Introdução às Infraestruturas de Chaves Públicas

Programa de Verão 2007

Luiz M. R. Gadelha Jr.

Coordenação de Sistemas e Redes

Laboratório Nacional de Computação Científica

E-mail: `lgadelha@lncc.br`

Tópicos

- Aula 1: Conceitos preliminares
  - Motivação sobre segurança da informação
  - Criptografia simétrica
  - Criptografia assimétrica
  - Funções de hash
  - Assinaturas digitais
  - Motivação sobre ICPs

Tópicos

- Aula 2: Infraestruturas de chaves públicas
  - Certificados digitais
  - Autoridades certificadoras
  - Autoridades de registro
  - Caminhos de validação
  - Revogação de certificados
  - Representação de certificados
  - Padrões técnicos (PKCS)

Tópicos

- Aula 3: Exemplos de aplicações
  - Protocolo SSL/TLS
  - E-mail seguro S/MIME
  - Selo cronológico digital (*Timestamping*)
  - *Globus Security Infrastructure*: segurança em grades computacionais

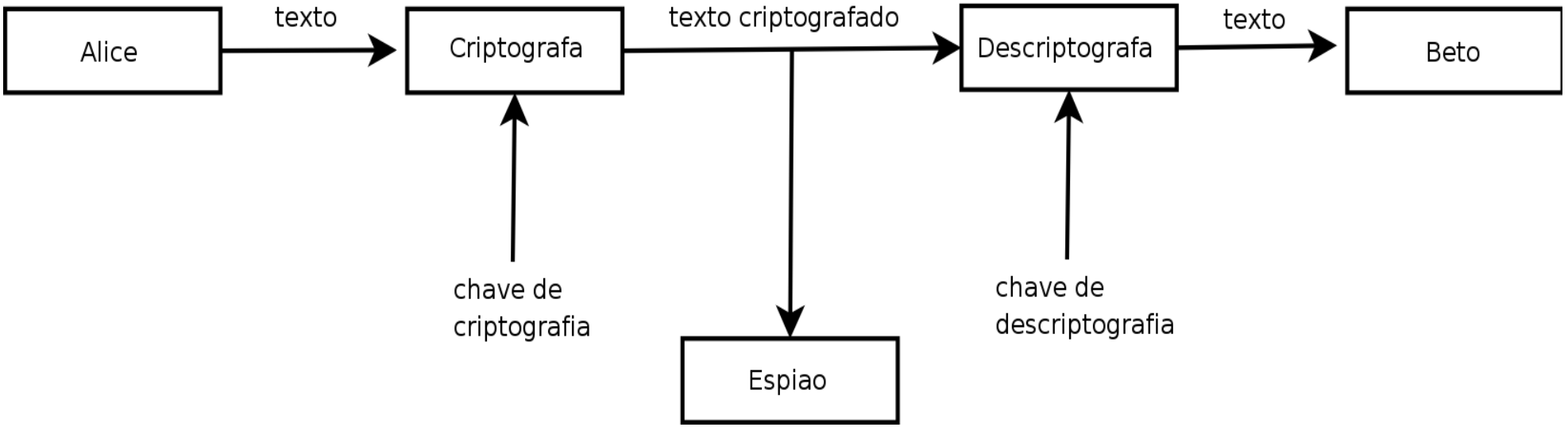
Tópicos

- Aula 4: Desenvolvimento com OpenSSL
  - Descrição da biblioteca de programação (API) do OpenSSL
  - Estudo de caso: aplicação cliente-servidor com protocolo SSL

Tópicos

- Aula 5: Implementação de uma AC
  - Requisitos operacionais de uma AC
  - Estudo de caso: Software RNP - ICPEDU
  - Instalação
  - Configuração
  - Operação

Motivação sobre Segurança da Informação



Motivação sobre Segurança da Informação

Espião pode desejar:

- Ler a mensagem
- Obter a chave de criptografia
- Alterar/adulterar a mensagem enviada por Alice
- Se passar por Alice para Beto

Motivação sobre Segurança da Informação

Ataques possíveis:

- Texto criptografado conhecido.
- Texto conhecido.
- Texto escolhido.
- Texto criptografado escolhido.

Motivação sobre Segurança da Informação

Propriedades desejáveis para a realização de transações eletrônicas de forma segura:

- Confidencialidade: o Espião não deve ser capaz de ler a a mensagem de Alice.
- Integridade: Beto deve poder verificar que a mensagem não foi alterada.
- Autenticação: Beto deve poder verificar que apenas Alice poderia ter enviado a mensagem.
- Não-repúdio: Alice não pode negar que ela enviou a mensagem.

Motivação sobre Segurança da Informação

Criptografia permite realizar transações com as propriedades mencionadas.

Tipos de criptografia:

- Criptografia simétrica

  Apenas uma chave para criptografar e descriptografar.

- Criptografia assimétrica

  Par de chaves: chave pública e chave privada.

Criptografia Simétrica

Exemplo 1. Cifra de César

a = 0, b = 1, c = 2, ... , w = 22, x = 23, y = 24, z = 25

$$x \rightarrow x + k \, (mod \, 26)$$

Escolhendo k = 3 como chave:

“atacar” → “dxdfdu”

Criptografia Simétrica

Exemplo 1. Cifra de César

Número de chaves possíveis: 26

Trivial de ser quebrado por *força bruta* com os recursos computacionais atuais.

Pode ser utilizada também *análise de freqüência*.

Criptografia Simétrica

Exemplo 2. Cifra de Vigenère

Chave da forma: $(a_1, a_2, \ldots, a_n)$

Texto:	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
	+	+	+	+	+	+	+	+	+	+
	a_1	a_2	a_3	a_4	a_1	a_2	a_3	a_4	a_1	a_2
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Texto cript.:	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}

Criptografia Simétrica

Exemplo 2. Cifra de Vigenère

Chave: (c, h, a, v, e)

Texto:	m	e	u	t	e	x	t	o
	+	+	+	+	+	+	+	+
	c	h	a	v	e	c	h	a
	↓	↓	↓	↓	↓	↓	↓	↓
Texto cript.:	o	l	u	o	i	z	a	o

Criptografia Simétrica

Exemplo 2. Cifra de Vigenère

Ataques de texto conhecido, texto escolhido e texto criptografado escolhido são triviais.

Ataque de texto criptografado conhecido:

- Descobrir o tamanho da chave.
- Análise de freqüência.

Criptografia Simétrica

Exemplo 3. *One-time pad*

Um texto pode ser representado como uma seqüencia binária.

“A” é 65 no código ASCII, ou 01000001 em binário.

“ABC” = 010000010100001001000011

Criptografia Simétrica

Exemplo 3. *One-time pad*

One-time pad utiliza a operação “ou exclusivo” (XOR $\oplus$):

$$0 \oplus 0 = 0$$

$$0 \oplus 1 = 1$$

$$1 \oplus 0 = 1$$

$$1 \oplus 1 = 0$$

Criptografia Simétrica

Exemplo 3. *One-time pad*

A chave é gerada randomicamente e tem o mesmo tamanho do texto.

Texto	010000010100001001000011
	$\oplus$
Chave	101001110101010001010011
Texto cript.	111001100001010000010000

Criptografia Simétrica

Exemplo 3. *One-time pad*

Descriptografia:

Texto cript.	111001100001010000010000
	$\oplus$
Chave	101001110101010001010011

Texto	010000010100001001000011

Teoricamente inquebrável.

Inviável na prática: geração e transmissão da chave.

Criptografia Simétrica

Exemplo 4. *Data Encryption Standard (DES)*

Selecionado pelo NIST (EUA) em 1975 como padrão americano de criptografia simétrica.

Utiliza permutações e operações como XOR em uma seqüencia de iterações.

Baseado no LUCIFER da IBM.

É uma *cifra de bloco*, texto é quebrado em blocos de 64 bits que são então criptografados com uma chave de 56 bits.

Criptografia Simétrica

Exemplo 4. *Data Encryption Standard (DES)*

Versão simplificada:

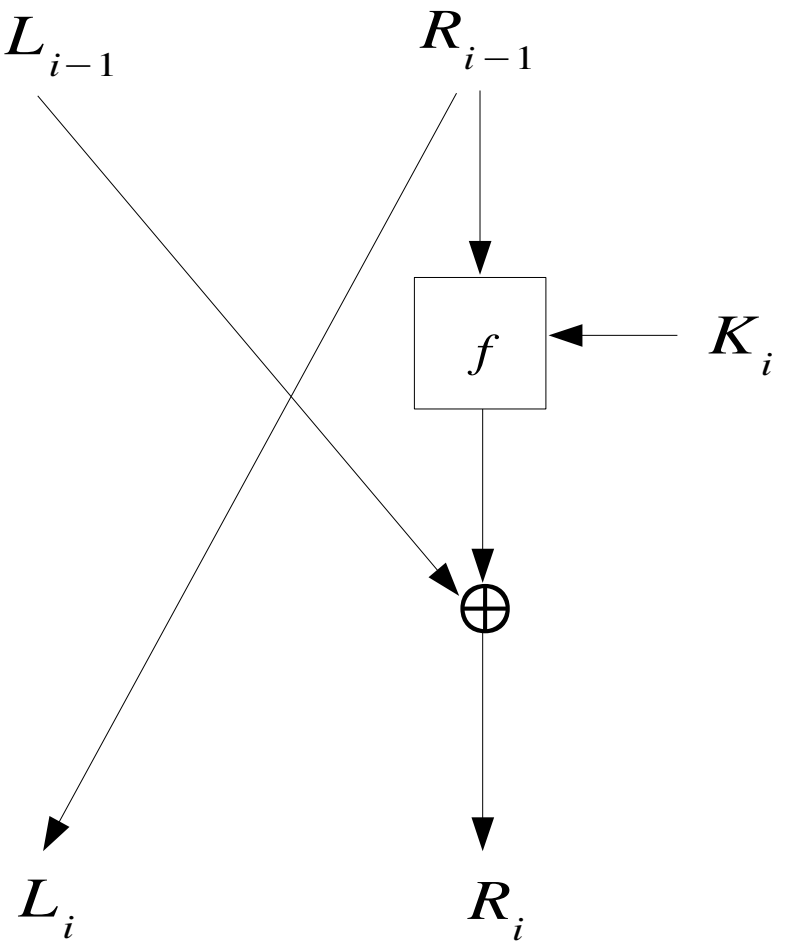
Mensagem de 12 bits escrita da forma: $M = L_0 R_0$ (6 + 6 bits)

Chave k de 9 bits.

k_i = 8 bits de k a partir da posição i.

