UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Douglas Bayer Santos

Sistema Gerenciador de Certificados de Atributos X.509

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Bacharel em Ciência da Computação.

Hendri Nogueira
Orientador

Prof. Ricardo Felipe Custódio, Dr.
Co-Orientador

Florianópolis, Novembro de 2013

Sistema Gerenciador de Certificados de Atributos X.509

Douglas Bayer Santos

Este Trabalho de Conclusão de Curso foi julgado adequado para a obtenção do título de Bacharel em Ciência da Computação e aprovada em sua forma final pelo Departamento de Informática e Estatística da Universidade Federal de Santa Catarina.

Prof. Vitório Bruno Mazzola, Dr.

Coordenador do Curso

Banca Examinadora

Hendri Nogueira

Prof. Ricardo Felipe Custódio, Dr.

Lucas Ferraro

Gustavo Banegas

"O insucesso é apenas uma oportunidade para recomeçar de novo com mais inteligência."
Henry Ford

Dedico este trabalho à minha família, cujo apoio e educação me permitiram chegar onde estou hoje. Dedico também aos meus amigos e namorada que sempre me apoiaram e incentivaram.

Agradecimentos

Agradeço ao professor Ricardo Felipe Custódio e a toda a equipe do LabSEC, que tornaram possível a realização deste trabalho. Em especial ao meu orientador Hendri, pela ajuda e paciência, e ao gerente do projeto, Gustavo, pelo companheirismo. E todos os outros que me acompanharam e ensinaram nesses três anos aqui no LabSEC.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas

CA	Certificado de Atributos
CD	Certificado Digital de Chave Pública
AC	Autoridade Certificadora
ICP	Infraestrutura de Chaves Públicas
XML	eXtensible Markup Language
AA	Autoridade de Atributo
EEA	Entidade Emissora de Atributo
CAV	Certificado de Atributo Vinculado
CAA	Certificado de Atributo Autônomo
FA	Fonte de Autoridade
IGP	Infraestrutura de Gestão de Privilégios
LCR	Lista de Certificados Revogados
LCAR	Lista de Certificados de Atributo Revogados
AR	Autoridade de Registro
PERMIS	PrivilegE and Role Management Infrastucture Standards validation
RBAC	Role Based Access Control
DAC	Discretionary Access Control
MAC	Mandatory Access Control
RFC	Request for Comments
LDAP	Lightweight Directory Access Protocol
ACL	Access Control List
OCSP	Online Certificate Status Protocol
UFSC	Universidade Federal de Santa Catarina
LabSEC	Laboratório de Segurança em Computação
ITI	Instituto Nacional de Tecnologia da Informação
ICP-Brasil	Infraestrutura de Chaves Públicas Brasileira
OID	Object Identifier
SGCA	Sistema Gerenciador de Certificados de Atributos

Resumo

O presente trabalho dedicou-se a implementação de um sistema para gerenciar certificados de atributos X.509. O trabalho tem como enfoque os certificados de atributos conforme o perfil brasileiro definido no conjunto normativo DOC-ICP-16 do Instituto Nacional de Tecnologia da Informação (ITI) e na *Request for Comments* (RFC) 5755, na qual o anterior é baseado. Ao longo do trabalho são apresentados alguns modelos de controle de acesso, a descrição de uma infraestrutura de gestão de privilégios X.509 conforme o padrão X.509 do ITU-T, o padrão brasileiro adotado para os certificados de atributos, alguns sistemas com controle de acesso baseado em certificados de atributos e a descrição do sistema desenvolvido, além de sua interface e operação.

Palavras chave: Sistema de Gerência, Certificados de Atributos, SGCA, Autorização, Controle de Acesso.

Abstract

The present work was dedicated to implement a system for managing X.509 attributes certificates. This work has a focus on the brazilian attribute certificate profile defined by the normative set DOC-ICP-16 of the Instituto Nacional de Tecnologia da Informação (ITI) and the standard on wich it is based upon the Request for Comments (RFC) 5755. Through this work will be presented a few access control models, the description of the X.509 privilege management infrastructure as of the ITU-T X.509 standard, the brazilian stardard adopted for attributes certificates, a few systems with access control based upon attributes certificates and the description of the developed system, its interface and operation.

Key words: Management System, Attribute Certificates, Authorization, Access Control.

