Humberto José de Sousa

Infra-estrutura de Chaves Públicas com OpenCA

São José – SC

março / 2010

Humberto José de Sousa

Infra-estrutura de Chaves Públicas com OpenCA

Monografia apresentada à Coordenação do Curso Superior de Tecnologia em Sistemas de Telecomunicações do Instituto Federal de Santa Catarina para a obtenção do diploma de Tecnólogo em Sistemas de Telecomunicações.

Orientador:
Prof. Odilson Tadeu Valle

CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES
INSTITUTO FEDERAL DE SANTA CATARINA

São José – SC

março / 2010

Monografia sob o título *"Infra-estrutura de Chaves Públicas com OpenCA"*, defendida por Humberto José de Sousa e aprovada em 12 de março de 2010, em São José, Santa Catarina, pela banca examinadora assim constituída:

Prof. Odilson Tadeu Valle, M. Eng.
Orientador

Prof. Emerson Ribeiro de Mello, Dr.
IFSC

Prof. Marcelo Maia Sobral, M. Sc.
IFSC

Se o conhecimento pode criar problemas,
não é através da ignorância que podemos solucioná-los.
Isaac Asimov

Agradecimentos

Gostaria de agradecer primeiramente a Deus pela vida e saúde que possuo. Aos meus pais, irmão, irmã, namorada e aos familiares pelo apoio, carinho e incentivo. A todos os professores, amigos e colegas que contribuíram durante todo o curso, em especial ao professor Odilson, que me orientou nesse projeto. Ao ensinamento passado pelos professores do curso, que foram de grande valia para a elaboração deste trabalho.

Resumo

Este trabalho apresenta conceitos de criptografia, função de resumo, assinatura digital e Infra-estrutura de Chaves Públicas. Este trabalho baseou-se no estudo de Infra-estrutura de Chaves Públicas e um programa de código aberto que implementa uma Infra-estrutura de Chaves Públicas X.509, o OpenCA. Usando certificados digitais emitidos pelo OpenCA foi feito um experimento de trocas de e-mails assinados e cifrados.

Abstract

This work presents concepts of cryptography, hash function, digital signature and Public Key Infrastructure. This work was based on the study of Infrastructure Public Key and a open source program that implements an Public Key Infrastructure X.509, the OpenCA. Using digital certificates issued by OpenCA an experiment was done for the exchange of e-mails signed and encrypted.

Sumário

Lista de Figuras

1 Introdução p. 11

1.1 Motivação . . . . . p. 11

1.2 Organização do texto . . . . . p. 12

2 Embasamento Teórico p. 13

2.1 Criptografia . . . . . p. 13

2.1.1 Criptografia de chave secreta ou simétrica . . . . . p. 14

2.1.2 Criptografia de chave pública ou assimétrica . . . . . p. 15

2.2 Funções de resumo . . . . . p. 16

2.3 Assinatura Digital . . . . . p. 16

2.4 Infra-estrutura de chaves públicas . . . . . p. 17

2.4.1 ICP X.509 . . . . . p. 19

3 OpenCA p. 25

3.1 O projeto . . . . . p. 25

3.2 Interfaces do OpenCA . . . . . p. 26

3.3 Caso de uso 1: da requisição à emissão do certificado . . . . . p. 27

3.4 Caso de uso 2: Troca de e-mails assinados e cifrados com o ThunderBird . . p. 29

4 Conclusões p. 36

5 Apêndice - aplicativos utilizados pelo OpenCA p. 38

5.1 Apache . . . . . p. 38

5.2 mod_ssl . . . . . p. 38

5.3 OpenSSL . . . . . p. 39

5.4 OpenLDAP . . . . . p. 39

5.5 Perl . . . . . p. 40

Lista de Abreviaturas p. 41

Referências Bibliográficas p. 43

Lista de Figuras

2.1 Cifragem simétrica (NAKAMURA, 2007). . . . . p. 14

2.2 Cifragem assimétrica (NAKAMURA, 2007). . . . . p. 15

2.3 Assinatura digital (NAKAMURA, 2007). . . . . p. 17

2.4 Verificação da assinatura digital (NAKAMURA, 2007). . . . . p. 17

2.5 Modelo de confiança baseado em hierarquia. . . . . p. 18

2.6 Modelo de confiança centrada em usuário. . . . . p. 19

2.7 Estrutura do certificado X.509 (BURNETT; PAINE, 2002). . . . . p. 20

2.8 Usuário solicitando seu certificado digital dentro da ICP (NAKAMURA, 2007). p. 22

2.9 Intera (BURNETT; PAINE, 2002). . . . . p. 23

3.1 Usuário solicitando um certificado. . . . . p. 27

3.2 Confirmação de requisição de certificado. . . . . p. 28

3.3 Certificado pendente por aprovação da AR. . . . . p. 28

3.4 Certificado sendo emitido pela AC. . . . . p. 28

3.5 Usuário adquirindo o certificado solicitado. . . . . p. 29

3.6 Mensagem do navegador de certificado instalado. . . . . p. 29

3.7 Visualização do certificado no gerenciador do browser. . . . . p. 29

3.8 *Backup* do certificado. . . . . p. 30

3.9 Mensagem do navegador de *backup* efetuado com êxito. . . . . p. 30

3.10 E-mail de Alice para Beto com assinatura inválida. . . . . p. 30

3.11 Detalhe do Thunderbird sobre a assinatura inválida do e-mail. . . . . p. 31

3.12 Confiando na AC que assinou o certificado de Alice. . . . . p. 31

3.13 Detalhe do Thunderbird sobre a assinatura válida do e-mail. . . . . p. 32

3.14 E-mail de Alice para Beto com assinatura válida. . . . . p. 32

3.15 Chave pública adquirida após receber e-mail assinado. . . . . p. 33

3.16 E-mail de Beto para Alice criptografado. . . . . p. 33

3.17 Detalhe do Thunderbird sobre a criptografia do e-mail. . . . . p. 34

3.18 E-mail de Beto para Alice assinado e criptografado. . . . . p. 34

3.19 Detalhe do Thunderbird sobre a assinatura e criptografia do e-mail. . . . . p. 35

3.20 Simulação de e-mails interceptados. . . . . p. 35

1 *Introdução*

A evolução dos meios de transmissão e a disseminação das tecnologias da informação nos meios comerciais e residenciais fizeram com que se difundisse o uso de serviços de comunicação via rede.

Diariamente usa-se a Internet para executar diversas transações. É possível compartilhar arquivos e informações confidenciais via e-mail, atualizar dados bancários, fazer pagamentos, adquirir produtos, ter acesso aos resultados de exames médicos, emitir certidões com validade legal, entre outros.

Sabe-se que uso redes locais e remotas não são seguras, pois os dados estão sujeitos a interceptação enquanto trafegam da origem até o destido. Por isso é necessário o uso de mecanismos de segurança para garantir a autenticidade, integridade, confidencialidade e o não repúdio das informações digitais.

Com o uso de uma Infra-estrutura de Chaves Públicas (ICP), que provê certificação digital, é possível aumentar a segurança em transações eletrônicas através de uma rede não segura.

Este trabalho apresenta os principais conceitos utilizados na certificação digital e os componentes de uma ICP. É apresentando também a ICP OpenCA, que é um programa de código aberto, que em conjunto com outros programas, também de código aberto, implementam uma ICP.

1.1 Motivação

A motivação deste trabalho foi o interesse na Infra-estrutura de Chaves Públicas e verificar como este tipo de tecnologia pode auxiliar ainda mais na difusão de serviços via rede, principalmente os serviços que requeiram segurança na troca de informações. Durante o desenvolvimento de todo o trabalho foi possível conhecer os processos envolvidos no funcionamento de uma ICP, bem como usar o conhecimento adquirido ao longo do curso.

1.2 Organização do texto

O texto está organizado da seguinte forma:

No capítulo 2 são apresentados conceitos de criptografia e os principais componentes de uma ICP.

No capítulo 3 é apresentado o aplicativo OpenCA e suas principais características, a requisição de certificados e o uso destes certificados para troca de e-mails.

No capítulo 4 são apresentadas as conclusões e as propostas para trabalhos futuros.

No capítulo 5 são apresentados os aplicativos utilizados pelo OpenCA.

2 *Embasamento Teórico*

2.1 Criptografia

A origem da palavra criptografia é grega. Esta palavra é formada pela junção de *kryptos* (oculto, envolto, escondido) e *graphos* (escrever). Pode-se entender então que criptografia é a escrita em códigos.

O objetivo da criptografia é possibilitar a troca de mensagens entre duas partes através de um canal não seguro, de tal maneira que uma terceira parte não consiga entender as mensagens. Para isso são usados métodos e técnicas para tornar um texto claro em um texto cifrado e vice-versa.

A criptografia pode prover cinco diferentes papéis. Cada um desses papéis diferentes é identificado por uma determinada palavra-chave. Os papéis são(GARFINKEL, 2002):

Autenticação – A assinatura digital pode ser usada para a identificação dos participantes de uma transação eletrônica ou o autor de uma mensagem de e-mail; pessoas que recebem uma mensagem que foi assinada por uma assinatura digital pode verificar a identidade do assinante.

Controle de acesso – Considerando que a autenticação é usada para determinar a identidade de um participante, técnicas de autorização são usadas para determinar se essa pessoa está autorizada a realizar uma operação em particular.

Confidencialidade – Criptografia é usada para embaralhar as informações, enviadas através de redes e armazenadas em servidores, para que curiosos não pode acessar o conteúdo dos dados.

Integridade – Métodos utilizados para verificar se a mensagem não foi modificada durante a transmissão. Normalmente associada com a possibilidade de evitar o acesso as informações por terceiros.

Não repúdio – Recibos de criptografia são criados de forma que um autor de uma mensagem não pode negar o envio da mensagem.

2.1.1 Criptografia de chave secreta ou simétrica

Na criptografia de chave secreta ou simétrica, as informações são codificadas e decodificadas fazendo uso de uma mesma chave. Para tal é necessário o compartilhamento desta chave. Na figura 2.1 é ilustrado a criptografia de chave simétrica, em que João esta mandando uma mensagem cifrada com uma chave privada para Maria, e Maria está usando uma cópia desta mesma chave para decifrar a mensagem.