$L_i = R_{i-1}$

$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$



Criptografia Simétrica

Exemplo 4. *Data Encryption Standard (DES)*

Descriptografia:

$$L_n R_n \rightarrow R_n L_n \rightarrow [L_n][R_n \circ f(L_n, K_n)]$$

como $L_n = R_{n-1}, R_n = L_{n-1} \circ f(R_{n-1}, K_n)$

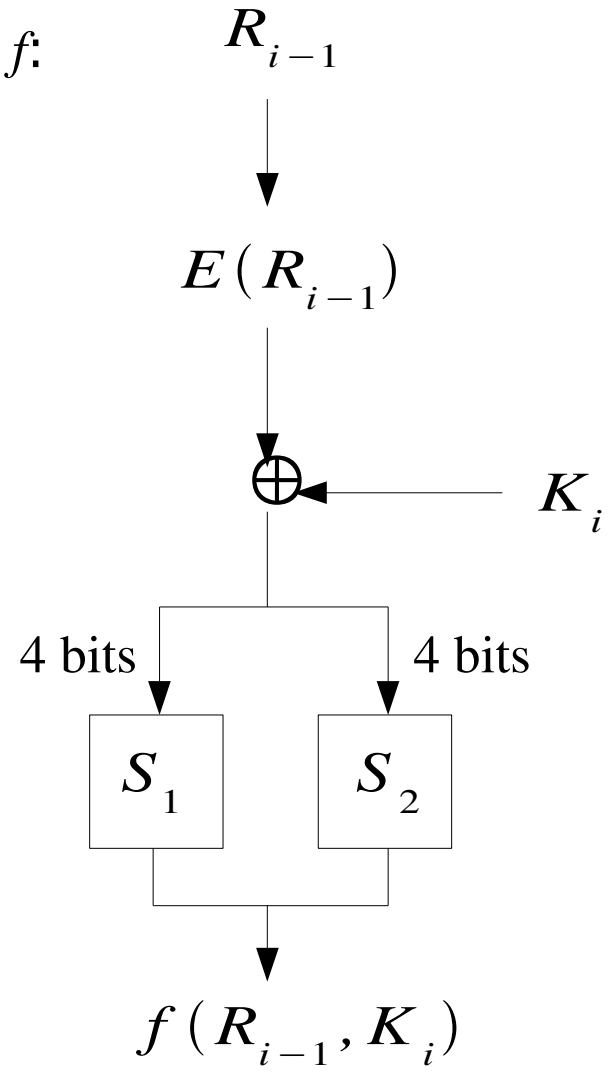
$$[L_n][R_n \circ f(L_n, K_n)] \rightarrow [R_{n-1}][L_{n-1} \circ f(R_{n-1}, K_n) \circ f(L_n, K_n)]$$

$$\rightarrow [R_{n-1}][L_{n-1}]$$

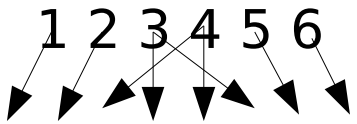
Criptografia Simétrica

Exemplo 4. *Data Encryption Standard (DES)*

A função f:



E:



1 2 4 3 4 3 5 6

S_1:

101 010 001 110 011 100 111 000

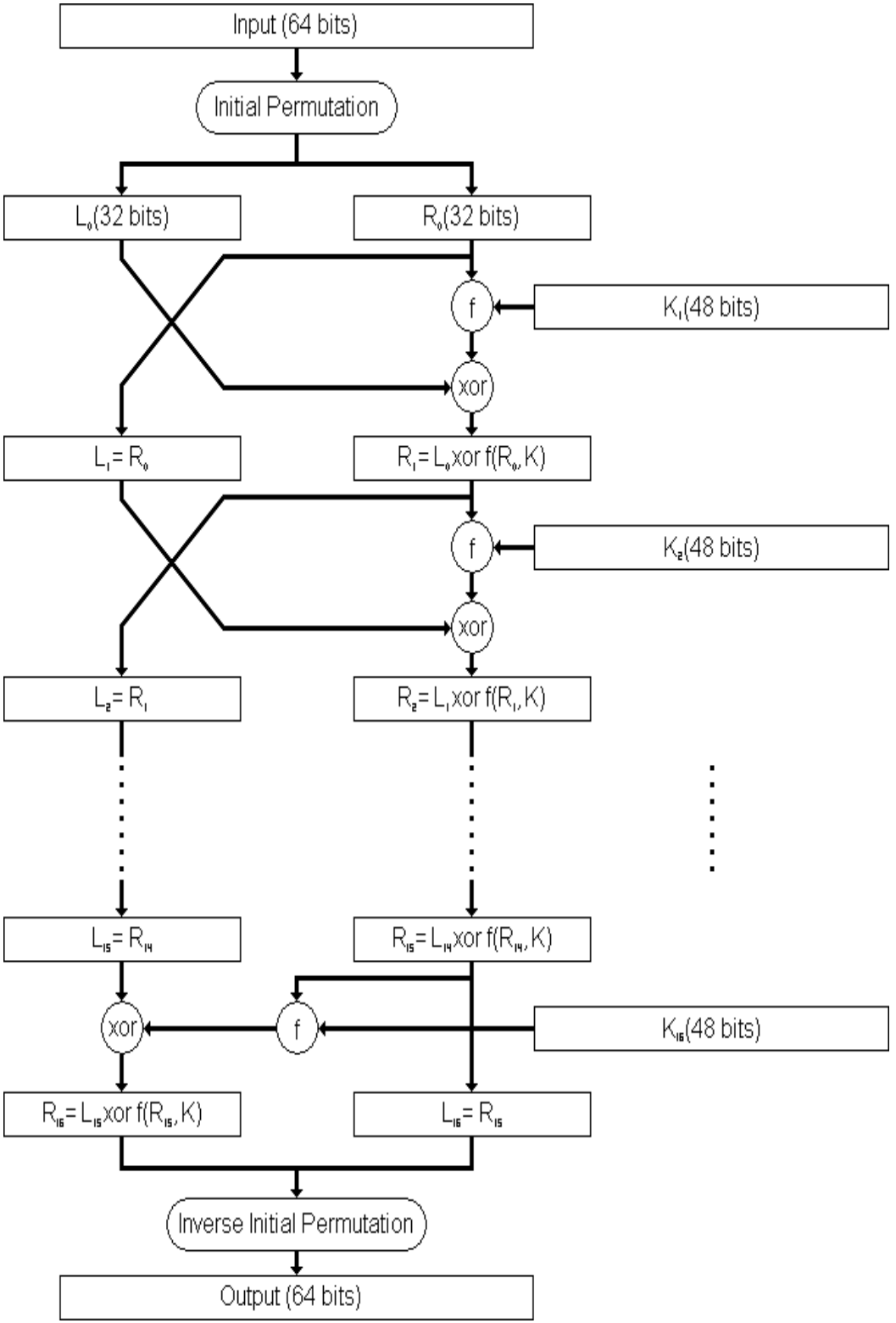
001 100 110 010 000 111 101 011

S_2:

100 000 110 101 111 001 011 010

Criptografia Simétrica

Exemplo 4. *DES*



Criptografia Simétrica

Na década de 90 o NIST realizou uma nova seleção pública para escolha de um novo algoritmo de criptografia simétrica como padrão.

Vencedor: Rijndael [Rijmen, Daemen].

Definido pelo NIST como *Advanced Encryption Standard* (AES).

Problema fundamental com criptografia simétrica:

Necessidade de transmissão da chave.

Criptografia Assimétrica

Cada participante possui duas chaves:

Chave pública (P): é divulgada para outros participantes.

Chave privada (S): somente o proprietário a conhece.

São mutuamente inversas:

$$E(E(M, S),P) = E(E(M, P), S)$$

Criptografia Assimétrica

Exemplo 4. RSA [Rivest, Shamir, Adleman]

- Gere dois números primos grandes p e q.
- Calcule $n = pq$ e $\text{phi}(n) = (p\text{-}1)(q\text{-}1)$.
- Escolha $e < \text{phi}(n)$ tal que $\text{mdc}(e, \text{phi}(n)) = 1$.
- Calcule d tal que $de \equiv 1 \pmod{\text{phi}(n)}$

(e, n) é a chave pública.

(d, n) é a chave privada.

Criptografia Assimétrica

Criptografia:

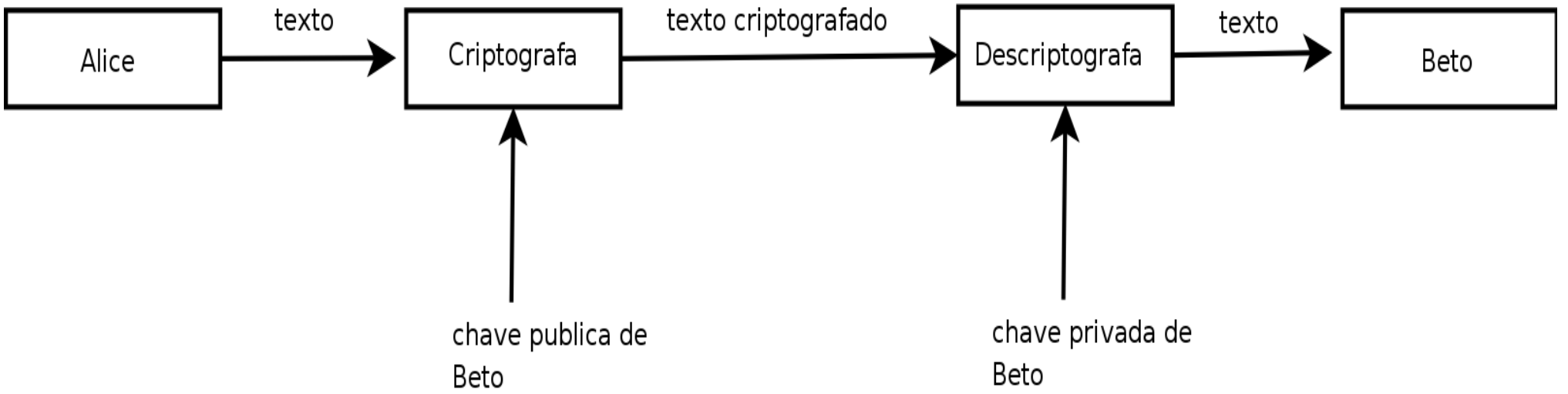
M elemento de $\mathbf{Z}_n$

$C \equiv M^e \pmod{n}$

Descriptografia:

$C^d \equiv (M^e)^d \equiv M^{de} \equiv M \pmod{n}$

Criptografia Assimétrica



Criptografia Assimétrica

Criptogradia assimétrica é extremamente mais lenta que a criptografia simétrica (1000 vezes em alguns casos).

Alice e Beto podem utilizar criptografia assimétrica para trocar chaves simétricas.

1. Alice gera uma chave simétrica K e a criptografa com a chave pública de Beto e a envia para ele.
2. Beto descriptografa K com a sua chave privada.
3. Alice é Beto passam a trocar mensagens criptografadas com a chave simétrica K (mais rápido).

Funções de hash

Entrada: um texto

Saída: uma seqüencia de comprimento fixo (p.ex. 128 bits)
chamada de *hash*.

São utilizadas para verificar a integridade na transmissão de dados.

Exemplos de algoritmos: MD5, SHA1

Funções de hash

Mudança em um bit da mensagem altera consideravemente o hash.

MD5(“teste”) = `1ca308df6cdb0a8bf40d59be2a17eac1`

MD5(“peste”) = `c469d874c4226f2f6bb0696d3801b4b1`

Assinaturas digitais

Alice pode assinar digitalmente M como segue:

1. Alice calcula $M_{S_{Alice}}$.
2. A assinatura digital de M por Alice é dada pelo par $\langle M, M_{S_{Alice}} \rangle$.
3. Alice envia $\langle M, M_{S_{Alice}} \rangle$ para Beto.
4. Beto calcula $(M_{S_{Alice}})_{P_{Alice}}$.
5. Se $(M_{S_{Alice}})_{P_{Alice}} = M$ então a assinatura é válida.

Assinaturas digitais

Alice pode assinar digitalmente M (com hash) como segue:

1. Alice calcula M_{hash}.
2. Alice calcula $(M_{hash})_{S_{Alice}}$.
3. A assinatura digital de M por Alice é dada pelo par $\langle M, (M_{hash})_{S_{Alice}} \rangle$.

Assinaturas digitais

4. Alice envia $\langle M, (M_{hash})_{S_{Alice}} \rangle$ para Beto.

5. Beto calcula M_{hash}.

6. Se $(M_{hash})_{P_{Alice}} = M_{hash}$ então a assinatura é válida.

Assinaturas digitais

Alice pode assinar digitalmente e criptografar M (com hash) como segue:

1. Alice calcula M_{hash}.
2. Alice calcula $(M_{hash})_{S_{Alice}}$.
3. Alice calcula $(\langle M, (M_{hash})_{S_{Alice}} \rangle)_{P_{Beto}}$.
4. Alice envia $(\langle M, (M_{hash})_{S_{Alice}} \rangle)_{P_{Beto}}$ para Beto.

Assinaturas digitais

5. Beto calcula $(((\langle M, (M_{hash})_{S_{Alice}}\rangle)_{P_{Beto}})_{S_{Beto}} = \langle M, (M_{hash})_{S_{Alice}}\rangle$

6. Beto calcula M_{hash}.

7. Se $((M_{hash})_{S_{Alice}}))_{P_{Alice}} = M_{hash}$ então a assinatura é válida.

Motivação sobre ICPs

No esquema de criptografia assimétrica existe o problema de como determinar a identidade do proprietário de uma chave pública.

Um usuário *A* pode tentar se passar por um usuário *B* para um usuário *C*.