Capítulo 1

Introdução

Com as inovações tecnológicas da sociedade moderna, as pessoas dependem cada vez mais de meios eletrônicos e online para a realização de tarefas cotidianas. Podemos citar como exemplo o ato de fazer compras através da internet ao invés de ir pessoalmente ao supermercado, ou a utilização de uma porta eletrônica com controle de acesso através de cartões eletromagnéticos ao invés de um porteiro. É possível reduzir o tempo gasto em diversas tarefas por parte dos usuários destes serviços e também redução do custo em manter uma infraestrutura física e alocação de pessoal por parte das empresas. Por outro lado, esta ausência de interação humana gera a necessidade de mecanismos de autenticação e autorização seguros para estes sistemas.

Dentre os mecanismos de autenticação e autorização seguros estão aqueles que fazem uso da certificação digital, com o Certificado Digital de chave pública (CD) para autenticação e o Certificado de Atributos (CA) para autorização.

1.1 Contextualização

Muitos documentos e processos não são feitos de forma digital devido aos custos envolvidos nesta mudança. Documentos ou informações onde os períodos de validade sejam curtos ou sofram constantes alterações não são adequados para constarem em um CD. Cada vez que alguma informação constante no CD seja alterada ou deixe de ser valida é necessário revogar e emitir o CD novamente e o alto custo operacional envolvido torna esta possibilidade inviável.

Devido ao custo e às limitações envolvidas na utilização dos certificados de chave

pública, estes não são viáveis para utilização como mecanismos de autorização e controle de acesso. Sendo assim, a Infraestrutura de Chaves Públicas Brasileira [dTdI a], responsável pela certificação digital no Brasil, optou por adotar o uso dos certificados de atributos em sua infraestrutura.

O certificado de atributos foi pensado especificamente para suprir esta deficiência da certificação digital em relação a função de autorização e controle de acesso. É possível utilizar CAs com baixo custo operacional e em diversos cenários, por exemplo, sistemas de autorização online, emissão de certidões. As informações que podem constar em um CA são as mais variadas possíveis, não havendo nenhum tipo de restrição aos atributos.

1.2 Objetivos

1.2.1 Gerais

Este trabalho tem como objetivo a realização do estudo sobre certificados de atributos e o desenvolvimento de um sistema de gerência de certificado de atributos compatível com os perfis da RFC 5755 [FAR 10] e DOC-ICP-16 do ITI [dTdI 12].

1.2.2 Específicos

Os objetivos específicos deste trabalho são:

- Descrever a infraestrutura para utilização dos certificados de atributos conforme o *framework* do padrão X.509 [UNI 08];
- Emitir e verificar certificados de atributos em conformidade com DOC-ICP-16;
- Revogar certificados de atributos;
- Implementar suporte de atributos em formato interoperável XML;
- Integrar o sistema com base LDAP com finalidade de publicar os certificados de atributos;

1.3 Motivação

O Laboratório de Segurança em Computação (LabSEC) localizado na Universidade Federal de Santa Catarina (UFSC), local onde o presente trabalho foi realizado, vem desen-

volvendo diversos trabalhos na área de certificação digital e suas aplicações em parceria com o Instituto Nacional de Tecnologia da Informação (ITI), órgão responsável pela Infraestrutura de Chaves Públicas Brasileira. Entretanto, os diferentes sistemas desenvolvidos até hoje no LabSEC não fazem uso de certificados de atributos. Os estudos desta tecnologia foram iniciados recentemente com a aprovação da utilização dos mesmos na ICP-Brasil, com objetivo de desenvolver um sistema de gerenciamento de certificados de atributos para o ITI. Através das atividades desenvolvidas pelo autor dentro do LabSEC, surgiu a oportunidade de estudar a informação presente na literatura e implementar um modelo de uso.

1.4 Metodologia

Para alcançar os objetivos do presente trabalho foi necessário uma ambientação do autor com os conceitos envolvidos. Foram estudados as normas que descrevem a estrutura e conceitos sobre certificação digital, infraestrutura de chaves públicas, ICP-Brasil, infraestrutura de gestão de privilégios e modelos de controle de acesso.

No decorrer do estudo, foi realizado um levantamento de referências do estado da arte na área e dos sistemas encontrados que utilizam certificados de atributos para autorização.

Após este levantamento, foi necessário buscar uma biblioteca criptográfica com suporte à certificação digital, de modo a auxiliar a implementação dos certificados de atributos. A biblioteca foi estudada e testada. Por último, o sistema foi implementado e testado.

1.5 Limitações do Trabalho

No presente trabalho não serão abordadas questões de privacidade envolvendo um CA, nem da criação e manutenção de políticas de controle de acesso e políticas de atributos. Serão abordados somente os aspectos básicos necessários à implementação do modelo e à aplicação, com objetivo de facilitar a gerência.