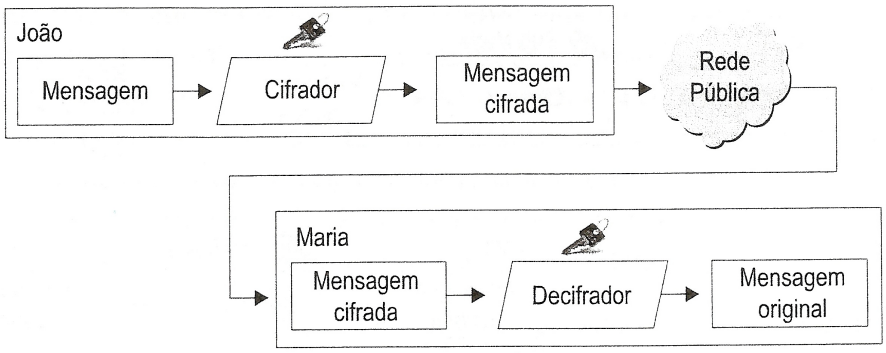

Figura 2.1: Cifragem simétrica (NAKAMURA, 2007).

Um grande problema no uso de chaves simétricas é como se vai compartilhar essa chave entre as partes envolvidas. Se um terceiro conseguir capturar a chave no momento da distribuição, ou de qualquer outra forma, as mensagens cifradas com esta chave secreta poderão ser decifradas facilmente por este terceiro, pois a chave secreta foi comprometida.

Quando um terceiro intercepta uma mensagem, um bom algoritmo simétrico deve impedir que a chave seja descoberta por ataque de força bruta[1]. Para isso é necessário escolher uma boa chave, com letras maiúsculas e minúsculas, números e caracteres especiais. Uma boa chave pode fazer com que esse tipo de ataque leve anos ou décadas para encontrar a chave secreta.

Alguns exemplos de algoritmos de criptografia de chave secreta são:

- DES (*Data Encryption Standard*) (KARN; METZGER; SIMPSON, 1995a);
- 3DES[2] (*Triple DES*) (KARN; METZGER; SIMPSON, 1995b);

[1]Ataque por força bruta é quando se tenta encontrar a chave necessária para decifragem explorando todas as possibilidades.

[2]Esta técnica usa a técnica DES três vezes, aumentando o comprimento da chave de 56 bits para 168 bits.

- IDEA (*International Data Encryption Algorithm*) (TEIWES; HARTMANN; KUENZI, 2001);
- RC2 (*Rivest Ciphers 2*) (RIVEST, 1998);
- RC4 (*Rivest Ciphers 4*) (JAGANATHAN; ZHU; BREZAK, 2006);
- AES (*Advanced Encryption Standard*) (CHOWN, 2002).

2.1.2 Criptografia de chave pública ou assimétrica

O surgimento da criptografia de chave pública ou assimétrica, foi motivada pelo problema de compartilhamento de chaves criptográficas privadas. Na criptografia de chave assimétrica cada usuário tem um par de chaves, uma pública e outra privada. Cada chave privada tem um chave pública correspondente. Uma mensagem cifrada com a chave pública, poderá ser decifrada somente com a chave privada correspondente e vice-versa.

Na figura 2.2 é ilustrado a criptografia de chaves públicas. Nesta ilustração, João esta mandando uma mensagem cifrada com uma chave pública de Maria, para a própria Maria. Quando Maria recebe a mensagem cifrada, ela pode usar a chave privada dela para decifrar a mensagem. Como ninguém além de Maria tem acesso a chave privada, somente ela poderá ter acesso ao conteúdo da mensagem.

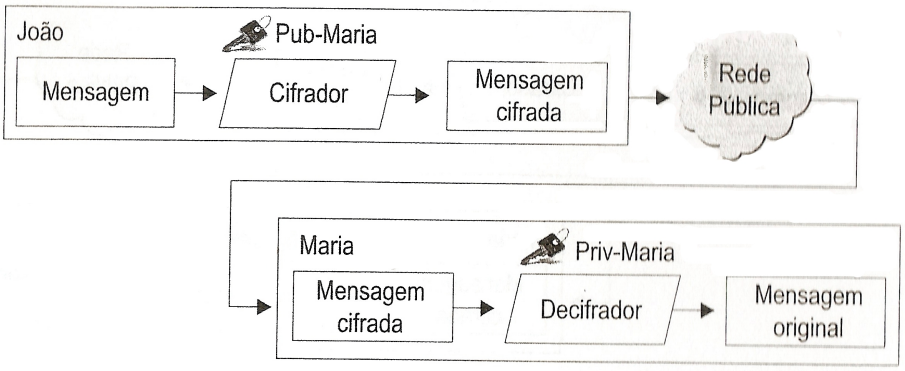

Figura 2.2: Cifragem assimétrica (NAKAMURA, 2007).

Como a criptografia assimétrica é mais cara computacionalmente e a criptografia simétrica tem o problema de compartilhamento da chave, a criptografia assimétrica pode ser usada em conjunto com a criptografia simétrica, assim o problema de compartilhamento da chave simétrica e o problema de desempenho da chave assimétrica são solucionados(NAKAMURA, 2007). Com o uso desses dois algorítimos, pode-se aproveitar o melhor da característica de cada um.

Com referência a figura 2.2, ao invés de João enviar uma mensagem qualquer, ele pode enviar uma chave simétrica. A partir deste momento as próximas mensagens são cifradas com essa chave simétrica.

Apesar de vários algoritmos de chave pública terem sido propostos, como RSA, Rabin e outros, o mais utilizado atualmente continua sendo o primeiro proposto em 1976, o RSA[3] (COULOURIS, 2007). Vale lembrar que é muito difícil conceber um algorítimo com as características do RSA.

2.2 Funções de resumo

Uma função de resumo ou *hash* é uma função que gera, a partir de uma mensagem de qualquer tamanho, um resumo de tamanho fixo que identifica a mensagem original. As funções *hash* são unidirecionais, isto é, não se pode chegar na mensagem original a partir do resumo. Um dos objetivos dos algoritmos de resumo é tornar muito difícil que se encontre duas mensagens diferentes que resultem em um mesmo resumo.

Segundo (COULOURIS, 2007), duas funções resumo amplamente usadas para aplicações práticas são o algoritmo MD5[4] (*Message-Digest algorithm 5*) e o SHA-1 (*Secure Hash Algorithm*) (RFC 3174).

2.3 Assinatura Digital

Considerada análoga a assinatura no papel, a assinatura digital tem o objetivo de fornecer: autenticidade, integridade e não-repúdio. A assinatura digital é realizada através de criptografia assimétrica e funções de resumo.

Na figura 2.3 é ilustrado um esquema de assinatura digital. Inicialmente João gera um resumo da mensagem que ele quer enviar e assina esse resumo com sua chave privada, que somente ele deve ter acesso. Este resumo assinado é adicionado a mensagem e enviado para Maria.

Para verificar a assinatura da mensagem, como ilustra a figura 2.4, Maria faz uso da chave pública de João para verificar se a assinatura foi gerada pela chave privada correspondente. Quando essa verificação é positiva, é obtido o resumo assinado por João. Esse resumo é com-

[3] O algoritmo de chave pública RSA foi proposto por Ron Rivest, Adi Shamir e Leonard Adleman. A sigla RSA veio das iniciais dos sobrenomes deles.

[4] É o quinto em uma sequência de algoritmos de resumos de mensagem desenvolvidos por Ron Rivest.

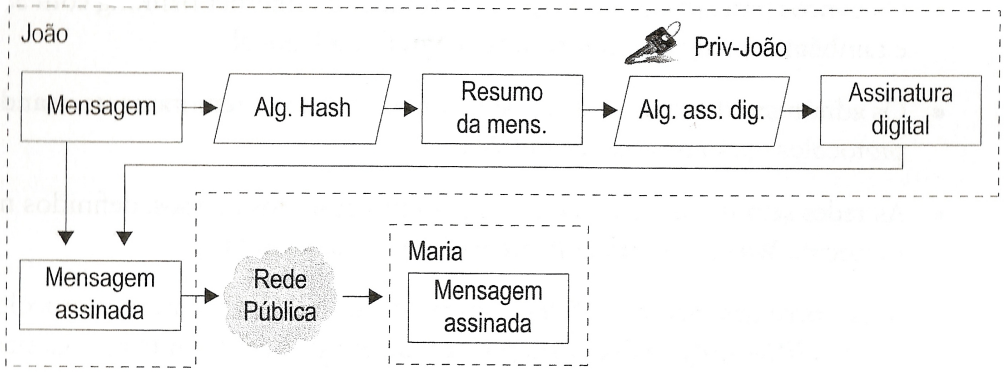

Figura 2.3: Assinatura digital (NAKAMURA, 2007).

parado com o resumo da mensagem gerado por Maria. Se os resumos forem iguais a assinatura digital é válida. Caso contrário a assinatura é inválida.

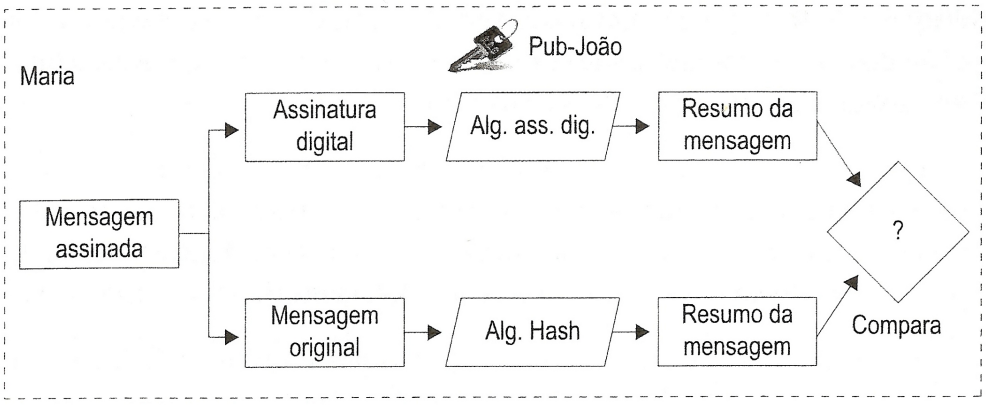

Figura 2.4: Verificação da assinatura digital (NAKAMURA, 2007).

2.4 Infra-estrutura de chaves públicas

Uma ICP ou *Public Key Infrastructure* (PKI), para (CUSTÓDIO; GRAAF; DAHAB, 2003), é um conjunto de ferramentas e processos para a implementação e a operação de um sistema de emissão de certificados digitais baseado em criptografia de chaves públicas. Engloba também os detalhes do sistema de credenciamento e as práticas e políticas que fundamentam a emissão de certificados e outros serviços relacionados.

Já para (SILVA; CARVALHO; TORRES, 2003), uma ICP é a combinação de software,

cifra e serviços que permitem a proteção e autenticação de comunicações digitais, através da gestão segura de chaves criptográficas. Esta tecnologia integra certificados digitais, criptografia de chaves públicas e autoridades de certificação numa arquitetura de segurança abrangente, introduzindo um elemento de garantia na criptografia assimétrica.