Se *C* “acredita” em *A*, *C* perde a confidencialidade na transmissão de dados.

Motivação sobre ICPs

Solução: utilização de um terceiro de confiança chamado de *autoridade certificadora*.

Suponha que A e B confiem em uma entidade C.

C pode verificar a identidade de A e B e emitir documentos atestando a posse de suas respectivas chaves públicas.

Certificados Digitais

Um *certificado digital* é um documento que estabelece uma relação de propriedade entre uma entidade e uma chave pública.

Os principais campos de um certificado digital são:

- Nome da entidade proprietária da chave pública (*Subject*).
- Chave pública do proprietário (*Subject Public Key Info*).
- Nome da autoridade certificadora (*Issuer*).
- Assinatura digital da autoridade certificadora (*Signature Algorithm*).

```
Subject: CN=Luiz M. R. Gadelha Jr.,OU=CSR,O=LNCC,C=BR

Validity
    Not Before: Aug  5 13:40:29 2005
    Not After:  Aug  5 13:40:29 2006

Subject Public Key Info:
    Public key Algorithm: rsaEncryption
    RSA Public Key (1024 bit):
        00:e7:02:a9:1a:27:8a:36:d1:7a:b0:d5:cc:e5:fd:
                        ...
        13:34:04:84:00:6d:53:f3:6b
    Exponent: 65537 (0x10001)

Issuer: CN=Autoridade Certificadora,O=LNCC,C=BR

Serial Number: 12 (0xc)

X509v3 CRL Distribution Points:
    URI:https://ac.lncc.br/AC_LNCC/pub/crl/cacrl.crl

Signature Algorithm: sha1WithRSAEncryption
    b2:33:b2:b6:9f:68:fb:86:b7:19:36:c3:05:19:41:71:d1:b0:
                        ...
    6e:f0:67:a6
```

Autoridades Certificadoras

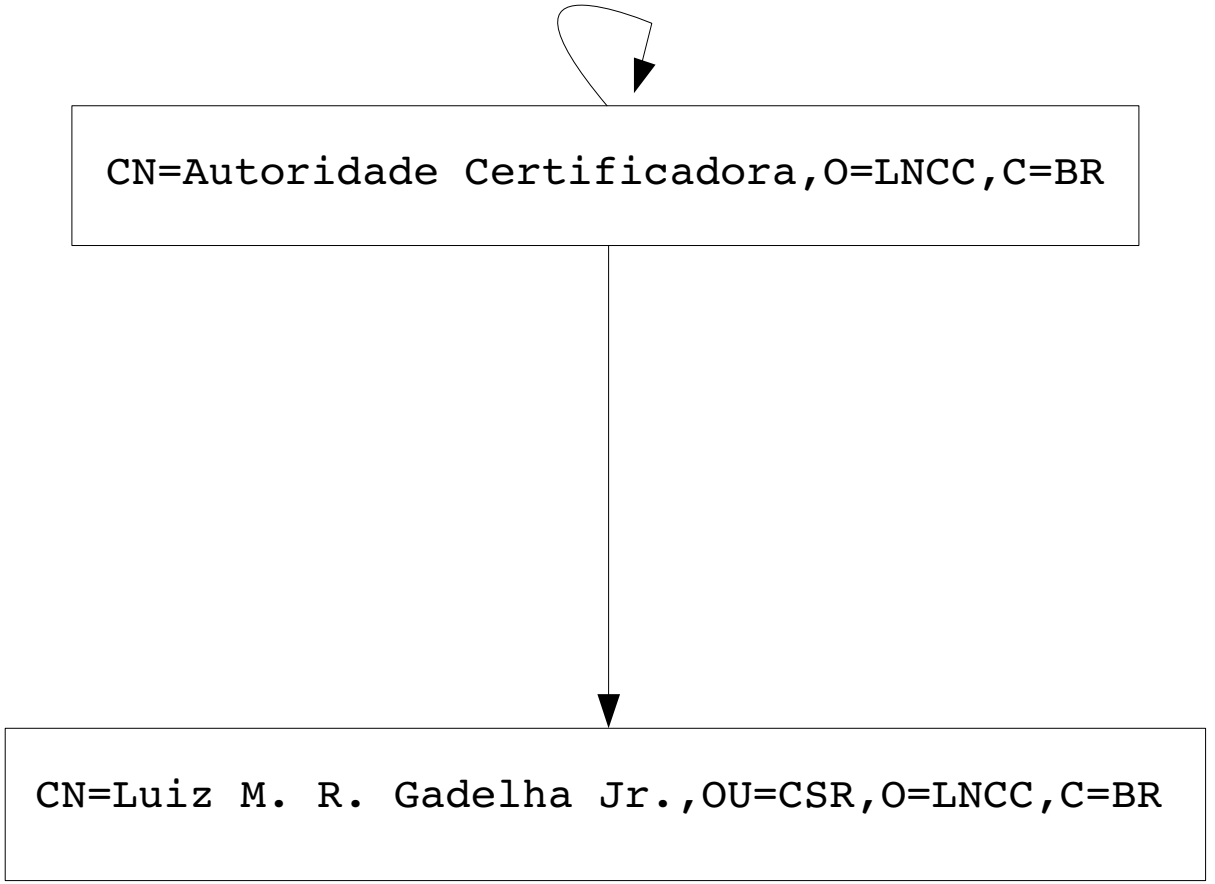
A autoridade certificadora (AC) age como *terceiro de confiança*.

A própria autoridade certificadora possui o seu certificado digital.

Esse certificado é auto-assinado ou é assinado por outra autoridade certificadora.

```
Subject: CN=Autoridade Certificadora,O=LNCC,C=BR

Validity
    Not Before: Apr 27 12:53:36 2005 GMT
    Not After:  Apr 22 12:53:36 2025 GMT

Subject Public Key Info:
    Public key Algorithm: rsaEncryption
    RSA Public Key (2048 bit):
        00:b8:13:f4:a8:1e:33:61:60:63:1d:fa:aa:4c:d0:
                        ...
        de:af
    Exponent: 65537 (0x10001)

Issuer: CN=Autoridade Certificadora,O=LNCC,C=BR

Serial Number: bf:b8:fa:de:b3:59:78:f4

X509v3 CRL Distribution Points:
    URI:https://ac.lncc.br/AC_LNCC/pub/crl/cacrl.crl

Signature Algorithm: sha1WithRSAEncryption
    98:e9:ce:fc:6c:04:99:7c:b4:79:3a:4b:79:c1:c9:ab:bc:ba:
                        ...
    20:3d:b9:55
```

Autoridades Certificadoras



Certificados Digitais

Outros campos de um certificado digital X.509v3:

- *Key Usage*: finalidade do certificado (assinatura, criptografia, etc.)
- *CRL Distribution Points*: endereço (URL) da lista de certificados revogados.
- *Subject Alternative Name*: nome alternativo do proprietário do certificados.

Autoridades de Registro

Uma *autoridade de registro* (AR) processa e faz a validação das solicitações de certificados.

Gerencia a interação entre a entidade final e a AC.

Diversos níveis de validação:

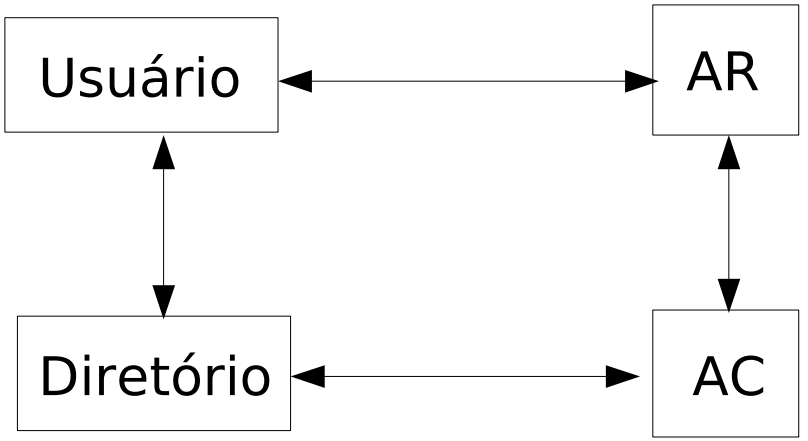
- Existência de caixa postal de e-mail.
- Validação presencial mediante apresentação de documentos.

Diretório

Um sistema de certificação digital deve manter um diretório (em LDAP ou HTTP p.ex.) para publicação de:

- Certificados emitidos.
- Listas de certificados revogados.
- Política e práticas de certificação.

Sistema de Certificação Digital



Infraestrutura de Chaves Públicas

Ao conjunto de software, hardware, procedimentos, autoridades certificadoras, autoridades de registro e diretórios que gerenciam a utilização de chaves de criptografia de chave pública chamamos de **infraestrutura de chaves públicas** (ICP).

Public-key Infrastructure (PKI)

Emissão de Certificado Digital

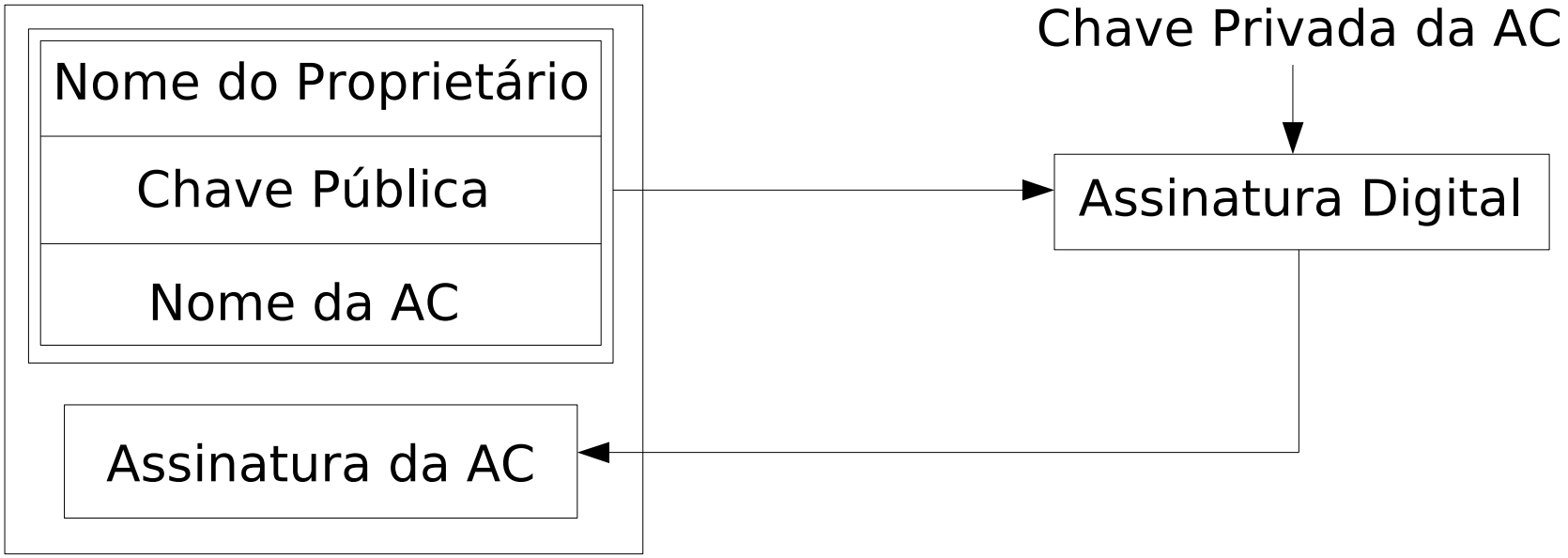
Emissão de um certificado digital para uma entidade final *E*:

1. *E* gera seu par de chaves (pública e privada)
2. *E* cria um documento (*Certificate Signing Request* – CSR) contendo a sua chave pública e os seus dados (nome, organização, etc.)
3. *E* envia a CSR para a sua respectiva AR.
4. A AR realiza o seu processo de validação de *E*.
5. Se a validação for bem-sucedida a AR assina digitalmente uma solicitação para emissão de certificado que é enviada para a AC.

Emissão de Certificado Digital

Emissão de um certificado digital para uma entidade final *E*:

6. A AC verifica a assinatura contida na solicitação feita pela AR.
7. A AC gera um certificado para *E* e o assina digitalmente.
8. A AC publica o certificado de *E* no seu diretório.



