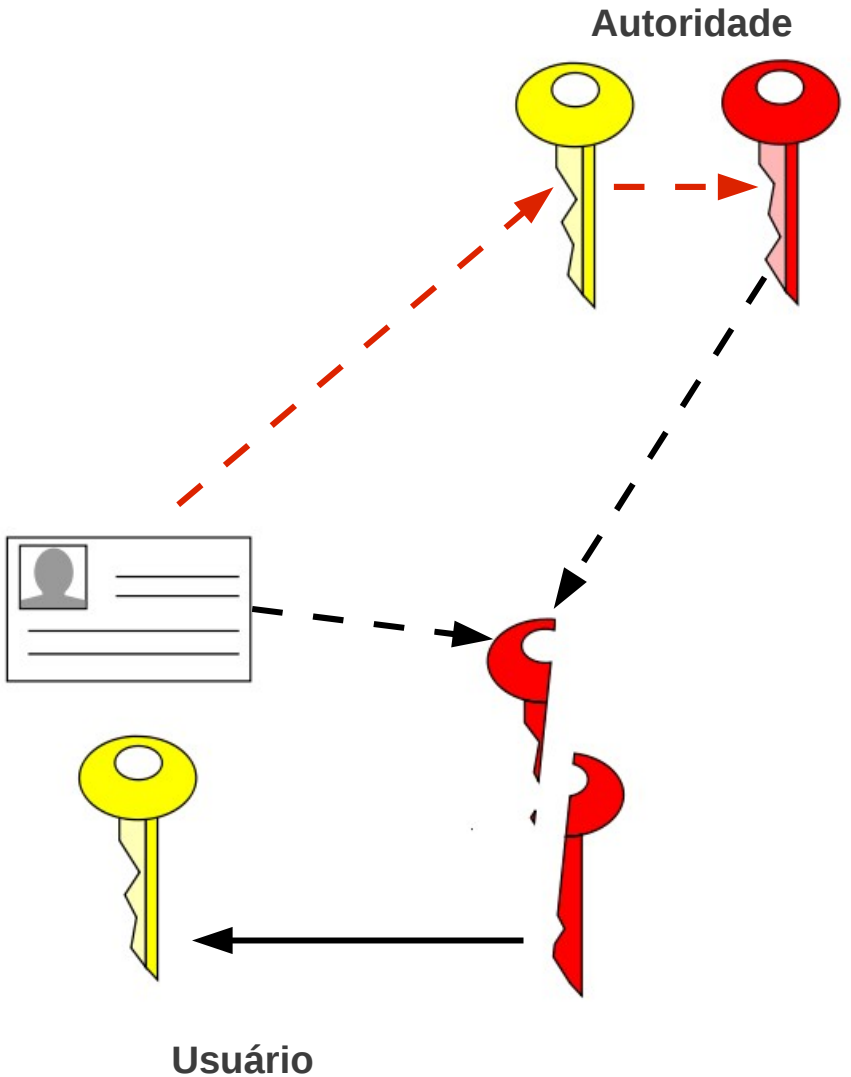
Usuário e Autoridade sob Certificateless



Usuário e Autoridade sob Certificateless



Autoridade Mal Intencionada sob Certificateless



Autoridade Mal Intencionada sob Certificateless

Au et al., 2007

- apresentam o caso da autoridade desonesta que cria parâmetros do sistema com atalhos para falsificar assinaturas ou decifrar mensagens
- estendem modelos de segurança e mostram protocolos seguros

Autoridade Mal Intencionada sob Certificateless

Au et al., 2007

- apresentam o caso da autoridade desonesta que cria parâmetros do sistema com atalhos para falsificar assinaturas ou decifrar mensagens
- estendem modelos de segurança e mostram protocolos seguros