Os modelos de confiança são utilizados para descrever o relacionamento entre os usuários finais, partes verificadoras e a Autoridade Certificadora (BURNETT; PAINE, 2002). Dois modelos muito conhecidos são:

Hierárquicos neste modelo a confiança é propagada no sentido de cima para baixo e cria um caminho único de cada folha até o nível mais alto da árvore. A autoridade certificadora no topo na árvore delega outras autoridades para emitirem certificados em seu nome. Essas podem ainda delegar outras autoridades certificadoras e assim por diante, até que uma delas emita um certificado. A figura 2.5 ilustra este modelo.

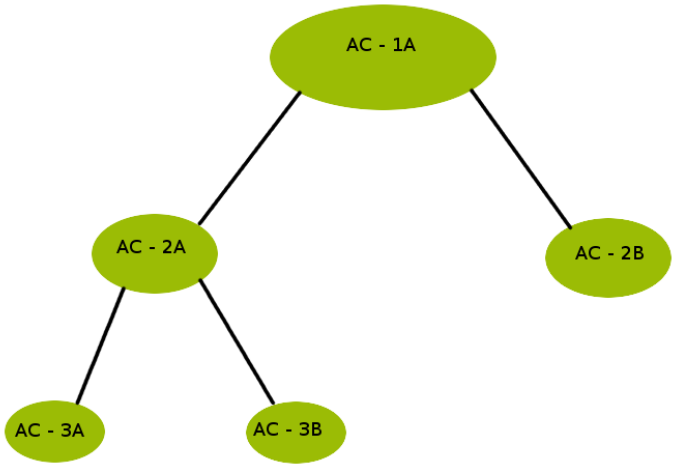

Figura 2.5: Modelo de confiança baseado em hierarquia.

Centrados no usuário neste modelo não existe uma figura central ditando uma hierarquia, cada usuário decide em quem quer confiar e também quanto quer confiar. Uma medida importante é quanto você confia na honestidade e no julgamento de uma pessoa que possui uma chave. Quanto mais você confiar em uma chave, mais você confia na pessoa que criou a chave e em outras pessoas que essa chave certifica. Cada usuário pode assinar os certificados de outros usuários. A assinatura em um certificado é o compromisso feito pelo signatário (quem assina) que a chave realmente pertence a pessoa cujo nome e endereço de email contam no certificado. Por exemplo, se Beto confia em Ana, e Ana confia em Maria, então Beto confia em Maria. Este modelo é a base do PGP *Pretty Good Privacy*. A figura 2.6 ilustra este modelo.

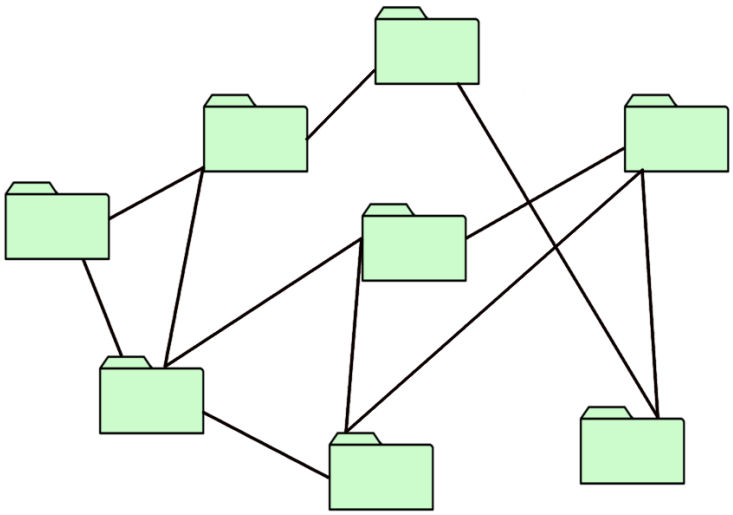

Figura 2.6: Modelo de confiança centrada em usuário.

Neste trabalho iremos focar no estudo do modelo hierárquico definido pelo padrão X.509 do ITU-T.

2.4.1 ICP X.509

O X.509 é o formato padrão mais usado para certificados (COULOURIS, 2007). A recomendação ITU-T X.509 faz parte das séries X.500[5] para construção de diretórios globais de nomes e atributos.

O X.509 define uma estrutura que fornece serviços de autenticação através do diretório X.500 para seus usuários. O diretório pode servir como um repositório de certificados de chave pública. Cada certificado contém a chave pública de um usuário e é assinado com a chave privada de uma AC confiável. Além disso, X.509 define protocolos de autenticação alternativos, baseado no uso de certificados de chave pública.

X.509 é baseada na utilização de criptografia de chave pública e assinatura digital. A norma não obriga a utilização de um algoritmo específico, mas recomenda RSA. O esquema de assinatura digital requer o uso de uma função *hash*. Novamente, a norma não cita um algoritmo de *hash* específico para ser utilizado.

[5]Série de padrões para redes de computadores que trata de serviços de diretório.

Certificados digitais X.509

Os certificados digitais são um meio seguro de distribuir para as partes verificadoras as chaves públicas dentro de uma rede. Em sua forma mais básica o certificado X.509 contém chave pública, a identidade do dono e a identificação da parte que atesta a validade deste certificado.

O padrão original do certificado X.509 que foi publicado em 1988 já teve duas alterações, a primeira em 1993 e a última em 1995. A figura 2.7 apresenta a estrutura de um certificado X.509. Todas as versões dos certificados x.509 contêm os seguintes campos(BURNETT; PAINE, 2002):

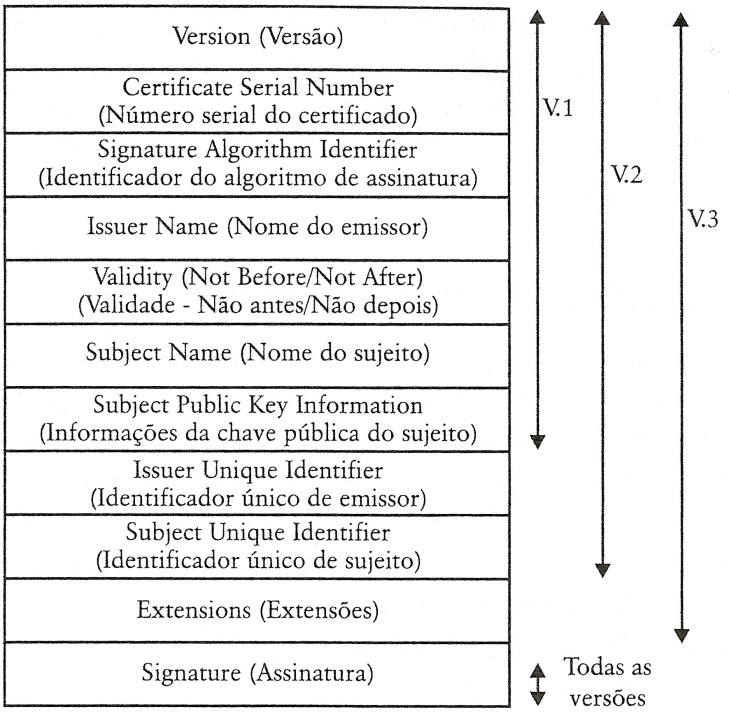

Figura 2.7: Estrutura do certificado X.509 (BURNETT; PAINE, 2002).

Versão esse campo aponta qual a versão do certificado.

Número serial do certificado esse campo contém um valor inteiro gerado pela AC que identifica cada certificado.

Identificador do algoritmo de assinatura esse campo armazena o algoritmo utilizado para assinar o certificado.

Nome do emissor esse campo possui o nome da AC que assinou o certificado.

Validade - Não antes/Não depois esse campo contém dois valores de data e hora que definem o período em que o certificado pode ser considerado válido.

Nome do sujeito esse campo identifica o dono do certificado.

Informações sobre a chave pública do sujeito esse campo contém a chave pública do dono e identifica o algoritmo utilizado.

Os certificados digitais normalmente são criados pelas ACs. A manutenção da AC envolve a segurança da sua própria chave privada que, se descoberta ou roubada, invalidará todos os certificados emitidos.

Autoridade Certificadora

A Autoridade Certificadora (AC) é uma entidade que emite certificados digitais. A AC é uma terceira parte que confirma as informações do certificado digital. Quando uma AC emite um certificado, ela assina este certificado com sua chave privada. Com o uso da chave pública da AC, as partes interessadas podem verificar a autenticidade de um certificado em questão.

A Autoridade Certificadora (AC) tem várias funções, dentre elas estão: geração, renovação e revogação de certificados digitais. Ela é responsável também por publicar os certificados e a Lista de Certificados Revogados (LCR), que ela mesma gera. A AC pode usar o protocolo OCSP (*Online Certificate Status Protocol*) para consulta do estado dos certificados, não precisando da LCR.

Uma Autoridade Certificadora Raiz (AC-Raiz) é a primeira autoridade certificadora da ICP hierárquica, AC-1A na Figura 2.5. A AC-Raiz está no topo da hierarquia e tem seu certificado auto assinado. A AC-Raiz usualmente delega a outras ACs (AC-2A, 2B, 3A e 3B na Figura 2.5) a tarefa de emitir certificados em seu nome com o objetivo de distribuir a carga de tarefas, para isso ela assina o certificado dessas ACs. Outra tarefa delegada é a de identificação dos usuários. Esta tarefa é feita pela Autoridade de Registro (AR) que será descrita na seção 2.4.1.

As ACs podem ser privadas ou públicas. As públicas operam via internet e fornecem serviços de autenticação para o público geral, usuários e empresas. As ACs privadas estão inseridas normalmente em uma corporação, onde se busca um controle de acesso e uma autenticação mais rigorosa.

Quanto a segurança as ACs necessitam de uma estrutura segura para guardar os dados dos clientes sigilosamente. A AC-Raiz normalmente necessita de sala cofre, sistema contra

incêndio, sistema conta inundação e proteção contra queda de energia elétrica. A AC intermediária não precisa de uma sala cofre, basta uma sala com acesso restrito.

Autoridade de Registro

A Autoridade de Registro (AR) é uma instituição associada a uma AC que faz a interface entre o usuário e a AC. Ela é responsável por receber o solicitante, cadastrar requisições e receber os documentos de identificação (pessoas ou empresas). Após a conferência dos dados, a AR envia uma solicitação à AC para a emissão do certificado.

A figura 2.8 exemplifica um dos papeis da AR na ICP.