Embora existam outros modelos de infraestrutura de chaves públicas e certificados digitais, o escopo deste trabalho restringe-se somente ao padrão X.509. A implementação do sistema fica restrita ao uso dos certificados de atributos, conforme o conjunto normativo DOC-ICP-16 [dTdI 12].

Capítulo 2

Fundamentação Teórica

2.1 Modelos de Controle de Acesso

A função do controle de acesso é limitar as ações ou operações que um usuário legítimo do sistema computacional pode realizar [SAN 94]. O controle de acesso restringe o que um usuário pode fazer direta ou indiretamente, e.g., restringe o que programas atuando em nome de usuários estão autorizados a realizar. Desta maneira, modelos de controle de acesso buscam prevenir atividades que possam levar a uma falha de segurança. Um modelo de controle de acesso é ilustrado na figura 2.1.

Em sistemas distribuídos, de grande porte ou complexidade, podem existir mais de um modelo de controle de acesso sendo utilizados. As políticas de controle de acesso não são necessariamente exclusivas, políticas diferentes podem ser combinadas para se adequar às necessidades do sistema. Essa combinação é possível desde que não haja conflito entre as políticas, e.g., uma política garantir determinado tipo de acesso enquanto uma outra proíbe este mesmo acesso. Caso exista conflito, é necessário mudanças nas políticas para eliminar este conflito de forma a permitir o uso conjunto das políticas de acesso.

Nos casos onde mais de um modelo estão sendo utilizados, o acesso a um recurso somente é concedido se for possível estabelecer uma relação entre os níveis de segurança associados, ou seja, somente a interseção das políticas de acesso é válida.

2.1.1 Controle de Acesso Discricionário

Controle de Acesso Discricionário, ou do inglês *Discretionary Access Control* (DAC), é um modelo de controle de acesso no qual todo usuário, ou um programa atuando em seu nome,

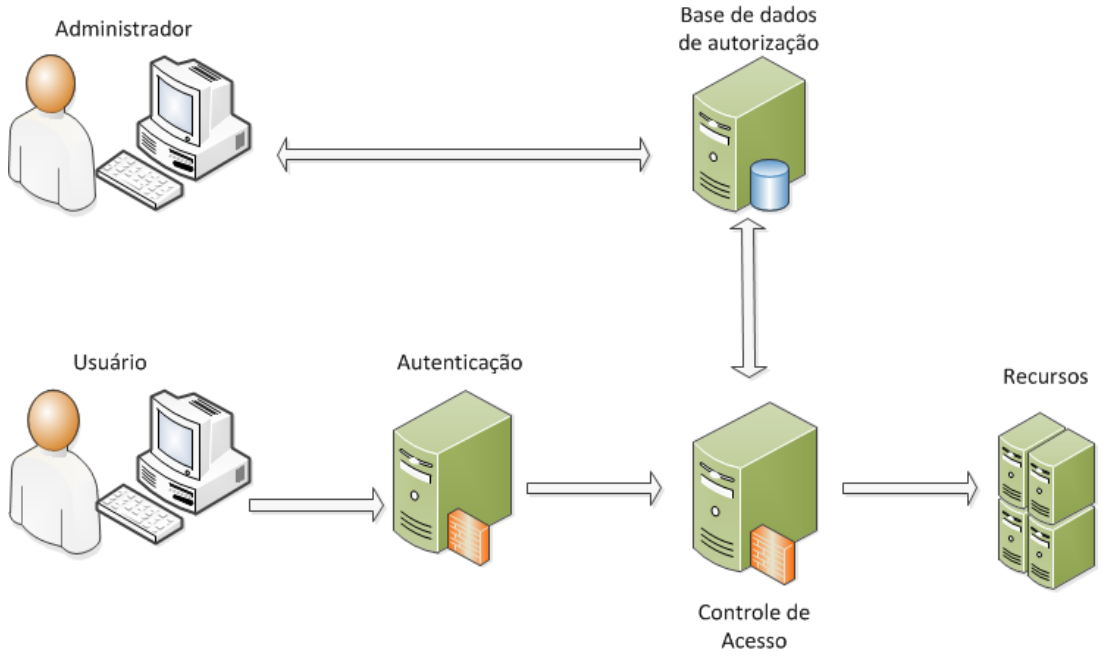

Figura 2.1: Controle de acesso e segurança.

tem permissão de especificar o tipo de acesso que outros usuários possuem sobre recursos ou informações dos quais é detentor [SAM 01]. Este modelo especifica que, sendo o usuário o dono do recurso, este pode ditar os direitos que os outros usuários podem exercer sobre este recurso.