Acordo de chave sem certificado: em aberto

Autoridade Mal Intencionada sob Certificateless

Au et al., 2007

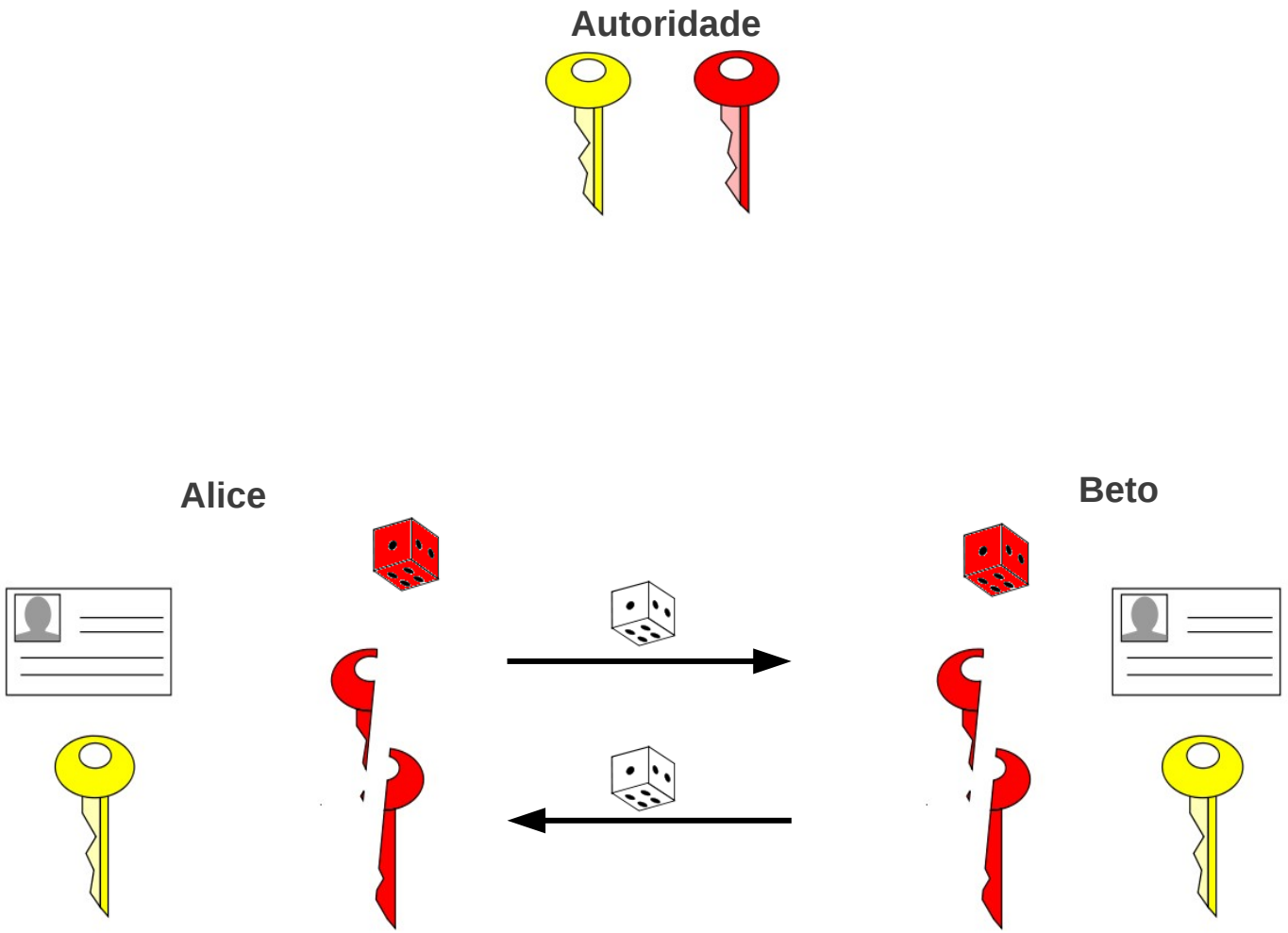
- apresentam o caso da autoridade desonesta que cria parâmetros do sistema com atalhos para falsificar assinaturas ou decifrar mensagens
- estendem modelos de segurança e mostram protocolos seguros

Acordo de chave sem certificado: em aberto $\Longleftarrow$

Roteiro

Acordo de Chave com Autenticação sem Certificado

Certificateless Authenticated Key Agreement (CL-AKA)