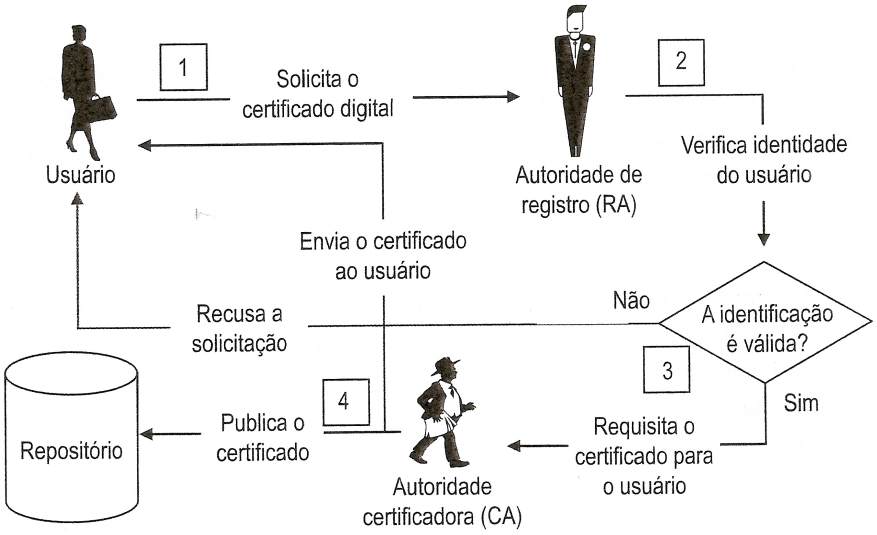

Figura 2.8: Usuário solicitando seu certificado digital dentro da ICP (NAKAMURA, 2007).

Diretório de certificado

O diretório de certificados serve para armazenar os certificados para um uso futuro. Isso evita que o usuário final necessite armazenar o certificado em máquinas locais. A ideia desse repositório é fornecer um ponto único para administração e distribuição dos certificados.

Diretórios X.500 estão se tornando mais amplamente aceitos porque além do fato de atuarem como repositório de certificados, eles fornecem aos administradores uma localização central das informações de atributos pessoais. O *Lighweight Directory Access Protocol* (LDAP) foi projetado para fornecer aos aplicativos um meio mais simples e fácil para utilizar o padrão X.500(BURNETT; PAINE, 2002).

Lista de Certificados Revogados

A Lista de Certificados Revogados (LCR) é uma lista de certificados que a AC já emitiu, mas por alguma razão não mais os considera válidos. Isso pode ocorrer quando um certificado é emitido por engano ou quando se perde o sigilo da chave privada desse certificado. Apesar da assinatura dos certificados estarem corretas e validade dentro do prazo, a AC tem de indicar que eles já não podem ser confiáveis. É da responsabilidade de qualquer parte que confia em outro certificado, verificar com a AC se o certificado não foi revogado.

As principais informações contidas em uma LCR são: números seriais dos certificados revogados, informações da AC emissora juntamente com sua assinatura, data e o motivo da revogação do certificado. A RFC 3280 (HOUSLEY, 2002) define o conteúdo da LCR.

OCSP

Dependendo do tamanho de uma ICP, trabalhar com LCR pode ser complicado e requisitar muitos recursos de rede para obter uma cópia dessa lista. O *Online Certificate Status Protocol* (OCSP) é um protocolo de internet usado para obter o estado de revogação de um certificado digital. Este foi criado como uma alternativa à LCR. Esse protocolo implementa um método simples de requisição e resposta para obter o estado do certificado, exemplificado na figura 2.9. O OCSP é descrito pela RFC 2560 (MYERS, 1999).

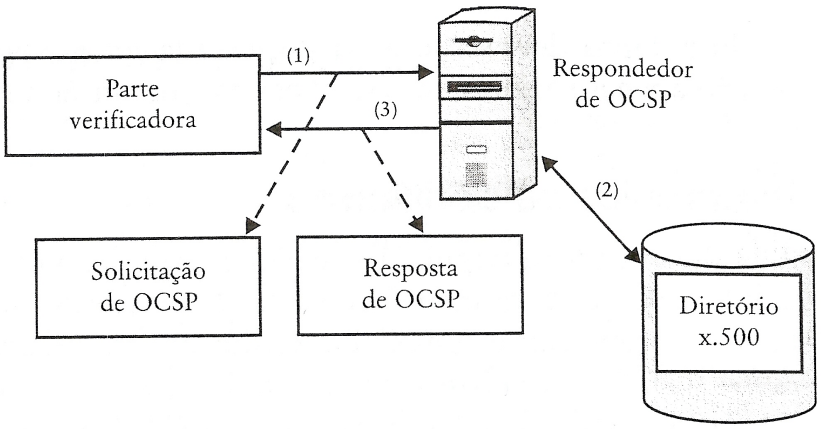

Figura 2.9: Intera (BURNETT; PAINE, 2002).

Suspendendo um certificado

As vezes se tem a necessidade de suspender o uso de um certificado ao invés de revogá-lo. Um exemplo disso é quando um funcionário de uma empresa entra em férias. Para evitar a revogação do certificado e posteriormente uma nova emissão pela AC, o certificado pode ser suspenso. Quando o funcionário voltar de férias o pedido de suspensão é retirado e o certificado volta a ser válido. Para suspender um certificado, a AC utiliza o valor Certificate Hold na extensão Reason Code da LCR (BURNETT; PAINE, 2002).

Cadeia de certificado X.509

Uma cadeia de certificado é o método mais comum utilizado para verificar a associação entre uma entidade e sua chave pública (BURNETT; PAINE, 2002). Para se confiar no certificado, é preciso verificar três coisas relacionadas a cada certificado até alcançar uma raiz confiável:

- Verificar se cada certificado da cadeia está assinado com a chave pública do próximo certificado da cadeia;
- Assegurar que nenhum desses certificados estão expirados ou revogados;
- Verificar se a autoridade raiz que assinou esse certificado indiretamente é de confiança.

Políticas de certificado e instruções da prática de certificação

As autoridades certificadoras, ao assinarem um certificado, confirmam o conteúdo deste certificado. Dois mecanismos são utilizados pelas ACs para estabelecer confiança entre os usuários finais e as partes verificadoras, políticas de certificado e instruções da prática de certificação(BURNETT; PAINE, 2002).

O padrão X.509 define uma política de certificado como “um conjunto nomeado de regras que indica a aplicabilidade de um certificado para uma comunidade em particular e/ou para uma classe de aplicativo com requisitos comuns de segurança”(BURNETT; PAINE, 2002). Nas extensões do certificado X.509 Versão 3 pode ser identificado uma ou mais políticas de certificado. Com base nessas informações pode-se confiar ou não em um determinado certificado.

A prática certificação é um conjunto de instruções que a autoridade certificadora emprega para emitir um certificado. Com base nestas informações é possível a decisão se confia-se em uma determinada autoridade certificadora.

3 *OpenCA*

3.1 O projeto

O projeto OpenCA é um projeto de código aberto que iniciou em 1999. Esse projeto usa um conjunto de aplicativos também de código aberto para construir uma ICP modelo X.509. Os seguintes aplicativos são utilizados:

Apache As interfaces de administração e de usuário são em formato Web, então o Apache funciona como esse servidor Web;

mod_ssl O mod_ssl é um módulo que fornece criptografia para o Apache. As interfaces de administração do OpenCA por padrão são somente acessadas via HTTPS. O Apache2 não necessita deste módulo, pois ele possui uma solução integrada;

OpenSSL O OpenSSL é uma biblioteca criptográfica usada pelo OpenCA. Esta biblioteca é responsável por toda a tarefa criptográfica requerida;

OpenLDAP O OpenLDAP é um protocolo de acesso aos diretórios padrão X.500. O OpenCA utiliza estes diretórios para armazenar principalmente certificados;

Perl É a linguagem de programação utilizada pelo OpenCA em scripts, que através das páginas Web, interage com os outros aplicativos. O OpenCA utiliza também vários módulos Perl.

A versão atual do OpenCA PKI é 1.1.0. Essa versão, lançada no dia 24-02-10, é muito mais fácil de instalar, pois os módulos necessários são instalados juntos com o OpenCA. Mesmo assim o processo de instalação não é trivial, mas observando atentamente os arquivos de configuração é possível obter êxito. Informações mais detalhadas sobre a instalação e configuração podem ser encontradas no guia do OpenCA (OPENCA, 2009a).

OpenCA é projetado para uma infra-estrutura distribuída. Com ele pode-se construir toda uma hierarquia com três ou mais níveis. OpenCA não é apenas uma solução para pequenas

instalações de pequenas e médias empresas. O objetivo é apoiar a máxima flexibilidade para grandes organizações como universidades, redes e empresas globais.

3.2 Interfaces do OpenCA

Através de interfaces Web é possível administrar o OpenCA. Os administradores têm acesso as interfaces AC, AR, Node e LDAP. Os usuários têm acesso a interface Pub.

Abaixo será descrito as principais funções de cada interface.

Node Esta interface gerencia o banco de dados e processa todas as funcionalidades de exportação e importação. A interface inclui algumas funções de *backup* (não inclui a chave privada) e recuperação. É possível configurar diferentes regras para sincronização com nós de um maior e menor nível da hierarquia.

AC A interface AC tem todas as funções necessárias para criar certificados e LCR. Somente através desta interface é possível alterar a configuração AC.

AR Esta interface é capaz de lidar com todos os tipos de solicitações. Estas solicitações são: pedidos de edição, aprovação de solicitações, criação de chaves privadas com cartões inteligentes, excluir os pedidos errados e os usuários de e-mail.

LDAP A interface LDAP foi projetada para separar a gestão LDAP completamente do resto do software. Essa interface foi separada porque há muitas funções específicas para administradores LDAP e poucos usuários necessitariam destes recursos.

Pub A interface pública compreende todas as pequenas coisas que os usuários precisam. As principais opções disponíveis na interface Pub são:

- Gera solicitação de certificado;
- Consulta de certificados;
- Consulta de LCR;
- Gera solicitação de revogação;
- Testes de certificados de usuário.

3.3 Caso de uso 1: da requisição à emissão do certificado

Após o OpenCA instalado e configurado, é demonstrado um exemplo de solicitação de certificado. Nesse exemplo o usuário não tem acesso as interfaces da AC e da AR, essas interfaces somente os respectivos administradores têm acesso. As etapas aqui descritas não seguem qualquer política de certificação.

Um usuário qualquer solicita um certificado através da interface pública como ilustra a figura 3.1. Nessa solicitação é necessário o preenchimento de alguns dados que vão identificar o usuário. Após a confirmação dos dados pelo usuário, é gerado uma confirmação de requisição de certificado como na figura 3.2. Os dados desta solicitação são importantes para depois pegar o certificado.