As políticas discricionárias governam o acesso dos usuários à informação com base em sua identidade e autorização, que especifica quais tipos de acesso do usuário são permitidos a cada objeto do sistema [SAN 94].

Políticas discricionárias fazem valer o controle de autorização com base na identidade do usuário e um conjunto explícito de regras que estabelecem quem pode e como pode acessar os recursos. Exemplos de uso deste modelo são o sistema de arquivos do Linux onde as permissões são definidas para usuário e grupo nos modos de acesso *r - read, rw - read/write, x - execute*, e o Windows que utiliza uma lista de controle de acesso discricionária.

2.1.2 Controle de Acesso Mandatório

O Controle de Acesso Mandatório, ou do inglês *Mandatory Access Control* (MAC), restringe todos os acessos no sistema, conforme uma política de acesso baseada em regulamentos mandatórios determinados por uma autoridade central [SAM 01]. Os usuários que são

detentores de objetos no sistema não podem realizar quaisquer alterações das permissões dos objetos que sejam contrárias à política de acesso.

A forma mais comum de política mandatória é a segurança multinível, que se baseia em classificar todos os usuários e objetos do sistema, rotulando-os com um nível de segurança [OSB 97]. São os rótulos quem definem os diferentes níveis de segurança a serem aplicados (e.g., confidencial, público), podem haver variados rótulos com diferentes níveis de segurança no sistema. A política de acesso faz um controle estrito e unidirecional do fluxo de informações com base nos rótulos de segurança [OSB 97].

Dado um sistema, os objetos são entidades passivas do sistema que contêm dados, os usuários são entidades ativas que realizam pedidos de acesso aos objetos. Os usuários têm seus pedidos processados pela política de acesso, que define quais usuários têm acesso a determinado objeto do sistema, com base no nível de segurança do objeto e o nível de segurança do usuário em questão. A política de acesso é centralizada, definida pelo administrador do sistema, os usuários não podem ir contra a política. Um usuário não pode dar permissões a outros usuários, mesmo que este tenha total controle sobre o objeto, ele não pode delegar nenhuma permissão que seja contra a política de acesso.

Apesar de MAC fazer um ótimo trabalho protegendo dados e recursos sensíveis, sua restritividade não permite que satisfaça todos os requisitos de segurança necessários em um determinado sistema. Isto pode tornar o modelo inadequado dependendo dos requisitos do sistema desejado.

2.1.3 Lista de Controle de Acesso

Lista de Controle de Acesso, em inglês *Access Control List* (ACL), é uma lista ordenada de controle de acesso, com entradas que definem permissões aplicadas a um objeto [MIC 05]. Cada entrada identifica um usuário do sistema e define um conjunto específico de regras de acesso que o usuário pode utilizar no objeto do sistema. Quando um acesso a um objeto do sistema é realizado, este deve primeiro percorrer toda a ACL até que o usuário requisitante seja encontrado. Em seguida, devem ser identificadas as permissões de acesso a ser repassadas para o usuário. Usuários que não estão explicitamente na ACL, ou não pertencem a um dos grupos identificados na ACL, têm sua permissão negada.

Uma ACL é tipicamente uma forma de controle de acesso que se assemelha ao mandatório. Apesar de não ser necessariamente uma única lista exclusiva no sistema, as permissões

válidas são somente as listadas nas entradas da ACL. Mas existe também outras formas de implementação de ACL que fogem a essa semelhança do mandatório, como a lista de controle de acesso discricionária (DACL) que é uma ACL onde os usuários têm permissão de delegar um subconjunto de suas permissões a outros usuários. A DACL é tipicamente controlada pelo dono do objeto ou qualquer outro usuário que tenha recebido permissão de alterar as permissões do objeto.

2.1.4 Controle de Acesso Baseado em Papéis

O modelo de Controle de Acesso Baseado em Papéis, ou em inglês *Role-Based Access Control* (RBAC), começou a ser discutido na década de 1990 e rapidamente ganhou popularidade. O RBAC emergiu como tecnologia comprovada para gerenciar e controlar a segurança de sistemas de larga escala em grandes organizações [OSB 07]. Isto foi motivado pelo fato de que os modelos tradicionais de controle de acesso não eram muito adequados a sistemas com muitos usuários e dados.