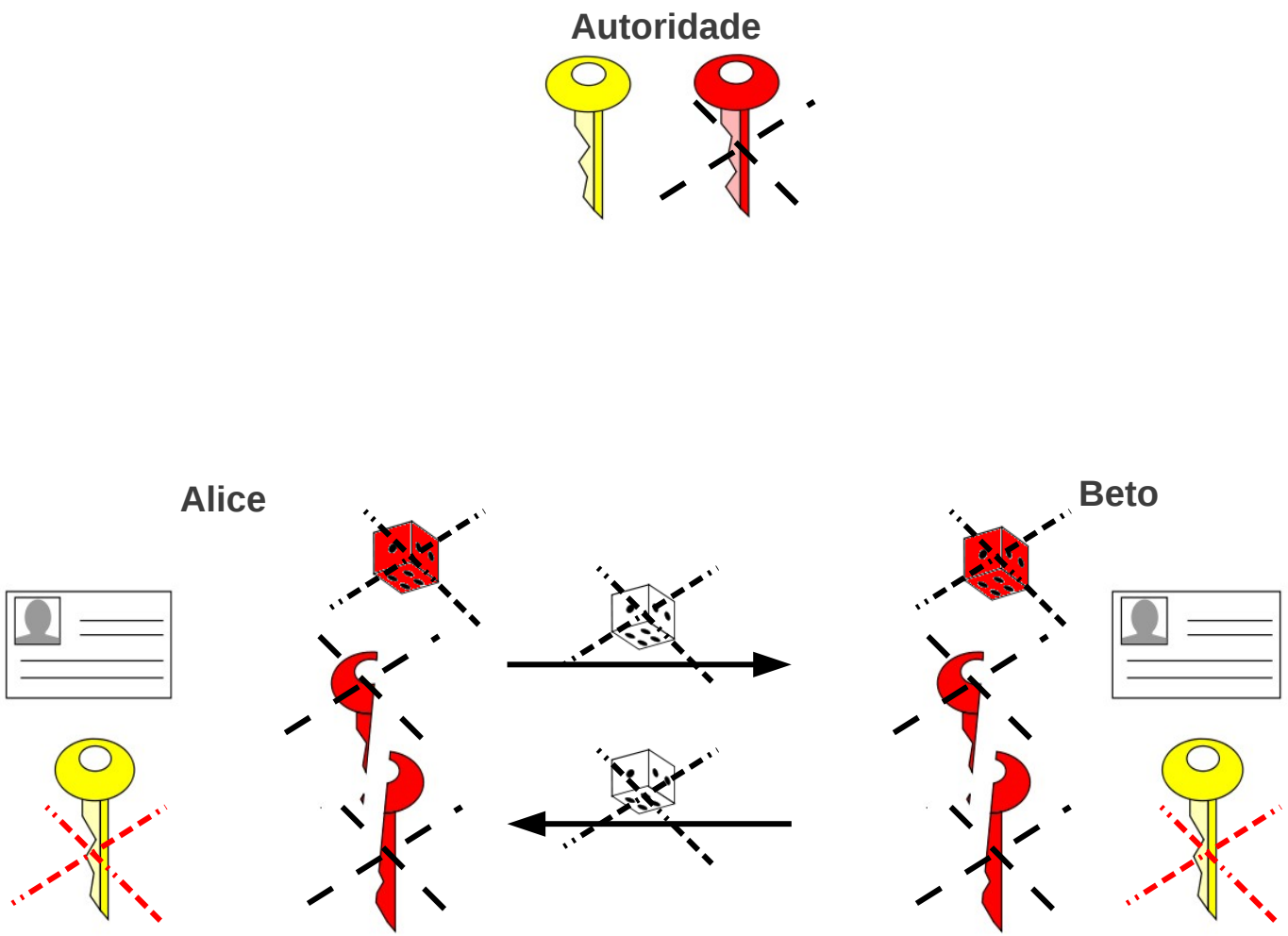


Modelo de Segurança CL-AKA – Poder do Adversário

Adversário pode:

- substituir a chave pública de um dado usuário ou revelar o valor secreto correspondente
- revelar a chave secreta parcial de determinados usuários ou revelar a chave mestra secreta
- revelar o segredo temporário de uma dada sessão ou escolhê-lo ativamente
- revelar a chave secreta de uma dada sessão
- interagir de forma adaptativa com o protocolo, iniciando sessões ou registrando novos usuários arbitrariamente