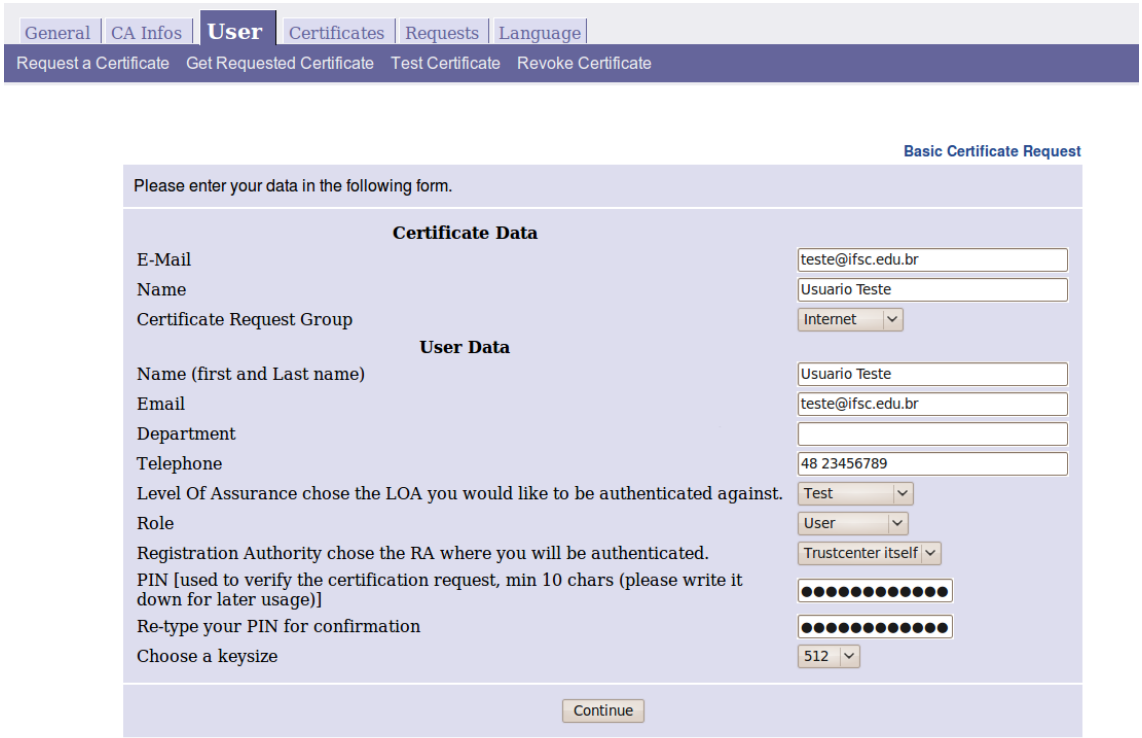

Figura 3.1: Usuário solicitando um certificado.

Após o usuário ter solicitado o certificado, cabe inicialmente a AR aprovar o certificado. Assim é necessário que o solicitante dirija-se a AR para confirmar seus dados. Na figura 3.3 aparece o certificado solicitado pendente por aprovação.

Após a AR confirmar os dados e aprovar a solicitação, a AC precisa emitir e confiar nos dados do certificado. Um dos processos de emissão do certificado é ilustrado na figura 3.4.

Por fim, o usuário pode adquirir o certificado solicitado, como mostra na figura 3.5. Neste

General | CA Infos | **User** | Certificates | Requests | Language

Request a Certificate Get Requested Certificate Test Certificate Revoke Certificate

Certificate Request Confirm

Thank you for requesting your certificate from our organization, your request with the serial 1568 it's been successfully archived and it is now waiting for approval by any of our Registration Authorities (if you are unsure about the receiving of your request by this server, you can check the list of new requests).
To complete the certification process you have to go to one of our Registration Authority office with one of the following documents: o ID card or passport. o Documnetation asserting your role and authorization for requesting a certificate for your organization. If you still have doubts about the issuing process, just use the links provided in the Information section to learn how to complete all the needed steps.

ADDITIONAL_ATTRIBUTE_DEPARTMENT	
ADDITIONAL_ATTRIBUTE_EMAIL	teste@ifsc.edu.br
ADDITIONAL_ATTRIBUTE_REQUESTERCN	Usuario Teste
ADDITIONAL_ATTRIBUTE_TELEPHONE	48 23456789
LOA	10
NOTBEFORE	Tue Dec 29 20:59:59 2009 GMT
PIN	00df26136c085a480c83ca547291d82fdb218ec5
RA	Trustcenter itself
ROLE	User
SERIAL	1568
TYPE	SPKAC

Print

Figura 3.2: Confirmação de requisição de certificado.

General | **Active CSRs** | Active CRRs | Information | Utilities | Language

New Renewed Pending (be processed already) Waiting for additional signature

New Certificate Signing Requests

Tue Dec 29 16:01:54 2009

Serial	Submit Name	Submitted On	Requested Role	
n/a	1568	emailAddress=teste@ifsc.edu.br, CN=Usuario Teste, OU=Internet, O=OpenCA, C=US	n/a	User Test

No Extra References

Figura 3.3: Certificado pendente por aprovação da AR.

General | **Usual Operations** | Batch System | Active CSRs | Active CRRs | Information | Language

Approved Certificate Requests Approved Revocation Requests Issue new CRL

Approved Certificate Signing Requests

Tue Dec 29 16:06:43 2009

Operator	Serial	Submit Name	Approved On	Requested Role	
n/a	1568	emailAddress=teste@ifsc.edu.br, CN=Usuario Teste, OU=Internet, O=OpenCA, C=US	n/a	User	Test

No Extra References

Figura 3.4: Certificado sendo emitido pela AC.

caso o certificado é instalado no navegador, como ilustra a figura 3.6. É possível fazer o *backup* do certificado e protege-lo com senha como ilustra as figuras 3.7, 3.8 e 3.9.

Figura 3.5: Usuário adquirindo o certificado solicitado.

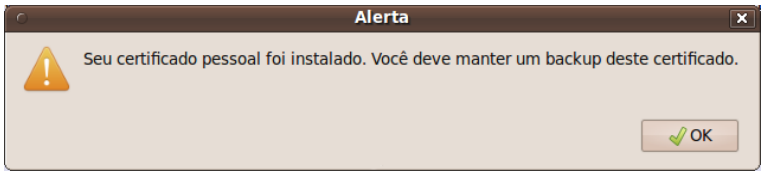

Figura 3.6: Mensagem do navegador de certificado instalado.

Figura 3.7: Visualização do certificado no gerenciador do browser.

3.4 Caso de uso 2: Troca de e-mails assinados e cifrados com o ThunderBird

No exemplo a seguir será demonstrado trocas de e-mails assinados e criptografados. Para tal, foram criadas duas contas de e-mail no Gmail e configuradas no cliente de e-mail Thunder-

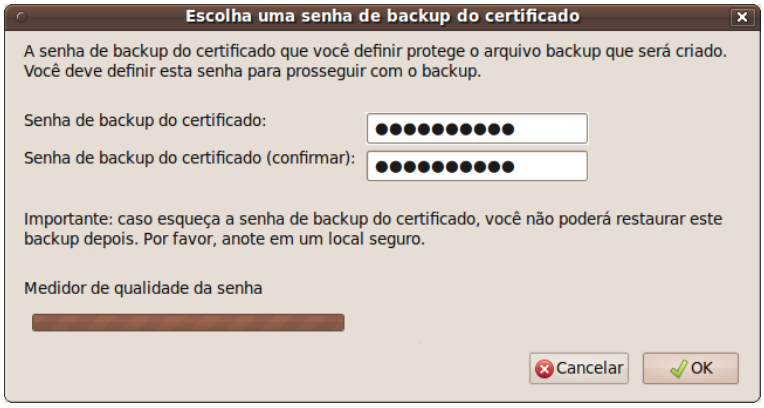

Figura 3.8: *Backup* do certificado.

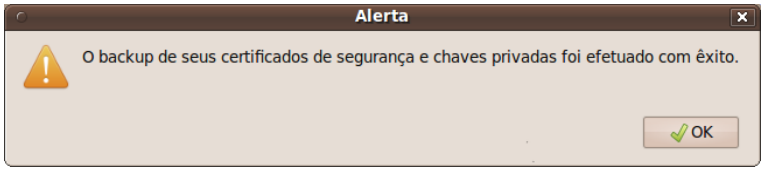

Figura 3.9: Mensagem do navegador de *backup* efetuado com êxito.

bird. Neste exemplo o e-mail user01ifsc@gmail.com pertence a Alice e user02ifsc@gmail.com pertence a Beto, de acordo com a figura 2.2.

No primeiro exemplo, Alice manda uma mensagem assinada para Beto, mas Beto não confia na AC que assinou o certificado de Alice. Como Beto não confia no certificado de Alice, ele se torna inválido para Beto. As figuras 3.10 e 3.11 ilustram o exemplo.

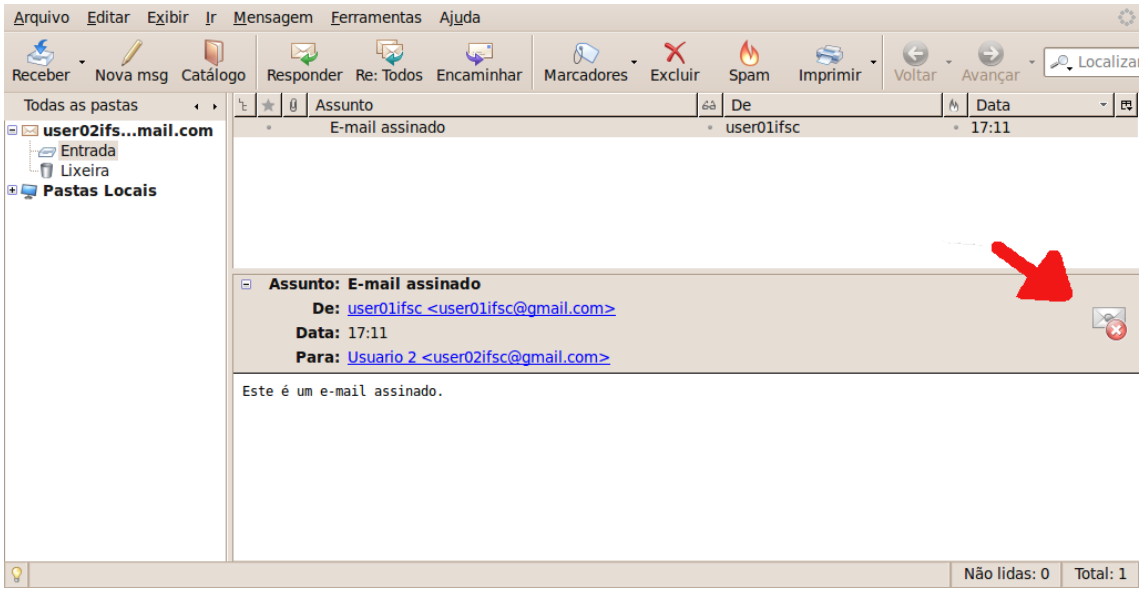

Figura 3.10: E-mail de Alice para Beto com assinatura inválida.