Revogação de Certificados

Eventualmente um certificado digital precisará ser revogado:

- Perda da privativa.
- Roubo da chave privativa.
- Desligamento de usuário.

Revogação de Certificados

A revogação pode ser feita diretamente pela AC ou por solicitação da entidade final.

A AC publica em seu diretório um documento chamado de *lista de revogação de certificados* (*certificate revocation list* - CRL).

O documento contém uma lista de números de série de certificados revogados e é assinado digitalmente pela AC.

Periodicamente a CRL é atualizada.

```
Certificate Revocation List (CRL):
        Version 2 (0x1)
        Signature Algorithm: md5WithRSAEncryption
        Issuer: /C=BR/O=SINAPAD/OU=GRADGIGA/CN=Autoridade Certificadora
        Last Update: Jan 20 15:56:04 2006 GMT
        Next Update: Jul 19 15:56:04 2006 GMT
        CRL extensions:
            X509v3 CRL Number:
                24
Revoked Certificates:
    Serial Number: 02
        Revocation Date: May  3 19:13:48 2005 GMT
    Serial Number: 03
        Revocation Date: May  3 19:13:34 2005 GMT
    Serial Number: 04
        Revocation Date: Dec 21 13:06:26 2005 GMT
    Serial Number: 05
        Revocation Date: Dec 21 13:06:00 2005 GMT

          ....


    Signature Algorithm: md5WithRSAEncryption
        d0:a7:a0:8b:55:f6:da:bb:53:9d:af:7a:b2:5d:9f:a2:01:bc:
        46:23:e1:7c:d2:fc:5f:e1:d0:ec:db:c9:c4:6a:ed:6d:e5:19:
        61:a7:2e:29:7a:b9:0f:df:90:fb:dd:bf:9b:f3:2e:e9:17:8c:
        4f:e9:f5:c0:92:69:c7:c0:22:4c:4c:8c:b8:44:af:72:fe:57:
        d9:c8:e4:ca:95:b2:71:c1:68:61:40:41:c8:04:2c:4d:db:f9:
        16:17:3d:62:bd:c9:bf:b3:81:0b:ac:1e:f5:3b:50:80:23:7c:
        3d:93:63:8c:ec:dd:14:69:bd:86:c0:8d:7c:fb:28:29:55:da:
        6d:af:84:02:31:39:a1:d2:08:7c:93:60:5a:67:43:43:10:ab:
        9e:69:b8:bf:99:60:75:9d:c9:8b:63:74:0e:b4:23:53:83:53:
        ce:af:f3:6c:d5:00:17:24:77:1e:d6:5f:3f:e6:b4:c1:58:83:
        b9:f1:1c:cc:64:1a:7d:0c:10:5c:d4:71:7f:18:b9:ce:73:30:
        bc:47:ab:de:04:2e:89:2a:70:79:6d:de:b5:df:fc:a9:82:b4:
        4b:d5:89:95:e3:0f:d3:e5:90:78:a9:d4:7f:5c:f4:ed:3c:ac:
        7e:8e:98:c1:91:2a:51:25:a5:12:66:e7:f1:56:e3:5b:05:f6:
        48:a6:57:00
```

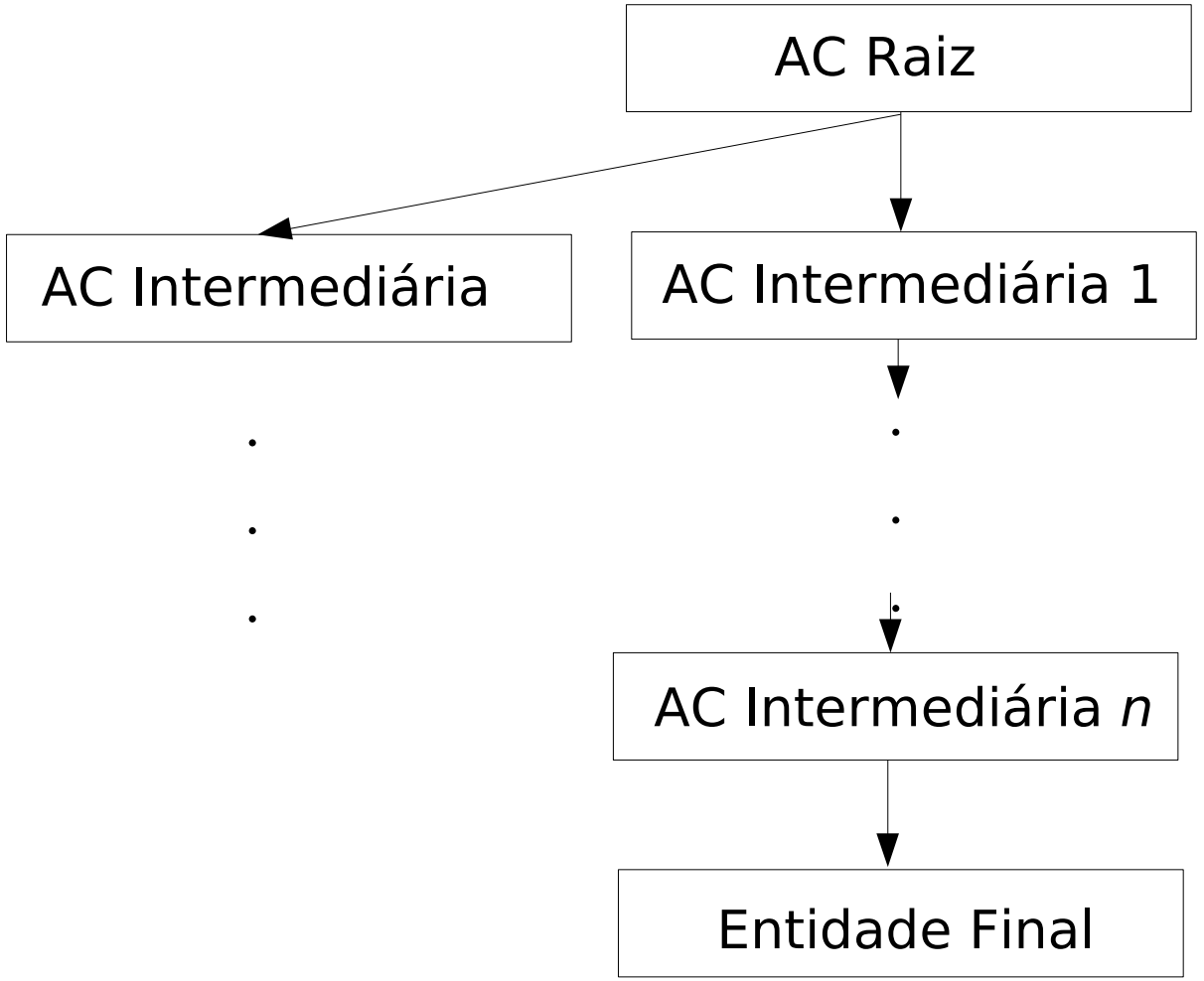
Revogação de Certificados

O protocolo OCSP (*Online Certificate Status* Protocol) permite a verificação *online* da validade de um certificado:

1. O usuário se conecta ao servidor OCSP e consulta pelo número serial do certificado o seu status.

2. O OCSP verifica na base de dados da AC o status do certificado e devolve ao usuário uma resposta assinada pela AC.

Hierarquias de Certificação



Exemplos de ICPs

Maior AC mundial, certificados para servidores web, *code signing*, e-mail.

`http://www.verisign.com`

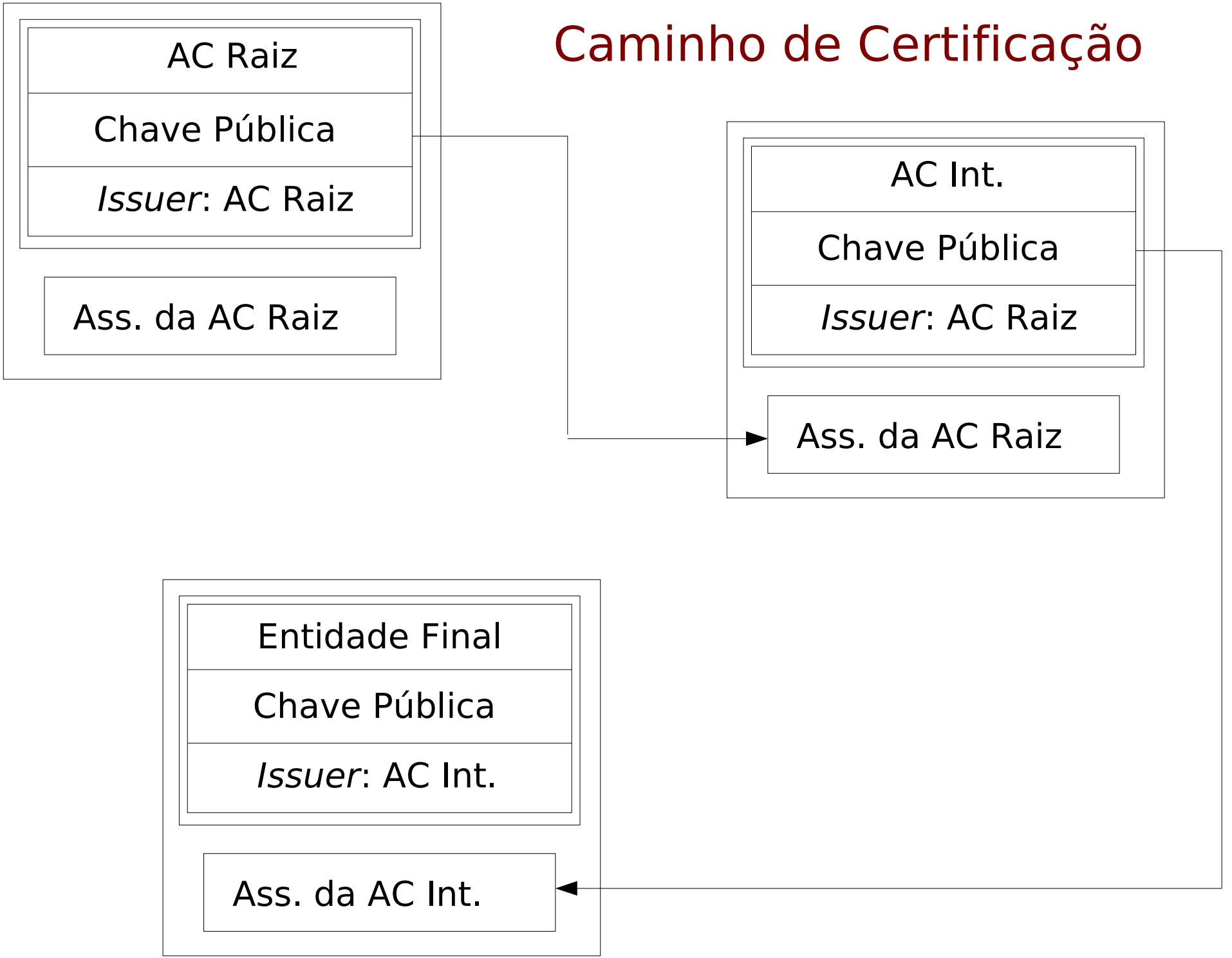
ICP do governo federal. e-CPF e e-CNPJ.

`http://www.icpbrasil.gov.br`

ICP para grade computacional (GRADGIGA) do SINAPAD usada para autenticação de usuários e hosts.

`http://ac.lncc.br`

Caminho de Certificação



Proteção da Chave Privada

As propriedades de segurança obtidas dependem da proteção adequada da chave privada de cada entidade.

Métodos de proteção:

- Armazenamento em arquivo criptografado em disco rígido ou alguma mídia similar.
- Armazenamento em hardware criptográfico protegido.
- Armazenamento em um servidor de credenciais.

Renovação de Certificado

Ao final do seu prazo de validade o certificado digital pode ser renovado, i.e. emissão de um novo certificado com o mesmo *Subject*.

Eventualmente o certificado antigo deverá ser revogado para que seja feita a renovação.