Modelo de Segurança CL-AKA – Poder do Adversário



Modelo de Segurança CL-AKA – Objetivo Adversário

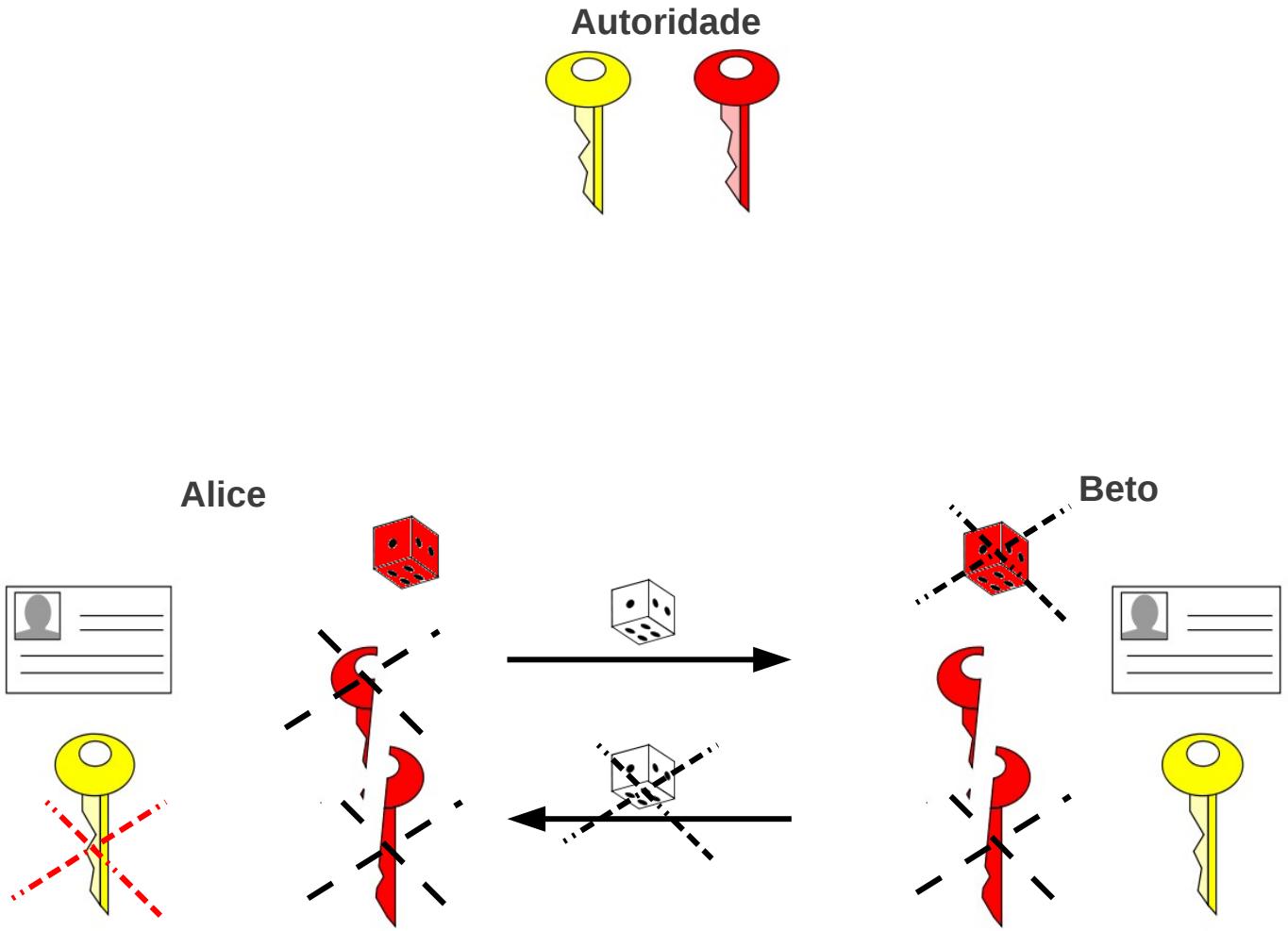
Objetivo do adversário:

- Diferenciar uma chave aleatória de uma verdadeira chave de sessão

Tecnicamente, o adversário pode:

- comprometer arbitrariamente usuários do sistema (com restrições para dois deles: A e B)
- revelar arbitrariamente chaves de sessão e segredos temporários de sessão (com restrições para a sessão de Teste, entre A e B)

Modelo de Segurança CL-AKA – Sessão *Fresh*



Modelo Estendido

Adversário:

- Gera os parâmetros do sistema
- Pode desconhecer o valor da chave mestra secreta e das chaves parciais secretas!