Os modelos tradicionais de controle de acesso DAC e MAC apresentam algumas deficiências quando lidando com sistemas complexos, com centenas de usuários e milhares de dados [OSB 07]. No modelo DAC, as permissões são dadas aos usuários diretamente, esta abordagem gera uma desvantagem em um sistema de grande porte que é a quantidade de tempo necessária e a dificuldade de gerenciar permissões de acesso para cada objeto do sistema individualmente, para cada usuário do sistema. Por outro lado, o modelo MAC é muito rígido, requer que a todo objeto e usuário do sistema seja atribuído um rótulo de segurança, além da definição de regras para os níveis de segurança. O modelo MAC é para aplicações onde é necessário um controle do fluxo de dados e manter sigilo dos dados, é muito difícil utilizar um modelo tão restrito de modo comercial.

O objetivo principal do modelo RBAC é prover um modelo capaz de auxiliar o administrador do sistema a gerenciar o controle de acesso em um ambiente complexo com muitos usuários e dados [OSB 07]. A noção central do RBAC é que as permissões são associadas a papéis e aos usuários são designados os papéis [OSB 07]. Com RBAC as decisões de acesso são baseadas nos papéis que os usuários têm na organização e não somente na sua identidade.

O modelo RBAC fornece um valioso nível de abstração para melhorar a segurança na administração do sistema, ao tratar do problema a nível da organização e não somente a nível de identidade do usuário [CHA 03a]. O conceito dos papéis é de estabelecer permissões com

base no papel funcional na organização e então de maneira apropriada atribuir os usuários a papéis ou conjunto de papéis [SAN 96].

Os papéis dentro de uma organização são relativamente persistentes, o modelo RBAC fornece um forte mecanismo para reduzir a complexidade, custo e o potencial erro ao atribuir permissões de usuários na organização [SAN 00]. Como os papéis dentro da organização muitas vezes têm permissões em comum, modelos RBAC costumam estabelecer uma hierarquia entre os papéis e permitir que um papel adquira as permissões dos papéis abaixo na hierarquia.

RBAC permite maior flexibilidade na especificação e utilização de políticas de controle de acesso, que podem ser ajustadas de organização para organização. Como as permissões são organizadas a partir dos papéis funcionais da organização, fica mais evidente conflitos entre as relações do que se estivesse lidando com as permissões individualmente para cada usuário.

2.2 Criptografia

Criptografia significa escondido, escrita secreta. Criptografia geralmente é pensado como o ato de embaralhar uma mensagem secreta de modo que somente seu destinatário saiba como desembaralhar e ler a mensagem original [HOU 01].

Técnicas de criptografia modernas, além de manter a confidencialidade da mensagem, podem ser utilizadas para verificar a integridade e a autenticidade da mensagem [HOU 01].

2.2.1 Resumo Criptográfico

Uma função de resumo criptográfico, ou função de *hash*, recebe um fluxo de dados e reduz estes dados a um tamanho fixo através de uma função matemática não inversível [HOU 01]. O resumo criptográfico tem um tamanho fixo. Funções de *hash* diferentes podem possuir resumos de tamanhos diferentes, mas a mesma função sempre resulta em um mesmo tamanho.

O resumo criptográfico é sempre o mesmo para um mesmo conjunto de dados. Com uso de uma boa função de *hash* é virtualmente impossível encontrar dois conjuntos de dados que resultem em um mesmo resumo [HOU 01].

As propriedades do resumo criptográfico permitem realizar verificações de integridade sobre os dados. Comparando dois resumos criptográficos de uma mesma mensagem, obtidos com o mesmo algoritmo em diferentes períodos de tempo, é possível determinar se a

mensagem se manteve íntegra durante todo este período. Se a comparação determinar que os resumos são idênticos então não houveram quaisquer alterações na mensagem, caso contrário houveram alterações ou perdas nos dados.

2.2.2 Criptografia Assimétrica

Criptografia de chaves assimétricas, também conhecida como criptografia de chaves públicas, é uma classe de algoritmos de criptografia na qual um par de chaves é criado, onde tudo que é cifrado com uma chave pode apenas ser decifrado com a outra chave correspondente [HOU 01].

Dentre o par de chaves geradas uma é a chave pública, que pode ser disponibilizada sem restrições, e a outra chave a privada, que deve ser armazenada em total segurança [DIF 76]. Dados cifrados com a chave privada são decifrados por qualquer pessoa com a chave pública correspondente, dados cifrados com a chave pública são decifrados unicamente pelo detentor da chave privada.

Através destas propriedades é possível enviar mensagens confidenciais ao detentor da chave privada e verificar a autenticidade das mensagens enviadas por este. No entanto, para autenticar uma mensagem é preciso saber quem possui a chave privada e qual a chave pública correspondente. O certificado digital de chave pública, explicado na seção 2.3.1), é utilizado para identificar o detentor da chave privada correspondente a chave pública que consta no certificado. O certificado é um documento eletrônico emitido por uma entidade confiável que coloca sua assinatura digital no certificado, garantindo sua autenticidade e integridade.