Pode ser mantido o par de chaves na emissão, no entanto é recomendado que um novo par de chaves seja gerado periodicamente.

Certificados para Finalidades Distintas

Assinatura digital e criptografia têm requisitos distintos para gerência de chaves criptográficas.

Assinatura digital:

- para suporte ao não-repúdio é importante que uma chave privada nunca deixe a mídia na qual foi gerada. I.e., não é recomendada a realização de *backups*.

- em caso de perda da chave um novo par pode ser gerado facilmente em substituição ao antigo.

- ao final do seu prazo de validade é recomendada a destruição da chave.

Certificados para Finalidades Distintas

Criptografia:

- é interessante armazenar a chave privada em *backup* mesmo após o término do seu prazo de validade para descriptografia de mensagens antigas.

Como os requisitos são conflitantes é comum que usuários tenham certificados distintos para cada finalidade.

Política de Certificação

Toda AC deve publicar um documento, a *política de certificação*, contendo:

- Garantias oferecidas pela AC, AR e outros.
- Política de identificação e autorização.
- Requisitos operacionais.
- Controles de segurança estabelecidos.
- Requisitos sobre o conteúdo dos certificados e CRLs.

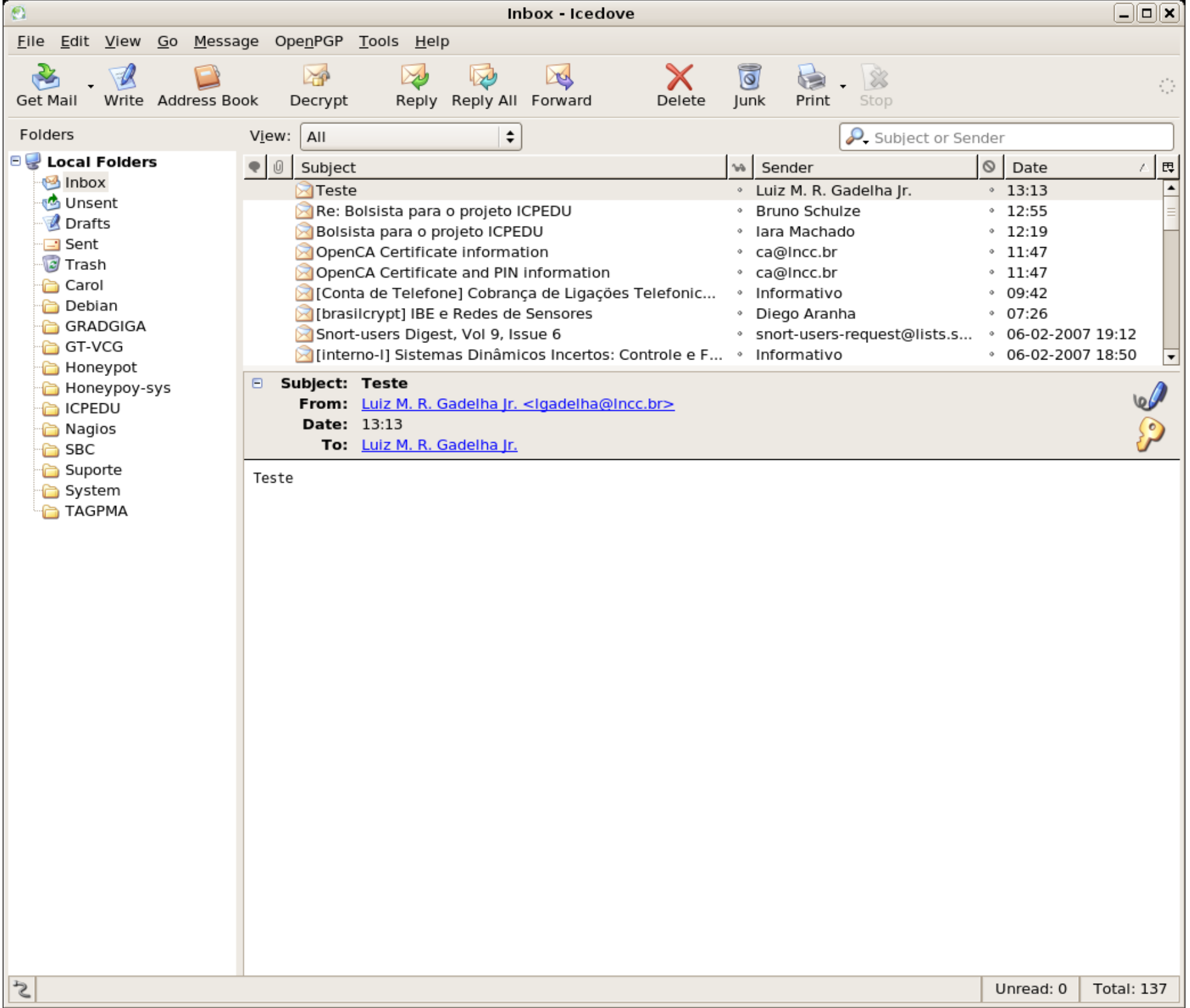
Aplicações: S/MIME (e-mail seguro)

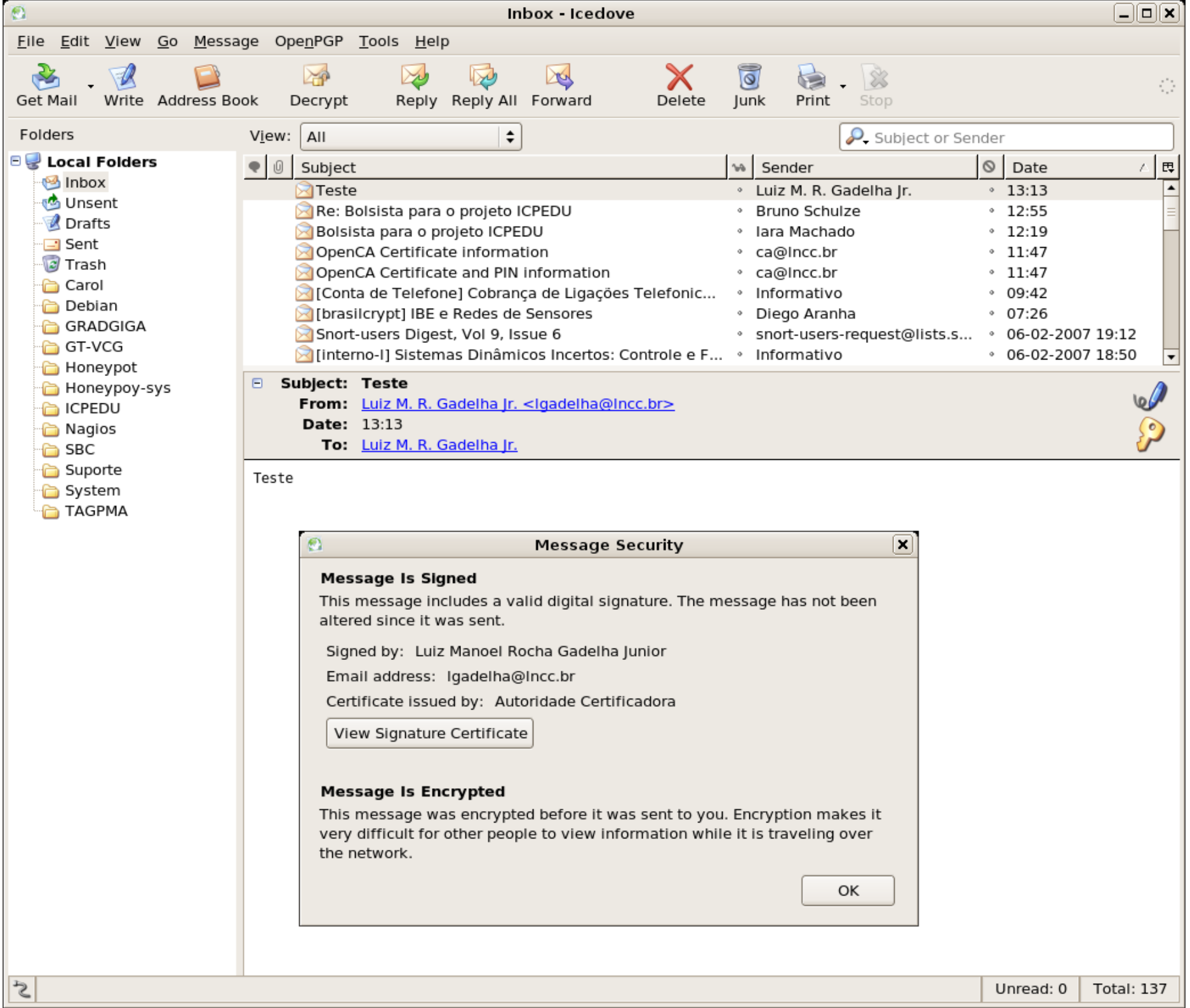
Provê confidencialidade, integridade e autenticação para mensagens de e-mail.

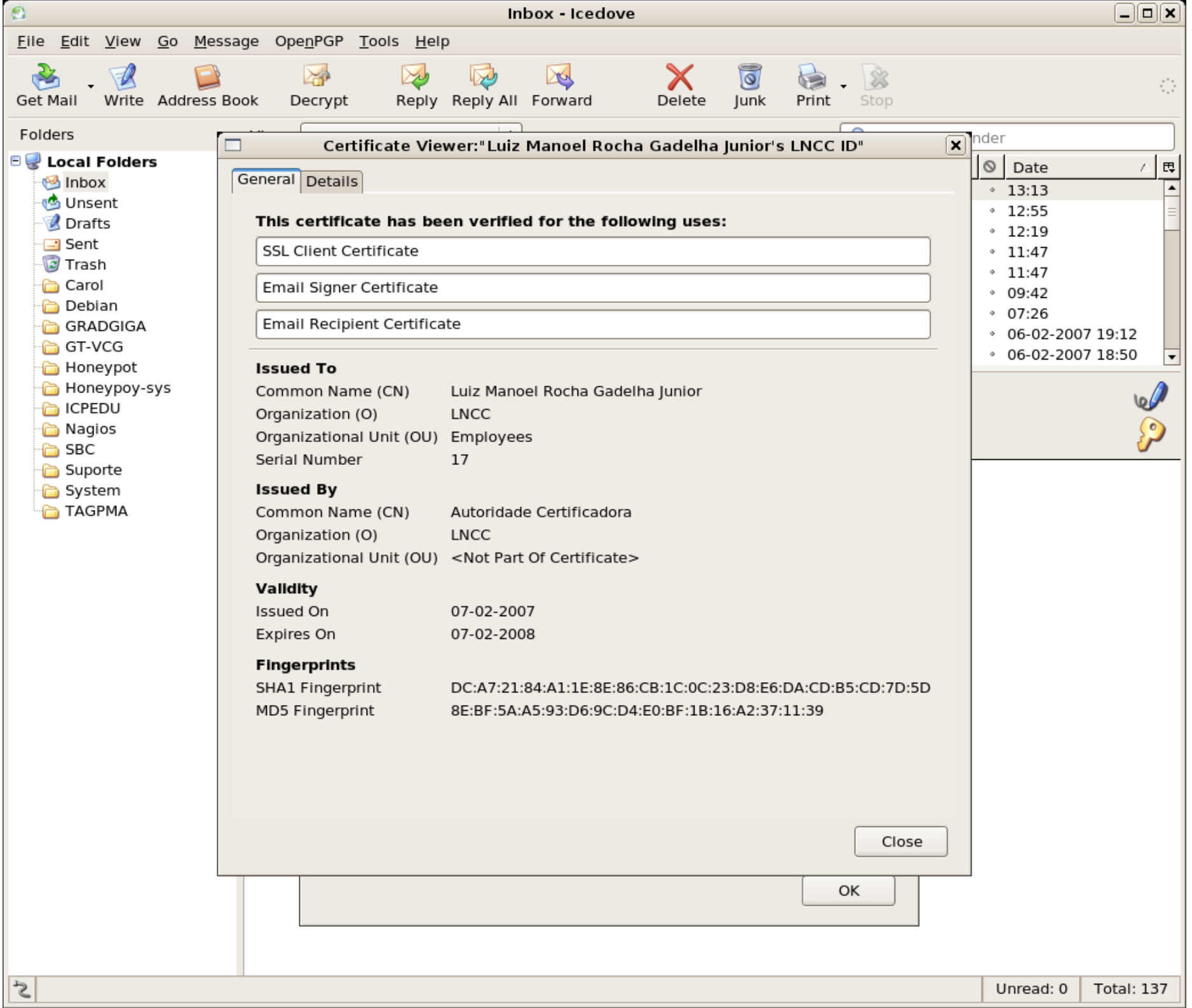
Para criptografia é utilizada uma chave simétrica gerada pelo remetente que é criptografada com a chave pública do destinatário.