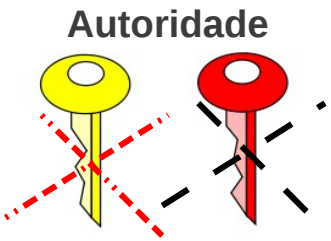
Modelo Estendido

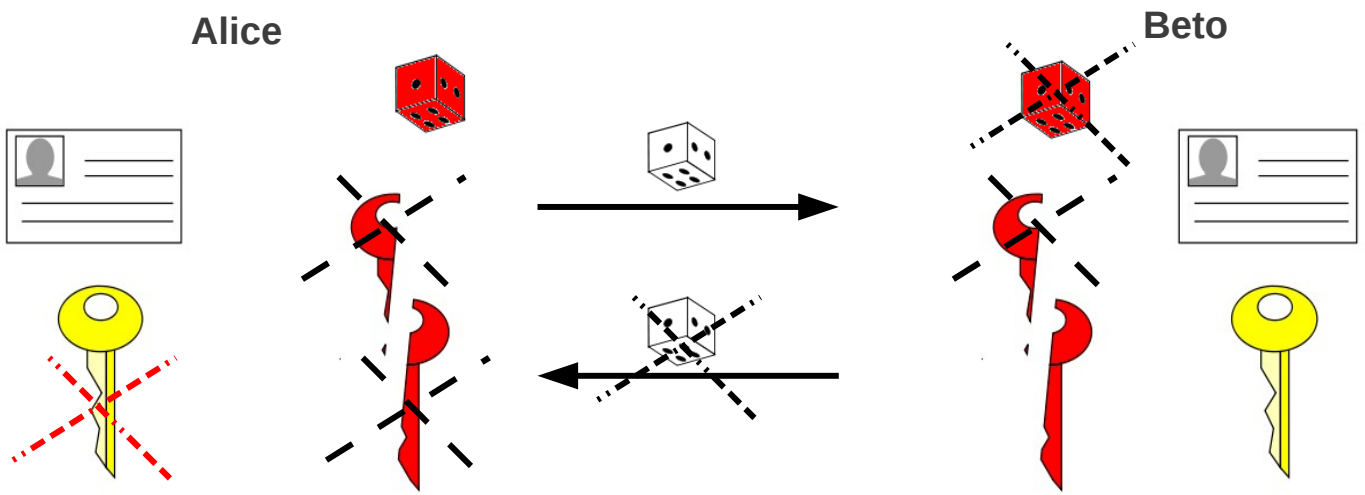
Adversário:

- Gera os parâmetros do sistema
- Pode desconhecer o valor da chave mestra secreta e das chaves parciais secretas!
- No jogo entre adversário e simulador, o adversário deve fornecer uma chave secreta parcial sempre que
  - criar novo usuário
  - solicitar a revelação de uma chave de sessão

Caso da Autoridade Mal Intencionada

Poder do Adversário sobre Sessão *Fresh*





Definição de Segurança (Modelo Estendido)

(CL-AKA seguro)

Um protocolo CL-AKA é dito **seguro**
se qualquer adversário,
externo ou interno mal intencionado,
tem vantagem negligenciável
sob o parâmetro de segurança.

Roteiro

Protocolo CL-AKA Proposto

- Baseado em emparelhamento bilinear
- Mostrado seguro no modelo estendido
- Hipótese de dificuldade do problema computacional Gap-BDH
- Modelo de oráculo aleatório

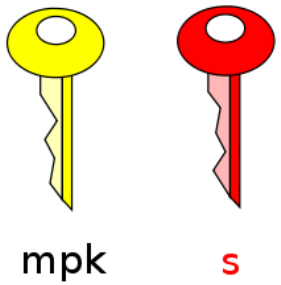
Fases do Protocolo Proposto

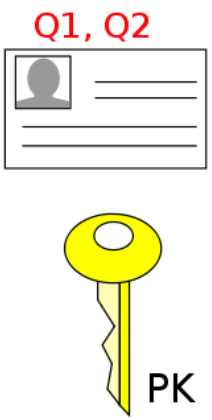
- Inicialização do Sistema
- Geração de Chaves de Usuário
- Acordo de Chave de Sessão

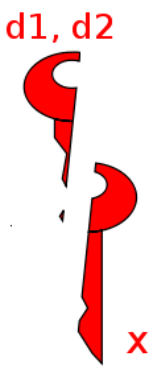
Inicialização do Sistema

- Escolha dos parâmetros do sistema e de um
- Emparelhamento bilinear
  $e : G \times G \rightarrow G_T$;