Se Beto passa a confiar na AC que emitiu o certificado de Alice, então a assinatura de Alice

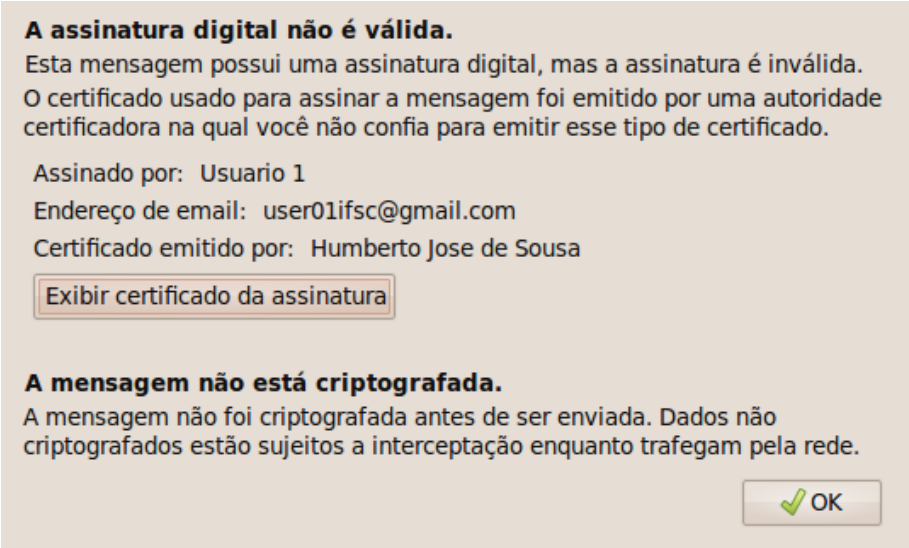

Figura 3.11: Detalhe do Thunderbird sobre a assinatura inválida do e-mail.

passa a ter validade. Quando o certificado da AC é instalado, ele pode ser usado para identificar *sites*, usuários ou autores de programas. Ao aceitar o certificado da AC, ilustrado na figura 3.12, o e-mail de Alice passa a ser válido para Beto, como se pode ver nas figuras 3.13 e 3.14.

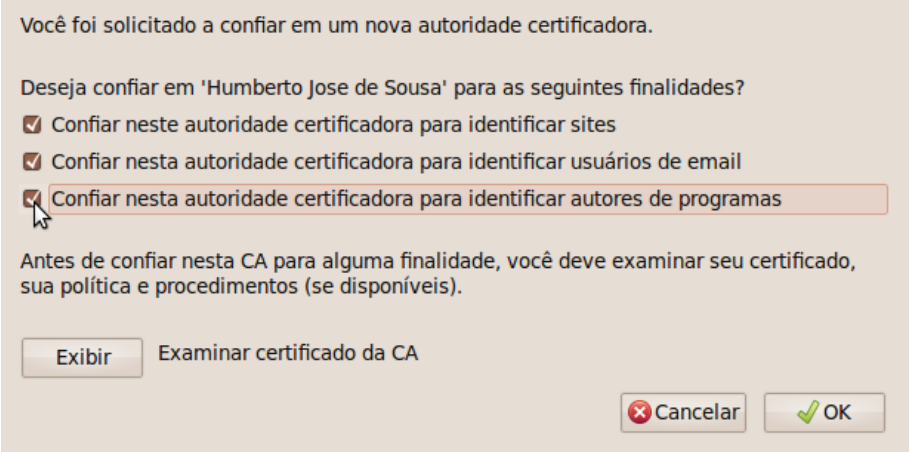

Figura 3.12: Confiando na AC que assinou o certificado de Alice.

Ao receber uma mensagem assinada, obtém-se a chave pública desse usuário, como ilustra a figura 3.15. Com esta chave pública pode-se mandar um e-mail criptografado que somente com a chave privada poderá ser descriptografado.

No segundo exemplo, Beto está enviando um e-mail criptografado para Alice, pois ele possui a chave pública de Alice. Este e-mail somente poderá ser descriptografado com a chave privada de Alice, que somente ela deve possuir. As figuras 3.16 e 3.17 ilustram o exemplo.

Um e-mail pode ser assinado e criptografado, como pode-se ver nas figuras 3.18 e 3.19. Neste caso o e-mail foi assinado com a chave privada de Beto e criptografada com a chave

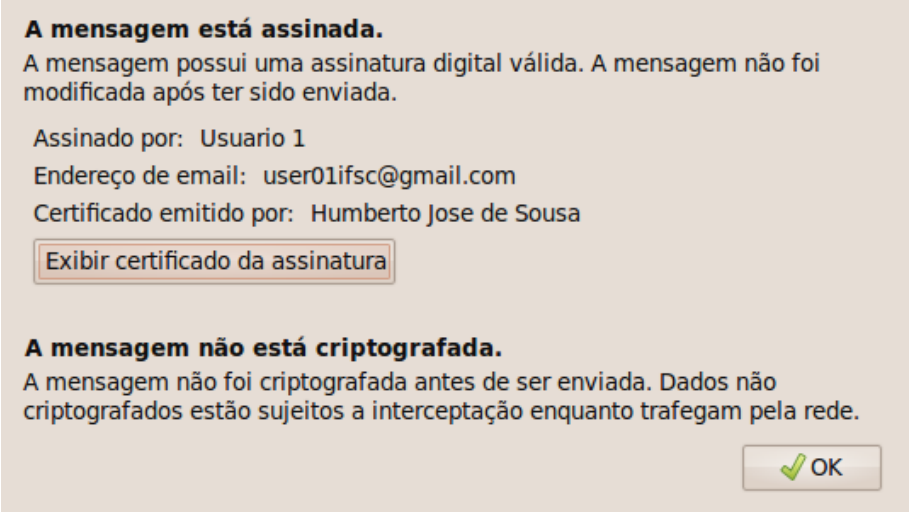

Figura 3.13: Detalhe do Thunderbird sobre a assinatura válida do e-mail.

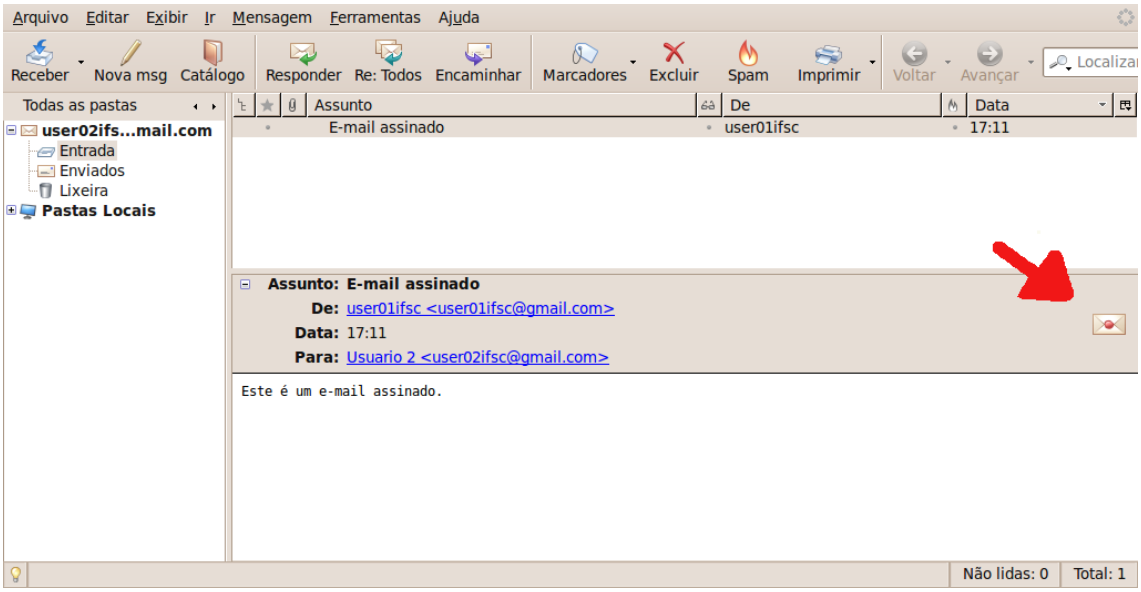

Figura 3.14: E-mail de Alice para Beto com assinatura válida.

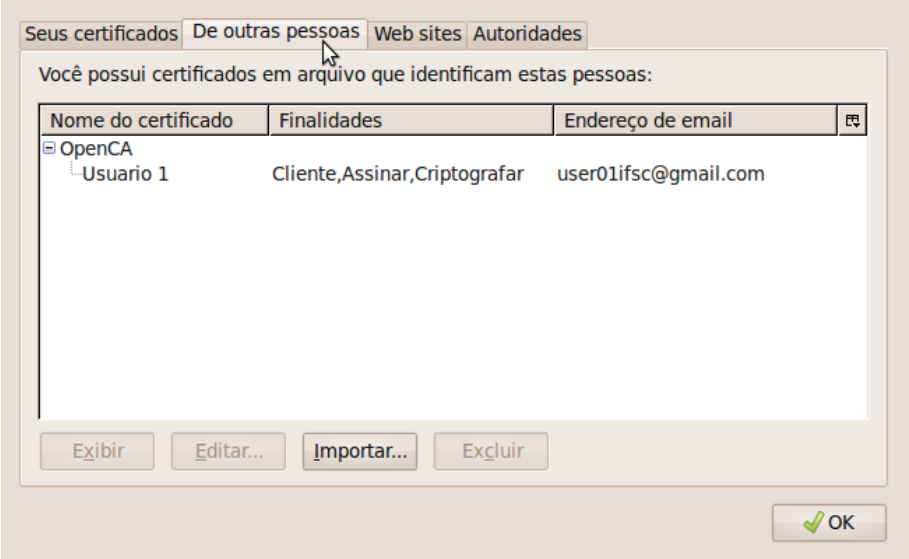

Figura 3.15: Chave pública adquirida após receber e-mail assinado.