Em criptografia simétrica, onde a mesma chave é usada para cifrar e decifrar, existe um grande problema em como armazenar as chaves utilizadas em segurança e como enviar uma chave em segurança para que o destinatário possa decifrar a mensagem. Com o uso da criptografia assimétrica é possível evitar a complexidade associada ao armazenamento e distribuição seguros de chaves simétricas [HOU 01]. Em contrapartida, a criptografia assimétrica é mais custosa e lenta que a simétrica, em termos de processamento, sendo inadequada para cifrar grandes quantidades de dados.

2.2.3 Assinatura Digital

O processo de assinatura digital é baseado na criptografia assimétrica, as suas propriedades permitem assinar uma mensagem e verificar esta assinatura [HOU 01]. A assinatura

é feita cifrando o texto com a chave privada, deste modo é possível autenticar a assinatura utilizando a chave pública.

A criptografia assimétrica apresenta problemas em cifrar grandes quantidades de dados, tornando o processo inviável. Como consequência, a assinatura de documentos e mensagens grandes é também inviável. Como assinar documentos e mensagens grandes é uma necessidade em aplicações reais, o processo de assinatura na realidade precisa fazer mais operações do que cifrar o texto com a chave privada.

Uma simplificação do processo de assinatura real consiste de computar o resumo criptográfico da mensagem a ser assinada, reduzindo assim a quantidade de dados a serem cifrados. O resumo criptográfico é cifrado com a chave privada do signatário. A utilização do resumo criptográfico permite também a verificação de integridade da mensagem. A figura 2.2 ilustra o processo de assinatura simplificado.

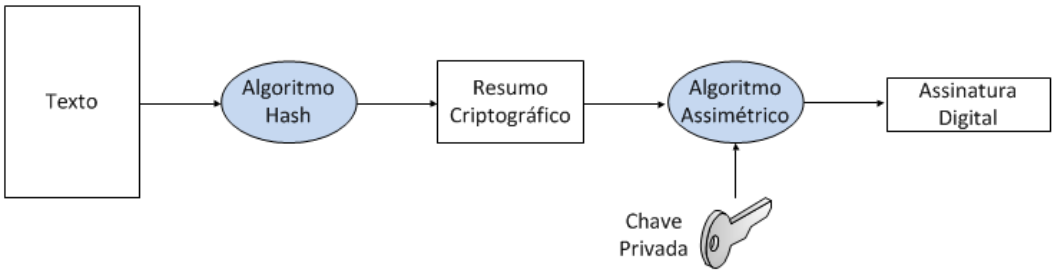

Figura 2.2: Esquema simplificado de assinatura digital.

A verificação de uma assinatura digital é feita através do uso da chave pública correspondente para decifrar o texto contido na assinatura. Através disso, somente o dono da chave privada poderia ter cifrado a mensagem, garantindo a autenticidade da assinatura. Computando o resumo criptográfico da mensagem assinada e comparando com o resumo contido na assinatura é possível verificar a integridade da mensagem.

2.3 Infraestrutura de Chaves Públicas

É a infraestrutura criada para o gerenciamento de chaves públicas e, consequentemente, fornecer suporte aos serviços de autenticação, cifragem, integridade e não-repúdio [UNI 08]. Infraestrutura de Chaves Públicas (ICP) é o conjunto formado por todos os recursos humanos, físicos, virtuais, e procedimentais necessários para gerenciar certificados digitais de

chave pública (CD) [HOU 01].

Uma ICP é baseada em um sistema de confiança, duas entidades que confiam em uma terceira fazem uso desta para verificar e confirmar a identidade uma da outra. Esta terceira parte confiável é chamada de âncora de confiança. A estrutura de uma ICP é uma hierarquia de autoridades, parecida com uma estrutura de dados em árvore, um nodo inicial que deriva os nodos filhos. Um nodo da árvore, no caso, seria uma Autoridade Certificadora (AC), sendo a autoridade certificadora inicial da ICP, a âncora de confiança, chamada de AC-Raiz.

A AC-Raiz é a autoridade máxima na estrutura da ICP, porém ela não lida diretamente com os usuários finais da ICP. Existem ACs chamadas de intermediárias e finais, as intermediárias têm seus certificados emitidos pela AC-Raiz e emitem os certificados para as ACs finais, que lidam diretamente com os usuários da ICP e emitem seus certificados.