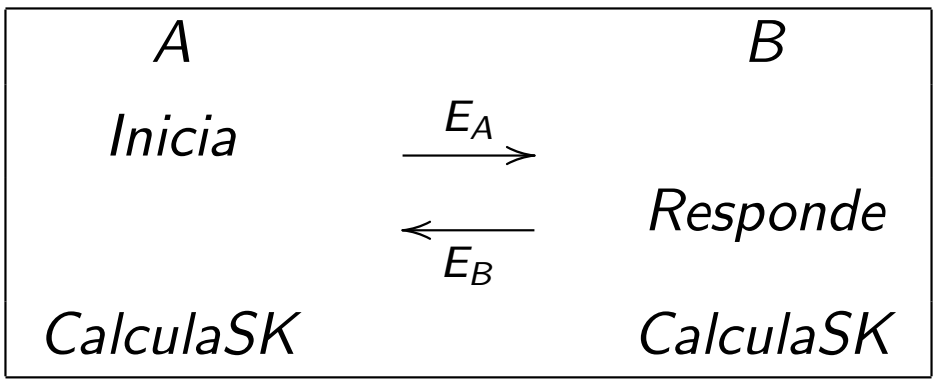
Chaves de Usuário e do Sistema - Variáveis







Acordo de Chave de Sessão



Acordo de Chave e Troca de Mensagens

A		B
Inicia:		*Responde:*
sorteia r_A		sorteia r_B
calcula $r_A P$		calcula $r_B P$
envia $E_A = (r_A P, x_A P)$		envia $E_B = (r_B P, x_B P)$
	$\xrightarrow{E_A}$	
	$\xleftarrow{E_B}$	
CalculaSK		*CalculaSK*

Cálculo da Chave – CalculaSK por A

$$K = e(r_B P + Q_{B_1}, r_A s P + d_{A_1})$$

$$L = e(r_B P + Q_{B_2}, r_A s P + d_{A_2})$$

$$M = e(x_B P, d_{A_1}) \cdot e(Q_{B_1}, x_A s P)$$

$$Z = (x_A x_B P, x_A r_B P, r_A r_B P, r_A x_B P)$$

$$SK = H(A, B, E_A, E_B, K, L, M, Z)$$

Avaliação da Proposta

Segurança:

- Modelo proposto é extensão
- Fortalece a segurança

Desempenho:

- Mantém custo computacional (tempo e espaço)
- Implementação para comparação com outros dois protocolos
  - Emparelhamento simétrico, curvas binárias
  - Relic

Protocolos Seguros sob o Problema Gap-BDH

Tabela: Cálculo normal, sem pré-computação

	LBG09-Gap	GOT10-Gap	Proposto (*)
Emparelhamentos	4	4	4
Exponenciações em G_T	2	0	0
Multiplicações em G_T	2	1	1
Multiplicações em G	5	7	7
Adições em G	0	2	4
Modelo de segurança	Lippold et al. (2009)		estendido
Tempo (s) B-271	0,062	0,061	0,062
B-1223	3,504	3,432	3,442

(*) Desempenho equivalente, porém com maior nível de segurança

Protocolos Seguros sob o Problema Gap-BDH

Tabela: Cálculo com pré-computação e armazenamento

	LBG09-Gap	GOT10-Gap	Proposto (*)
Emparelhamentos	1	1	2
Exponenciações em G_T	1	0	0
Multiplicações em G_T	1	0	0
Multiplicações em G	4	5	5
Adições em G	0	2	4
Modelo de segurança	Lippold et al. (2009)		estendido
Tempo (s) B-271	0,019	0,019	0,033
B-1223	0,992	0,955	1,769

(*) Com pré-computação, a proposta é menos eficiente em tempo quando comparada com os demais protocolos

Considerações Finais

Propusemos:

- uma extensão de modelo para CL-AKA, para segurança contra autoridade mal intencionada, que frauda na geração de parâmetros do sistema
- protocolo seguro nesse modelo estendido

Considerações Finais

Propusemos:

- uma extensão de modelo para CL-AKA, para segurança contra autoridade mal intencionada, que frauda na geração de parâmetros do sistema
- protocolo seguro nesse modelo estendido

Limitações:

- problema computacional Gap:
  - é possível evitá-lo? (no caso da autoridade mal intencionada)
- modelo com oráculos aleatórios

Perguntas?

Obrigada!