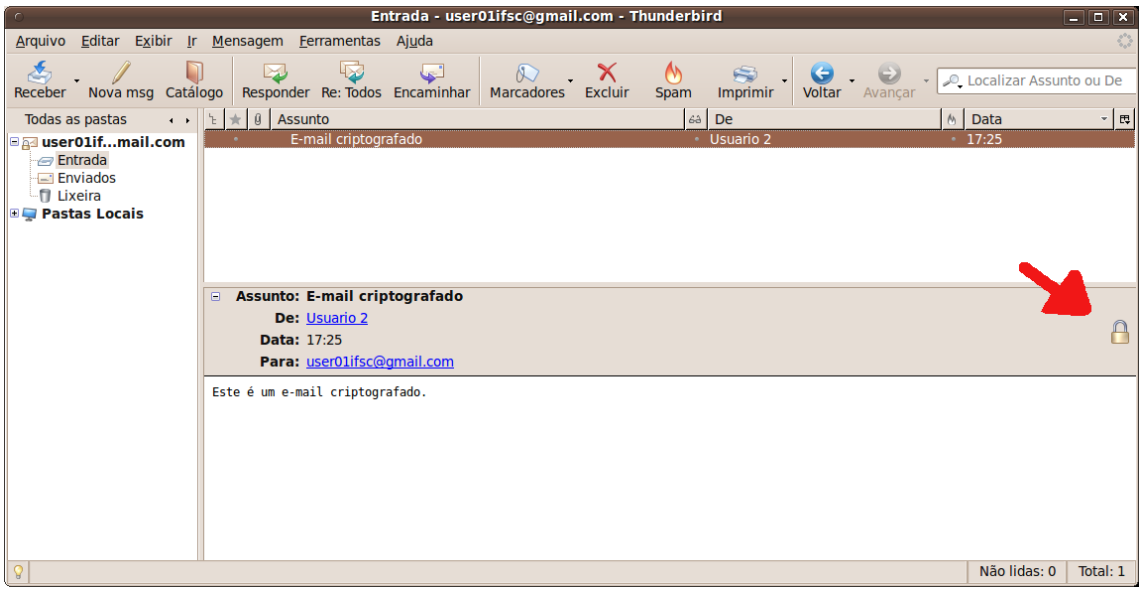

Figura 3.16: E-mail de Beto para Alice criptografado.

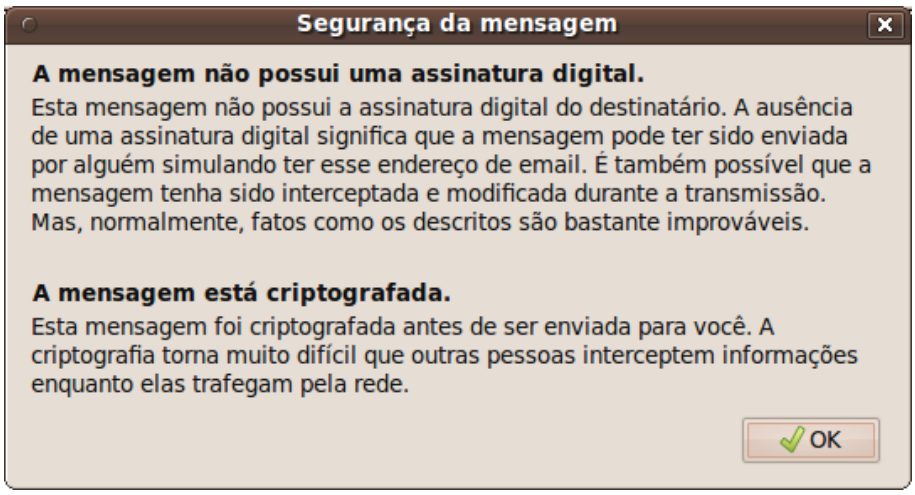

Figura 3.17: Detalhe do Thunderbird sobre a criptografia do e-mail.

pública de Alice.

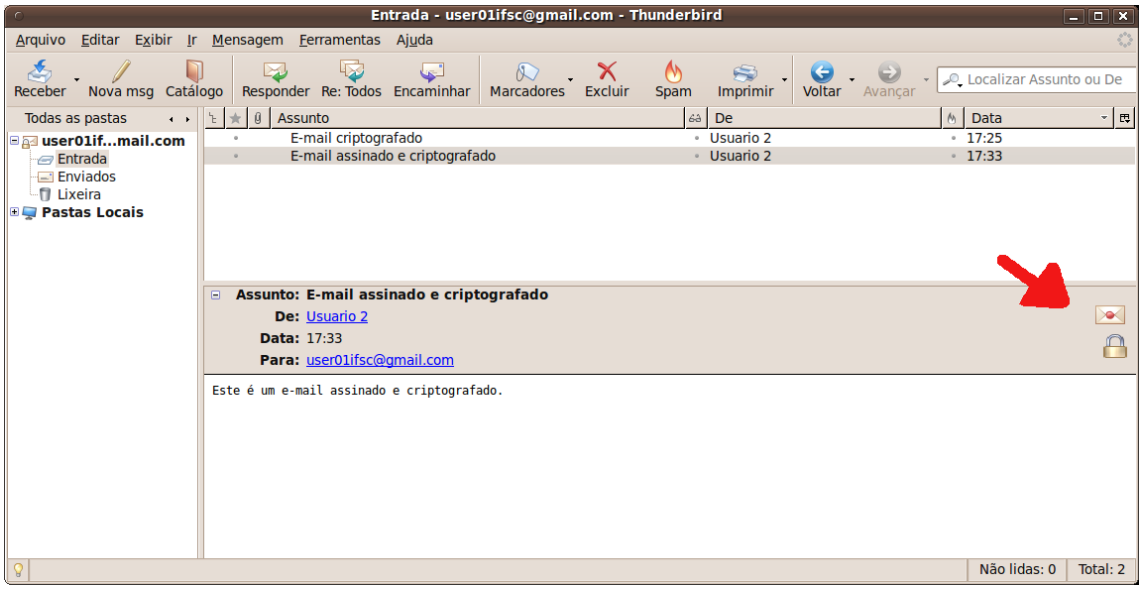

Figura 3.18: E-mail de Beto para Alice assinado e criptografado.

Como Beto mandou dados sigilosos para Alice, ele tomou cuidado de criptografar o e-mail. Com isso, se alguém no meio do caminho interceptar o e-mail, terá grande dificuldade para ler o conteúdo. A figura 3.20 ilustra a simulação de dois e-mails enviados por Beto para Alice interceptado por um terceiro.

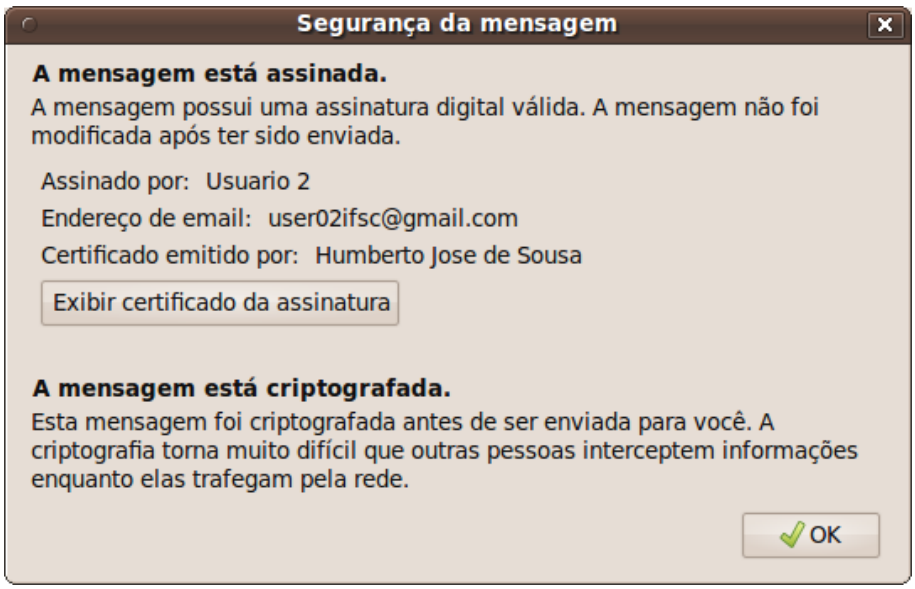

Figura 3.19: Detalhe do Thunderbird sobre a assinatura e criptografia do e-mail.

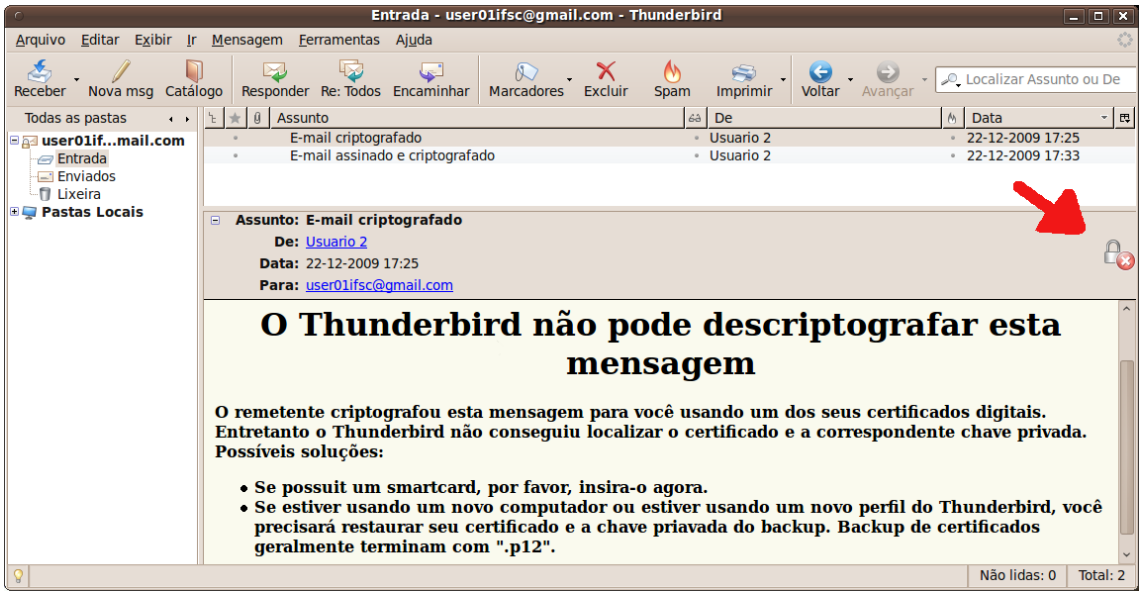

Figura 3.20: Simulação de e-mails interceptados.

4 Conclusões

Este trabalho apresentou uma opção de Infra-estrutura de Chaves Públicas utilizando o OpenCA e outros programas gratuitos de código aberto. O OpenCA em conjunto com outros softwares disponibiliza uma aplicação com custo zero em programas que provê segurança aos usuários.

Durante a elaboração deste trabalho foram estudados temas como: criptografia, funções de resumo e assinatura digital que propiciaram um bom entendimento sobre estas tecnologias e sua função em uma Infra-estrutura de Chaves Públicas. O estudo dos componentes que compõem uma Infra-estrutura de Chaves Públicas contemplou o entendimento sobre a função de cada componente.