Um sistema utilizando CDs precisa validar o CD antes que possa ser utilizado com segurança [HOU 01]. O período de validade deve estar dentro do prazo que consta no próprio certificado e, se por algum motivo, a AC revogou o certificado antes deste expirar, essa informação é divulgada na Lista de Certificados Revogados (LCR). Para validar um certificado digital de uma entidade é preciso conhecer a sua AC emissora. Conhecendo a AC que emitiu o certificado é possível verificar sua assinatura no certificado utilizando a chave pública da AC. Este procedimento se repete nos certificados das ACs até a AC-Raiz da infraestrutura.

Os componentes básicos de uma ICP são descritos na tabela 2.1.

Tabela 2.1: Componentes básicos de uma ICP.

Componente	**Descrição**
Autoridade Certificadora(AC)	Entidade confiável que emite os certificados
Autoridade de Registro (AR)	Entidade confiável que verifica a autenticidade das informações de uma outra entidade
Lista de Certificados Revogados (LCR)	Lista dos certificados revogados antes do seu período de expiração
Certificado Digital de Chave Pública (CD)	Traz informações sobre o titular juntamente com sua chave pública

2.3.1 Certificado Digital de Chave Pública

O Certificado Digital de Chave Pública (CD) é composto pela chave pública do usuário, juntamente com outras informações, e autenticado pela assinatura digital com a chave privada da AC emissora [UNI 08]. Um certificado digital de chave pública é um objeto inteiramente digital.

O certificado digital de chave pública contém campos para o nome do usuário e sua chave pública. O certificado também pode incluir informação de contato do titular, e.g., endereço de correio eletrônico. O certificado ainda inclui dois campos com datas que especificam respectivamente data de ativação e data de expiração do certificado. Também contém o nome da entidade confiável que emitiu o certificado. O emissor inclui no certificado um número de série único para cada certificado que cria, para identificar cada um claramente. Por fim, todo este conjunto de informações do certificado é validado pela assinatura digital do emissor [HOU 01].

2.3.2 Autoridade Certificadora

Autoridade Certificadora (AC) é uma autoridade de confiança de um ou mais usuários para criar e assinar certificados digitais de chave pública [UNI 08]. A autoridade certificadora é a unidade básica da ICP. Ela é formada pelo conjunto de hardware, software e pessoal que a operam. A AC é conhecida por dois atributos, seu nome e sua chave pública [HOU 01].

A AC desempenha quatro funções básicas da ICP:

- Emitir certificados, isto é, criar e assinar os certificados.
- Manter informação de status do certificado e emitir listas de certificados revogados.
- Publicar os certificados correntes, não expirados, e suas LCRs, para que os usuários possam obter a informação que precisam para validação.
- Manter informações sobre status de certificados expirados ou revogados que foram emitidos pela AC.

A AC coloca seu nome em cada certificado emitido por ela, e assina os certificados com sua chave privada. Assim os usuários que confiam na AC, direta ou indiretamente (através do caminho de certificação), podem identificar os certificados emitidos pela AC com seu nome

e reconhecendo sua assinatura utilizando a chave pública da AC. Com isto, os usuários têm certeza de que o certificado é genuíno e foi de fato emitido pela AC em que confiam.

Todo certificado emitido por uma AC é de sua responsabilidade, como toda a informação contida no certificado. Para verificar as informações do certificado, a AC conta com a Autoridade de Registro (AR). A informação sobre o status de revogação dos certificados pode ser provida utilizando o Protocolo Online de Status de Certificados, ou do inglês *Online Certificate Status Protocol* (OCSP), lista de certificados revogados (LCR), ou outro mecanismo. Quando o meio utilizado é a LCR, que normalmente é emitida pela própria AC, a AC pode delegar tal função à outra entidade [COO 08].

2.3.3 Autoridade de Registro

Uma Autoridade de Registro (AR) é uma entidade designada a verificar informações do certificado para a AC [HOU 01]. De forma similar à AC, uma AR é todo o conjunto de hardware, software e pessoal que a opera. Contudo, diferentemente da AC, muitas vezes uma única pessoa opera uma AR. Cada AC mantém uma lista de ARs credenciadas que são consideradas de confiança. Uma AR é conhecida pela AC por seu nome e chave pública [HOU 01].

Existem dois modelos básicos de verificação das informações do certificado feita pela AR. No primeiro modelo, a AR coleta e verifica as informações necessárias da entidade fazendo o pedido do certificado antes mesmo do pedido ser enviado à AC. A AC confia na informação do pedido porque já foi verificado pela AR.

No segundo modelo, a AC envia para a AR as informações de um pedido que ela já tenha recebido. A AR então revisa as informações do pedido e determina se as informações sobre o usuário estão corretas, enviando uma resposta de “sim” ou “não” para a AC [HOU 01].