Utiliza extensões de e-mail do tipo MIME para enviar objetos no formato PKCS#7.

```
Delivered-To: lgadelha@lncc.br
Received: from [146.134.201.102] (sino.lncc.br [146.134.201.102])
        by iota.lncc.br (Postfix) with ESMTP id 41029F2B7
        for <lgadelha@lncc.br>; Wed,  7 Feb 2007 12:06:30 -0200 (BRST)
Message-ID: <45C9DC79.4090409@lncc.br>
Date: Wed, 07 Feb 2007 12:04:41 -0200
From: "Luiz M. R. Gadelha Jr." <lgadelha@lncc.br>
User-Agent: Icedove 1.5.0.9 (X11/20061220)
MIME-Version: 1.0
To: "Luiz M. R. Gadelha Jr." <lgadelha@lncc.br>
Subject: Teste
X-Enigmail-Version: 0.94.1.0
Content-Type: multipart/signed; protocol="application/x-pkcs7-signature"; micalg=sha1;
boundary="------------ms040004050500010207040008"
X-UIDL: Z7B"!e'?!!<aK!![W7!!

This is a cryptographically signed message in MIME format.

--------------ms040004050500010207040008
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: 7bit

Teste

--------------ms040004050500010207040008
Content-Type: application/x-pkcs7-signature; name="smime.p7s"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7s"
Content-Description: S/MIME Cryptographic Signature

MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQEHAQAAoIIKHjCC
BQswggPzoAMCAQICARcwDQYJKoZIhvcNAQEFBQAwPzELMAkGA1UEBhMCQlIxDTALBgNVBAoT

                ...

0KvgBIJALnctqqamocnFLhNkFV/C+cnyBzbeI7wQlQMAz9cHc6k0QvHmAWTp0sU9w3Zk1DFM
TWyWL5jCzoTiMhIYoxsAAAAAAAA=
--------------ms040004050500010207040008--
```


Inbox - Icedove
File Edit View Go Message OpenPGP Tools Help
Get Mail
Write
Address Book
Decrypt
Reply
Reply All
Forward
Delete
Junk
Print
Stop
Folders
View: All
Subject or Sender
Local Folders
Inbox
Unsent
Drafts
Sent
Trash
Carol
Debian
GRADGIGA
GT-VCG
Honeypot
Honeypoy-sys
ICPEDU
Nagios
SBC
Suporte
System
TAGPMA
Subject
Sender
Date
Teste
Luiz M. R. Gadelha Jr.
13:13
Re: Bolsista para o projeto ICPEDU
Bruno Schulze
12:55
Bolsista para o projeto ICPEDU
Iara Machado
12:19
OpenCA Certificate information
ca@lncc.br
11:47
OpenCA Certificate and PIN information
ca@lncc.br
11:47
[Conta de Telefone] Cobrança de Ligações Telefonic...
Informativo
09:42
[brasilcrypt] IBE e Redes de Sensores
Diego Aranha
07:26
Snort-users Digest, Vol 9, Issue 6
snort-users-request@lists.s...
06-02-2007 19:12
[interno-l] Sistemas Dinâmicos Incertos: Controle e F...
Informativo
06-02-2007 18:50
Subject: Teste
From: Luiz M. R. Gadelha Jr. <lgadelha@lncc.br>
Date: 13:13
To: Luiz M. R. Gadelha Jr.
Teste
Unread: 0
Total: 137


Inbox - Icedove
File Edit View Go Message OpenPGP Tools Help
Get Mail Write Address Book Decrypt Reply Reply All Forward Delete Junk Print Stop
Folders
View: All
Subject or Sender
Local Folders
Inbox
Unsent
Drafts
Sent
Trash
Carol
Debian
GRADGIGA
GT-VCG
Honeypot
Honeypoy-sys
ICPEDU
Nagios
SBC
Suporte
System
TAGPMA
Subject Sender Date
Teste Luiz M. R. Gadelha Jr. 13:13
Re: Bolsista para o projeto ICPEDU Bruno Schulze 12:55
Bolsista para o projeto ICPEDU Iara Machado 12:19
OpenCA Certificate information ca@lncc.br 11:47
OpenCA Certificate and PIN information ca@lncc.br 11:47
[Conta de Telefone] Cobrança de Ligações Telefonic... Informativo 09:42
[brasilcrypt] IBE e Redes de Sensores Diego Aranha 07:26
Snort-users Digest, Vol 9, Issue 6 snort-users-request@lists.s... 06-02-2007 19:12
[interno-l] Sistemas Dinâmicos Incertos: Controle e F... Informativo 06-02-2007 18:50
Subject: Teste
From: Luiz M. R. Gadelha Jr. <lgadelha@lncc.br>
Date: 13:13
To: Luiz M. R. Gadelha Jr.
Teste
Message Security
Message Is Signed
This message includes a valid digital signature. The message has not been altered since it was sent.
Signed by: Luiz Manoel Rocha Gadelha Junior
Email address: lgadelha@lncc.br
Certificate issued by: Autoridade Certificadora
View Signature Certificate
Message Is Encrypted
This message was encrypted before it was sent to you. Encryption makes it very difficult for other people to view information while it is traveling over the network.
OK
Unread: 0
Total: 137


Inbox - Icedove
File Edit View Go Message OpenPGP Tools Help
Get Mail Write Address Book Decrypt Reply Reply All Forward Delete Junk Print Stop
Folders
Local Folders
Inbox
Unsent
Drafts
Sent
Trash
Carol
Debian
GRADGIGA
GT-VCG
Honeypot
Honeypoy-sys
ICPEDU
Nagios
SBC
Suporte
System
TAGPMA
Date
13:13
12:55
12:19
11:47
11:47
09:42
07:26
06-02-2007 19:12
06-02-2007 18:50
Certificate Viewer:"Luiz Manoel Rocha Gadelha Junior's LNCC ID"
General Details
This certificate has been verified for the following uses:
SSL Client Certificate
Email Signer Certificate
Email Recipient Certificate
Issued To
Common Name (CN) Luiz Manoel Rocha Gadelha Junior
Organization (O) LNCC
Organizational Unit (OU) Employees
Serial Number 17
Issued By
Common Name (CN) Autoridade Certificadora
Organization (O) LNCC
Organizational Unit (OU) <Not Part Of Certificate>
Validity
Issued On 07-02-2007
Expires On 07-02-2008
Fingerprints
SHA1 Fingerprint DC:A7:21:84:A1:1E:8E:86:CB:1C:0C:23:D8:E6:DA:CD:B5:CD:7D:5D
MD5 Fingerprint 8E:BF:5A:A5:93:D6:9C:D4:E0:BF:1B:16:A2:37:11:39
Close
OK
Unread: 0
Total: 137

Aplicações: SSL/TLS

Protocolo de segurança situado entre a camada de aplicação e a camada de transporte (TCP/IP).

Largamente empregado na Internet (é utilizado no HTTPS, indicado nos navegadores como um cadeado).

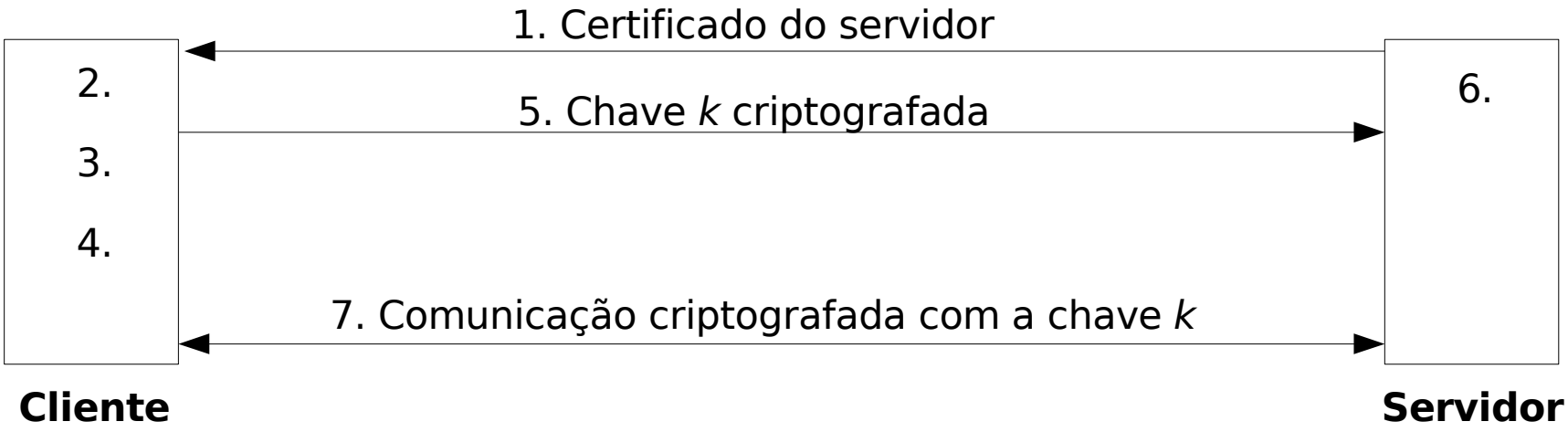
Provê confidencialidade, integridade e autenticação para aplicações de rede.

Opera dentro do modelo de ICPs.

Permite tanto autenticação do servidor como do cliente.

Aplicações: SSL/TLS

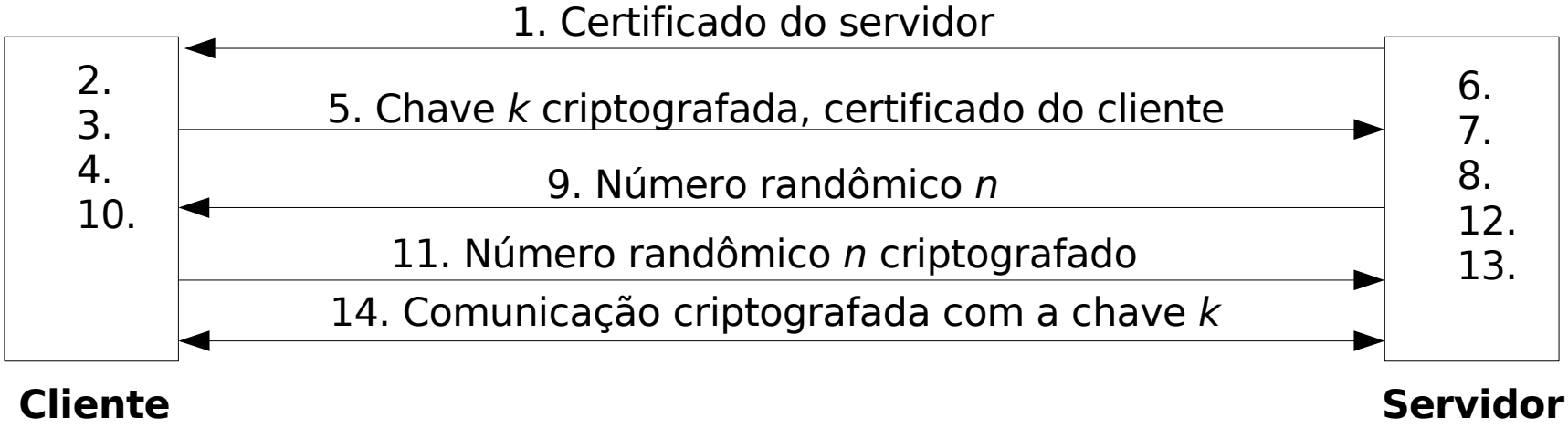
Exemplo 1: Autenticação de servidor.