Um grande problema ao longo deste trabalho foi a falta de documentação sobre o programa OpenCA. Apesar de possuir um guia, ele ainda esta bem incompleto. Uma vantagem é que por ser um programa gratuito e de código aberto, os usuários trocam informações e estão sempre dispostos a ajudar outros usuários.

Com os testes práticos descritos, podê-se trocar e-mails assinados para confirmar a autenticidade do remetente e que a mensagem não foi alterada no meio do caminho. Com o uso da criptografia garantiu-se sigilo, pois somente o destinatário conseguiu abrir a mensagem com a chave privada.

Com o uso da certificação digital é possível disponibilizar serviços com maior agilidade e menor burocracia. Outra vantagem é a redução de custos pela desmaterialização de processos através do uso de documentos digitais, além da economia com o gasto de papel, provê agilidade no trâmite de documentos.

O capítulo 1 apresentou uma breve introdução sobre segurança digital e Infra-estrutura de Chaves Públicas. O capítulo 2 apresentou conceitos sobre criptografia de chave privada e de chave pública, funções de resumo, assinatura digital e as partes que compõem uma Infra-estrutura de Chaves Públicas. O capítulo 3 apresentou o projeto OpenCA e uma breve descrição de suas interfaces e características. No capítulo 3 foram apresentados alguns passos necessários

para acertificação digital, desde a solicitação de um certificado pelo usuário até a emissão do certificado pela AC; e alguns testes com assinatura e criptografia de e-mails.

Como trabalho futuro, sugere-se o estudo de outras ferramentas de código livre que implementam uma Infra-estrutura de Chaves Públicas para possíveis comparações. Outra sugestão é o estudo mais aprofundado afim de colaborar com a documentação do programa. E por último sugere-se implantar o ICPEDU, que é uma estrutura hirárquica de chaves públicas, no IFSC, que foi o objetivo inicial deste projeto mas infelizmente não foi alcançado.

5 *Apêndice - aplicativos utilizados pelo OpenCA*

5.1 Apache

O Apache é um servidor Web de código aberto que processa solicitações HTTP[1] (Hypertext Transfer Protocol), protocolo padrão da Web. O Apache também suporta o protocolo HTTPS[2] (*Hypertext Transfer Protocol Secure*).

Apache é o mais conhecido e usado servidor Web. Ele possui excelente performance, segurança e compatível com diversas plataformas. Entre os códigos que o Apache pode executar estão: PHP (*Hypertext Preprocessor*), Perl, Shell Script e ASP (*Active Server Pages*).

5.2 mod_ssl

Mod_ssl é um módulo que fornece criptografia para o Apache através dos protocolos SSL e TLS. Esse módulo usa a biblioteca OpenSSL para prover tal criptografia. Algumas das características do mod_ssl:

- É um software código aberto;
- Compatível com plataformas Unix e Win32;
- Possui criptografia de 128 bits;
- Suporte para protocolos SSLv2, SSLv3 e TLSv1;
- É baseado no certificado X.509 para autenticação de cliente e servidor.

[1] protocolo de aplicação responsável pelo tratamento de pedidos e respostas entre cliente e servidor na *World Wide Web*

[2] O HTTP combinado com a camada de segurança SSL

5.3 OpenSSL

O OpenSSL é um projeto de código aberto que implementa os protocolos SSL e TLS e uma biblioteca criptográfica de uso geral. O projeto é gerido por uma comunidade mundial de voluntários que usam a Internet para se comunicar, planejar e desenvolver o kit de ferramentas OpenSSL e sua documentação.

OpenSSL é baseado na biblioteca SSLeay, desenvolvida por Eric A. Young e Tim J. Hudson. A biblioteca OpenSSL foi escrita na linguagem C e está disponível para sistemas Linux, Windows, BSD e Mac OS X.

5.4 OpenLDAP

O OpenLDAP é um software livre de código aberto que implementa o protocolo LDAP (*Lightweight Directory Access Protocol*). O LDAP é um serviço de diretório baseado no padrão X.500 que foi desenvolvido para atualização e pesquisa de diretórios que roda sobre o protocolo TCP/IP[3].

O pacote OpenLDAP inclui:

- Servidor slapd;
- Bibliotecas que implementam o protocolo LDAP;
- Utilitários e ferramentas.

Algumas características do OpenLDAP são:

- Banco de dados não relacional otimizado para leitura;
- Organização hierárquica;
- Otimizado para atender grande número de requisições;
- Leve e robusto;
- Suporte a múltiplas instâncias de dados;
- Suporte a SASL (*Simple Authentication and Security Layer*) e a TLS/SSL.

[3]TCP/IP é um conjunto de protocolos de comunicação entre computadores em rede. Seu nome vem de dois protocolos: o TCP (*Transmission Control Protocol*) e o IP (*Internet Protocol*).

5.5 Perl

Perl (*Practical Extraction and Report Language*) (WALL, 2009) é uma linguagem de programação de uso geral, originalmente desenvolvida para manipulação de textos, que agora é utilizada para uma infinidade de tarefas incluindo administração de sistemas, desenvolvimento Web, programação de redes, desenvolvimento de interfaces gráficas, e muitos outros. A linguagem foi escrita para ser prática (fácil de utilizar, eficiente, completa) ao invés de bonita (pequena, elegante, mínima). Suas maiores características são sua facilidade de uso, que inclui o suporte tanto para programação procedural quanto para programação orientada à objetos, tem um poderoso suporte incluso na linguagem para processamento de textos, e uma das maiores coleções de módulos escritos por terceiros, o CPAN[4].

[4] *Comprehensive Perl Archive Network* (CPAN), é uma grande coleção de software Perl e documentação.

Lista de Abreviaturas

3DES *Triple DES*

AC Autoridade Certificadora

AC-Raiz Autoridade Certificadora Raiz

AES *Advanced Encryption Standard*

AR Autoridade Registradora

ASP *Active Server Pages*

BSD *Berkeley Software Distribution*

CPAN *Comprehensive Perl Archive Network*

CRR *Certificate Revocation Request*

CSR *Certificate Signing Request*

DES *Data Encryption Standard*

FTP *File Transfer Protocol*

HTTP *Hypertext Transfer Protocol*

HTTPS *Hypertext Transfer Protocol Secure*

ICP Infra-Estrutura de Chaves Públicas

IDEA *International Data Encryption Algorithm*

IP *Internet Protocol*

LCR Lista de Certificados Revogados

LDAP *Lightweight Directory Access Protocol*

MD5 *Message-Digest algorithm 5*

OCSP *Online Certificate Status Protocol*

OpenLDAP *Open Lightweight Directory Access Protocol*

OpenSSL *Open Secure Socket Layer*

Perl *Practical Extraction and Report Language*

PHP *Hypertext Preprocessor*

PKI *Public Key Infrastructure*

RC2 *Rivest Ciphers 2*

RC4 *Rivest Ciphers 4*

RFC *Request for Comments*

RSA *Rivest, Shamir e Adleman*

SASL *Simple Authentication and Security Layer*

SHA-1 *Secure Hash Algorithm*

SSL *Secure Socket Layer*

TCP *Transmission Control Protocol*

TLS *Transport Layer Security*

Referências Bibliográficas

ALECRIM, E. *Conhecendo o Servidor Apache (HTTP Server Project)*. maio 2006. Disponível em: <http://www.infowester.com/servapach.php>.

APACHE. *The Apache HTTP Server Project*. 2009. Disponível em: <http://httpd.apache.org>.

BURNETT, S.; PAINE, S. *Criptografia e segurança: o guia oficial*. [S.l.]: Editora Campus Ltda., 2002.

CHOWN, P. *Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)*. [S.l.], June 2002.

COSTA, L. S. d.; BONDAN, G. I. R. *Autoridade Certificadora Otimizadora*. outubro 2007.

COULOURIS, G. *Sistemas distribuídos: conceitos e projeto*. [S.l.]: Bookman, 2007.

CUSTÓDIO, R. F.; GRAAF, J. V.; DAHAB, R. *GT - ICP-EDU: Uma Infra-estrutura de Chaves Públicas para Âmbito Acadêmico*. [S.l.]: RNP, 2003.

GARFINKEL, S. *Web Security, Privacy & Commeerce*. [S.l.]: O'Really, 2002.

HOUSLEY, R. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. [S.l.], abril 2002.

JAGANATHAN, K.; ZHU, L.; BREZAK, J. *The RC4-HMAC Kerberos Encryption Types Used by Microsoft Windows*. [S.l.], December 2006.

KARN, P.; METZGER, P.; SIMPSON, W. *The ESP DES-CBC Transform*. [S.l.], August 1995.

KARN, P.; METZGER, P.; SIMPSON, W. *The ESP Triple DES Transform*. [S.l.], September 1995.

KELLERMANN, G. A.; SILVELLO, J. C. *Manual OpenLDAP*. 2001. Disponível em: <http://www.inf.ufrgs.br/gppd/disc/inf01008/trabalhos/sem01-1/t1/openldap/>.

MOD_SSL. *The Apache Interface to OpenSSL*. 2009. Disponível em: <http://www.modssl.org/>.

MYERS, M. *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. [S.l.], june 1999.

NAKAMURA, E. T. *Segurança de redes em ambientes corporativos*. [S.l.]: Novatec Editora Ltda., 2007.

OPENCA. *OpenCA Guide for Versions 0.9.2+*. 2009. Disponível em: <http://www.openca.info/docs/guide/openca-guide.html>.

OPENCA. *OpenCA Research Labs - OpenCA PKI Project*. 2009. Disponível em: <http://www.openca.org/projects/openca/>.

OPENLDAP. *OpenLDAP*. 2009. Disponível em: <http://www.openldap.org/>.

OPENSSL. *The Open Source toolkit for SSL/TLS*. 2009. Disponível em: <http://www.openssl.org/>.

PERL. *The Perl Programming Language*. 2009. Disponível em: <http://perldoc.perl.org/perl.html>.

RIVEST, R. *A Description of the RC2(r) Encryption Algorithm*. [S.l.], March 1998.

SILVA, P. T.; CARVALHO, H.; TORRES, C. B. *Segurança dos Sistemas de Informação - Gestão Estratégica da Segurança Empresarial*. [S.l.]: CENTROATLANTICO.PT, 2003.

STALLINGS, W. *Cryptography and Network Security - Principles and Practices*. [S.l.]: Prentice Hall, 2003.

TEIWES, S.; HARTMANN, P.; KUENZI, D. *Use of the IDEA Encryption Algorithm in CMS*. [S.l.], Fabruary 2001.

THOMAS, S. A. *SSL and TLS Essentials - Securing the web*. [S.l.]: Wiley, 2000.

VALLE, O. T. *Linux, Básico, Gerência, Segurança e Monitoramento de Redes*. 2009.

WALL, L. *Perl*. 2009. Disponível em: <http://perldoc.perl.org/perl.html>.