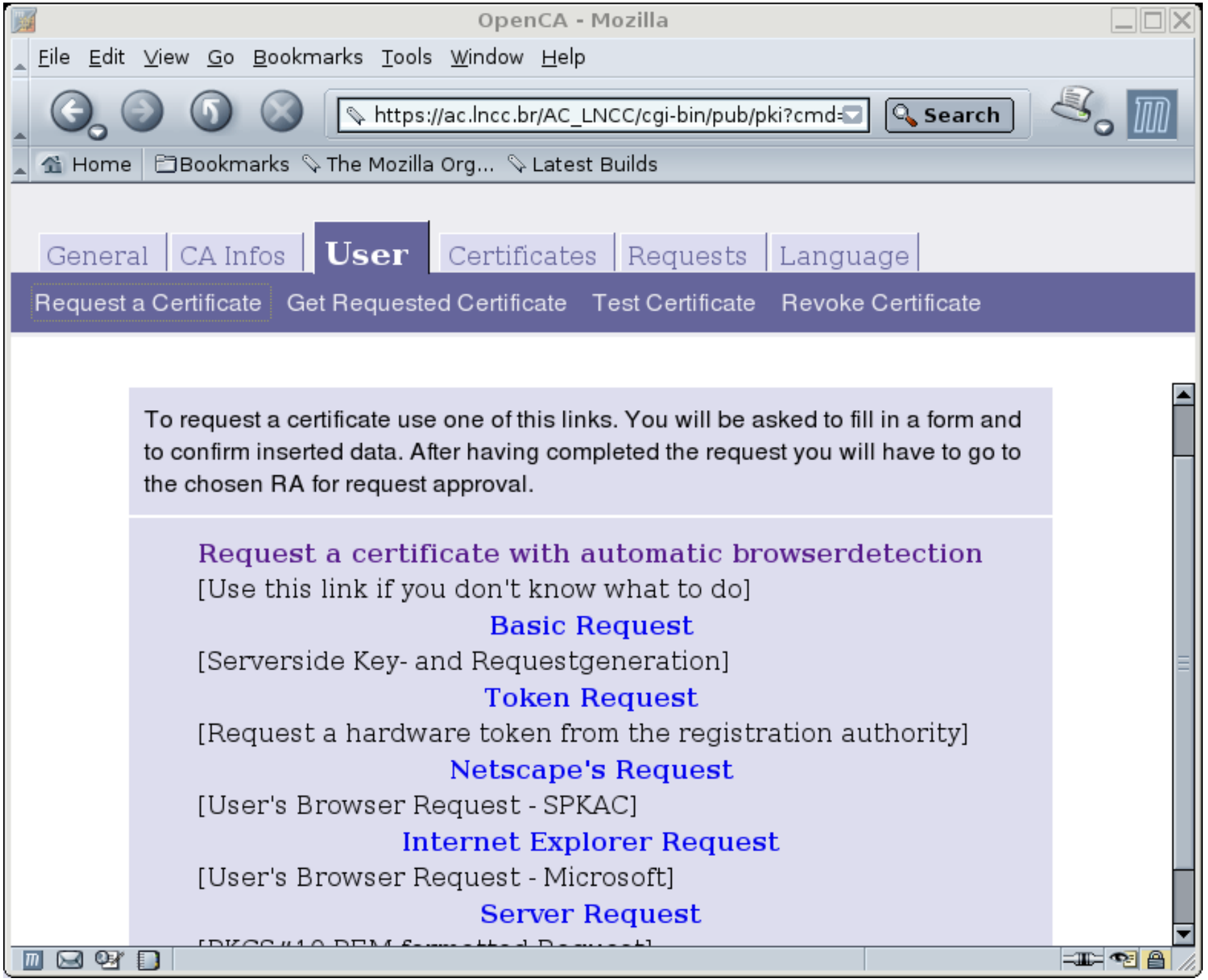
2. Verificação do certificado do servidor (caminho de certificação até certificado raiz).
3. Geração de chave de criptografia simétrica *k*.
4. Criptografia da chave *k* com chave pública do servidor.
6. Descriptografia da chave *k* com chave privada do servidor.

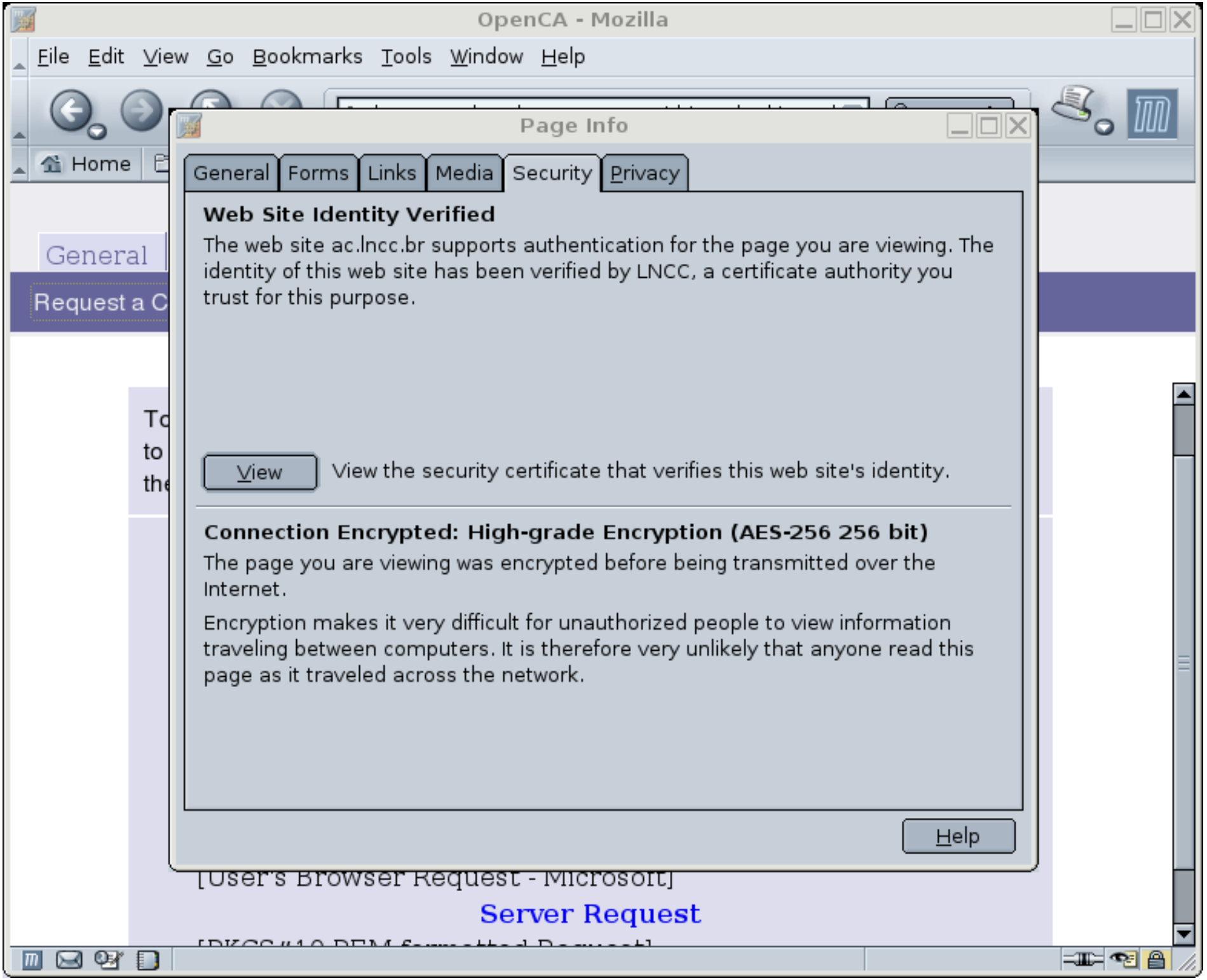
Aplicações: SSL/TLS

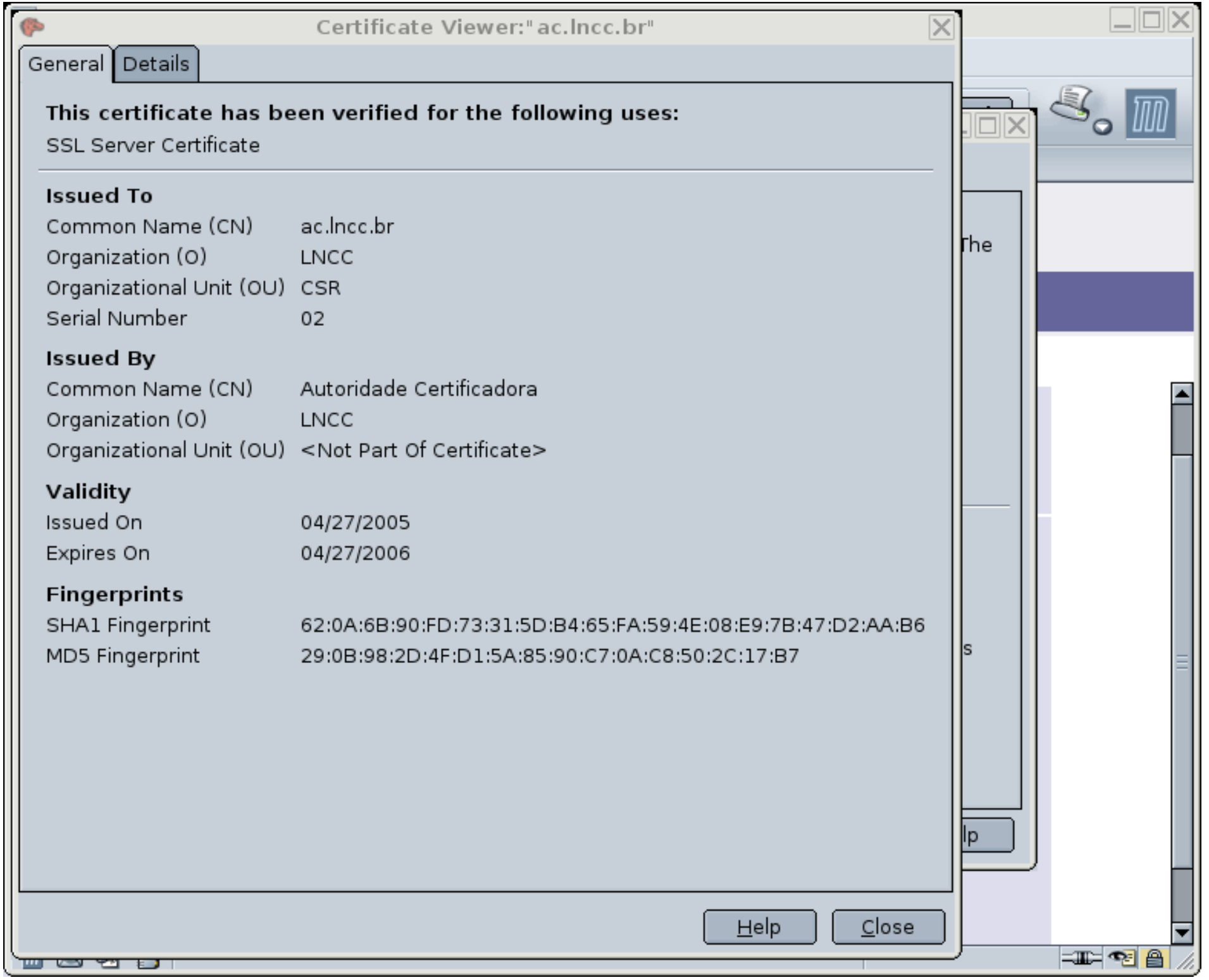
Exemplo 2: Autenticação mútua cliente-servidor.



2. Verificação do certificado do servidor (caminho de certificação até certificado raiz).
3. Geração de chave de criptografia simétrica *k*.
4. Criptografia da chave *k* com chave pública do servidor.
6. Verificação do certificado do cliente.
7. Descriptografia da chave *k* com chave privada do servidor.
8. Geração de número randômico *n* (chamado de *challenge*).
10. Criptografia de *n* com a chave privada do cliente.
12. Descriptografia do número randômico *n* com a chave pública do cliente.
13. Verificação se o resultado é igual ao *n* enviado.


OpenCA - Mozilla
File Edit View Go Bookmarks Tools Window Help
https://ac.lncc.br/AC_LNCC/cgi-bin/pub/pki?cmd=
Search
Home Bookmarks The Mozilla Org... Latest Builds
General CA Infos User Certificates Requests Language
Request a Certificate Get Requested Certificate Test Certificate Revoke Certificate
To request a certificate use one of this links. You will be asked to fill in a form and to confirm inserted data. After having completed the request you will have to go to the chosen RA for request approval.
Request a certificate with automatic browserdetection
[Use this link if you don't know what to do]
Basic Request
[Serverside Key- and Requestgeneration]
Token Request
[Request a hardware token from the registration authority]
Netscape's Request
[User's Browser Request - SPKAC]
Internet Explorer Request
[User's Browser Request - Microsoft]
Server Request


OpenCA - Mozilla
File Edit View Go Bookmarks Tools Window Help
Home
General
Request a C
Page Info
General Forms Links Media Security Privacy
Web Site Identity Verified
The web site ac.lncc.br supports authentication for the page you are viewing. The identity of this web site has been verified by LNCC, a certificate authority you trust for this purpose.
View
View the security certificate that verifies this web site's identity.
Connection Encrypted: High-grade Encryption (AES-256 256 bit)
The page you are viewing was encrypted before being transmitted over the Internet.
Encryption makes it very difficult for unauthorized people to view information traveling between computers. It is therefore very unlikely that anyone read this page as it traveled across the network.
Help
[User's Browser Request - Microsoft]
Server Request

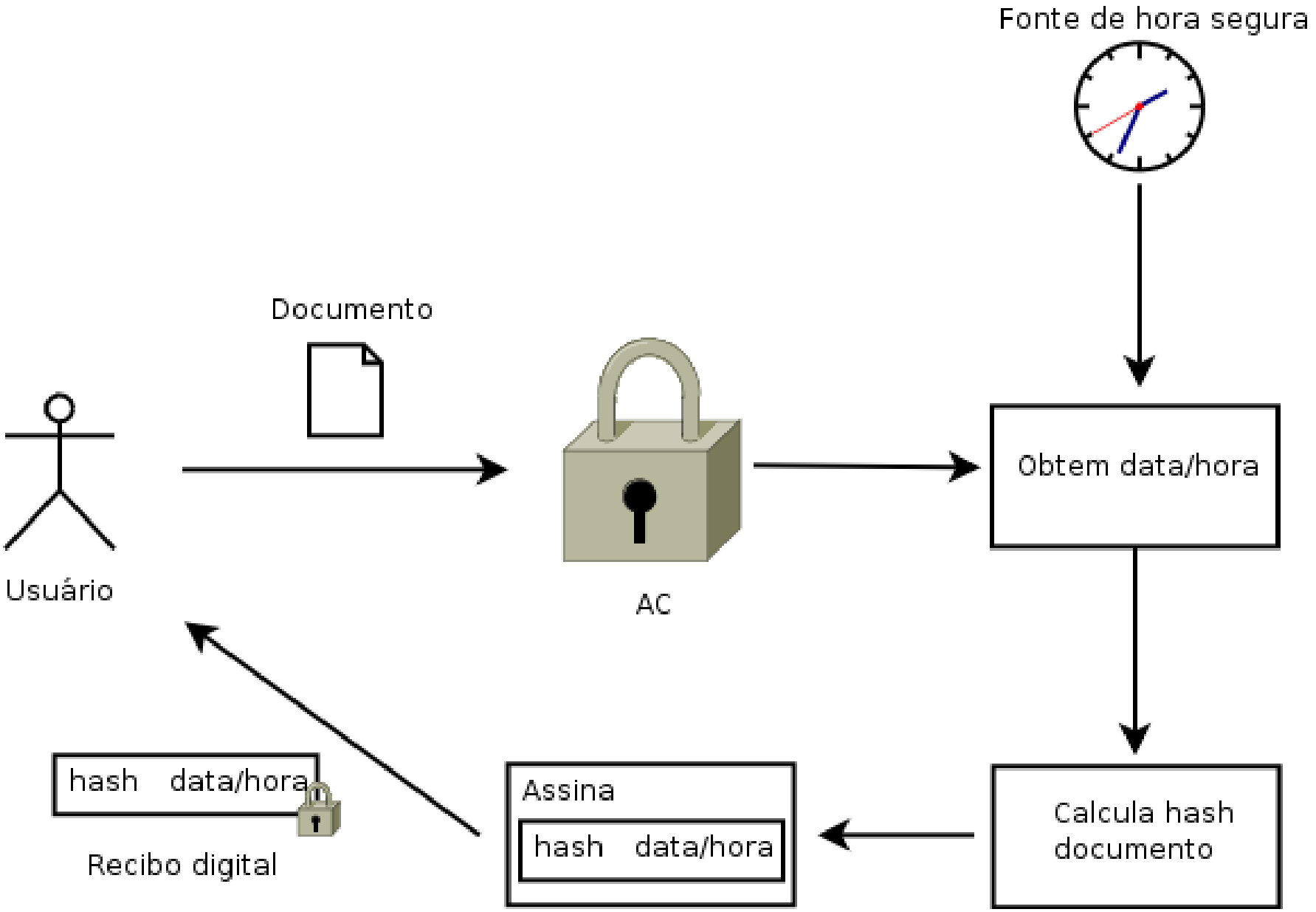

Certificate Viewer:"ac.lncc.br"
General
Details
This certificate has been verified for the following uses:
SSL Server Certificate
Issued To
Common Name (CN) ac.lncc.br
Organization (O) LNCC
Organizational Unit (OU) CSR
Serial Number 02
Issued By
Common Name (CN) Autoridade Certificadora
Organization (O) LNCC
Organizational Unit (OU) <Not Part Of Certificate>
Validity
Issued On 04/27/2005
Expires On 04/27/2006
Fingerprints
SHA1 Fingerprint 62:0A:6B:90:FD:73:31:5D:B4:65:FA:59:4E:08:E9:7B:47:D2:AA:B6
MD5 Fingerprint 29:0B:98:2D:4F:D1:5A:85:90:C7:0A:C8:50:2C:17:B7
Help
Close

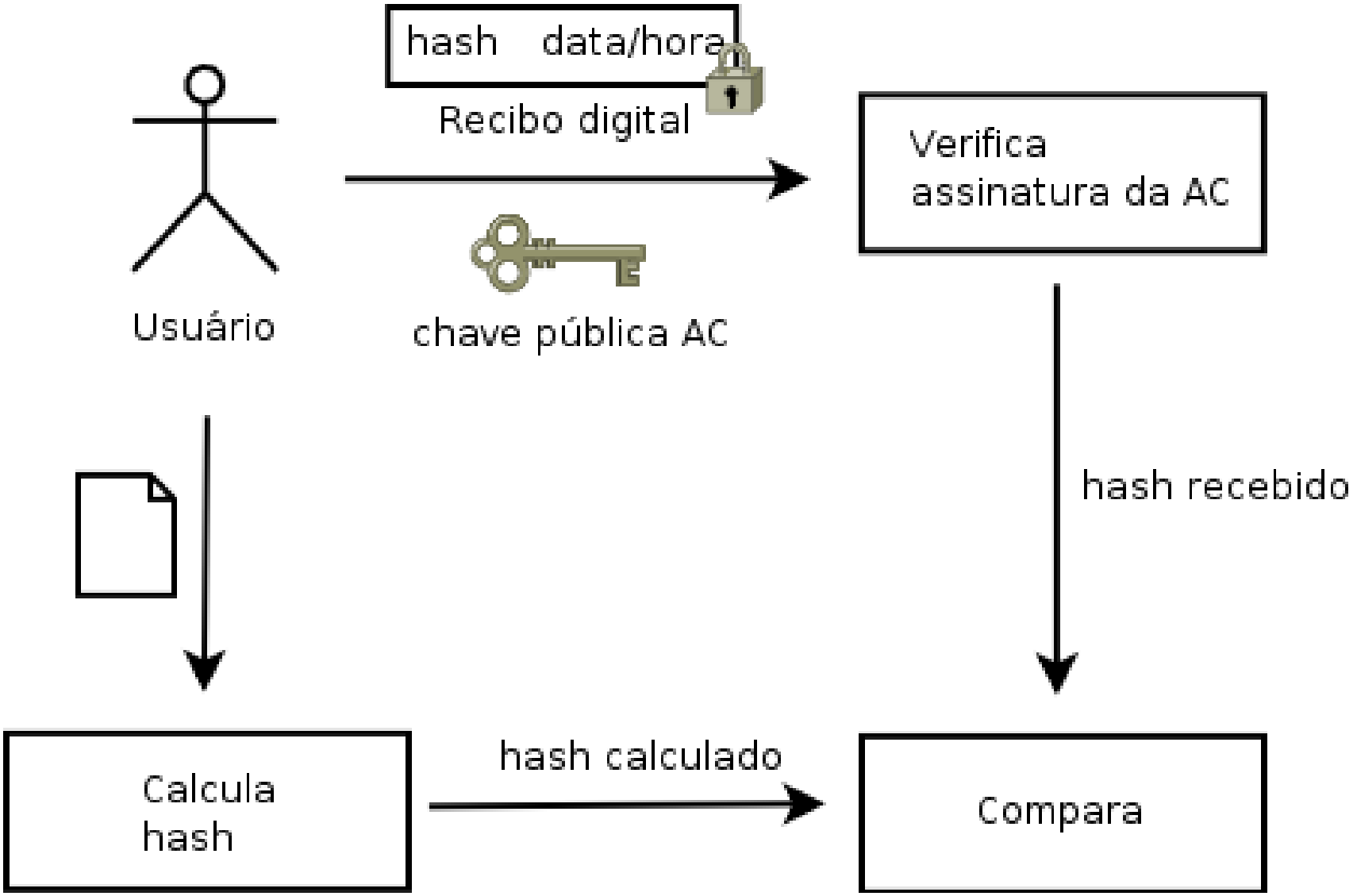
Aplicações: Datação eletrônica - *Timestamping*

Protocolo para datação eletrônica de documentos.

Permite verificar que documento existia em um determinado estado em uma data específica.

Requer que AC tenha acesso a uma fonte de hora confiável.


Fonte de hora segura
Documento
Usuário
AC
Obtem data/hora
hash data/hora
Recibo digital
Assina
hash data/hora
Calcula hash documento


hash data/hora
Recibo digital
Verifica
assinatura da AC
Usuário
chave pública AC
hash recebido
Calcula
hash
hash calculado
Compara

OpenSSL

O OpenSSL é um conjunto de ferramentas e bibliotecas de programação para trabalhar com criptografia e PKI.

`http://www.openssl.org`

Funcionalidades:

- Implementação de algoritmos de criptografia simétrica e assimétrica e funções de hash.
- Implementação de ferramentas para manipulação, gerenciamento e geração de certificados digitais.
- Biblioteca de programação (API) para linguagem C que permite habilitar SSL em aplicações.

OpenSSL

Algoritmos de criptografia simétrica:

```
> openssl enc <algoritmo> {-d} -in <entrada> -out <saída>
```

- AES
- Blowfish
- CAST
- DES
- DES3
- IDEA
- RC{2,4,5}

OpenSSL

RSA (criptografia assimétrica):

Geração de chave privada:

```
> openssl genrsa -out <saída> {-des,-des3,-aesXXX} <#bits>
```

Geração de chave pública:

```
> openssl rsa -in <ch. priv> -pubout -out <saida>
```

Criptografia de um arquivo com a chave pública:

```
> openssl rsautl -encrypt -pubin -inkey <ch. púb>
  -in <entrada> -out <saída>
```

Descriptografia de um arquivo com a chave pública:

```
> openssl rsautl -decrypt -inkey <ch. priv>
  -in <entrada> -out <saída>
```

OpenSSL

Funções de hash:

```
> openssl {md5, sha1} <entrada>
```

- MD{2,4,5}
- SHA1

Stunnel

Proxy SSL baseado no OpenSSL