O primeiro modelo é utilizado quando o usuário vai fisicamente até a AR, neste caso a identidade do usuário pode ser verificada por meios convencionais, e.g., documento de identidade, carteira de motorista. O segundo modelo é utilizado quando o usuário não pode ser identificado previamente, e gera o pedido do certificado diretamente para a AC.

2.3.4 Revogação

Nesta seção são apresentados os métodos utilizados para divulgação do status de revogação dos certificados.

2.3.4.1 Lista de Certificados Revogados

A Lista de Certificados Revogados (LCR), é uma lista indicando que um conjunto de certificados não é mais considerado válido por seu emissor [UNI 08]. Como nos certificados, as informações contidas na LCR devem ser corretas. A informação contida na LCR, e.g., identificação dos certificados, data em que o certificado foi revogado, razão pela qual o certificado foi revogado, deve ser o mais precisa possível. Um erro ou a omissão da informação poderia ter a consequência da aceitação de um certificado inválido [HOU 01]. Caso seja emitida uma lista contendo um certificado confiável válido ou data de revogação incorreta, pode causar com que um usuário rejeite um certificado confiável resultando em uma negação do serviço.

A LCR gerada por uma AC somente é útil se estiver disponível para os usuários [HOU 01]. Se a AC atende uma comunidade não restrita de usuários, a distribuição da LCR é focada em disponibilidade e performance e não em segurança. Não há necessidade de proteger o acesso à LCR pois a informação contida nela não é sigilosa. A AC pode, no entanto, restringir o acesso a uma comunidade restrita de usuários.

Com um grande número de usuários e constantes revogações de certificados as LCRs podem tornar-se excessivamente grandes, causando impacto no desempenho para sua obtenção e armazenamento. No entanto, os usuários podem armazenar cópias da LCR localmente. Neste caso, não há necessidade de conexão com a internet para realizar consultas constantes, somente quando uma nova LCR é publicada.

2.3.4.2 Protocolo Online de Status de Certificados

Uma alternativa viável para complementar ou mesmo substituir as LCRs é fazendo uso do Protocolo Online de Status de Certificados, ou do inglês *Online Certificate Status Protocol* (OCSP) [MYE 99]. Com a utilização do OCSP, é possível uma aplicação realizar consultas a um serviço que verifica o estado de um determinado certificado, mantido pela AC.

A resposta enviada pela AC é assinada digitalmente, a fim de garantir sua confiabilidade, e pode conter os seguintes valores: válido, revogado, desconhecido. Válido indica uma resposta positiva da integridade do certificado, no mínimo fornecendo indicação que o certificado não foi revogado. Deve ser levado em conta a data em que a resposta foi produzida com o período de validade do certificado. Revogado indica que o certificado foi revogado, permanentemente ou temporariamente. Desconhecido indica que o serviço de consulta não sabe sobre o determinado certificado em questão, ou que este certificado não foi emitido por esta AC.

Uma consulta OCSP contém muito menos informações que uma LCR, sendo mais eficiente na utilização da rede e recursos do cliente. Como o cliente não precisa fazer o parse da LCR em busca do certificado desejado, ele economiza o tempo de processamento. No entanto, uma conexão com a internet é sempre necessária. O cliente precisa estar sempre *online* para realizar as consultas, no caso de perda de conexão, o cliente não tem como consultar o status de nenhum certificado.

Capítulo 3

Certificados de Atributos X.509

Neste capítulo é apresentado a infraestrutura para utilização dos certificados de atributos de acordo com o *framework* do X.509, padronizado pelo ITU-T, e o perfil adotado pela ICP-Brasil como padrão brasileiro para certificados de atributos.

3.1 Infraestrutura de Gestão de Privilégios

A Infraestrutura de Gestão de Privilégios (IGP) é a infraestrutura capaz de gerenciar privilégios de usuários e sistemas, operando em conjunto a uma infraestrutura de chaves públicas [UNI 08]. A IGP oferece suporte a um serviço de autorização para complementar a autenticação fornecida por uma ICP [CHA 03a].

De maneira geral, privilégios de uma entidade não têm duração tão longa quanto a do período de validade de um certificado de chave pública, e devido ao custo envolvido na criação e gerenciamento de um CD não é viável utilizar estes certificados para conterem as informações de privilégios. A IGP faz uso de outra estrutura de dados para esta função. A principal estrutura de dados de uma IGP é o Certificado de Atributos (CA). Enquanto um CD mantém uma ligação forte entre usuário e sua chave pública, o certificado de atributos mantém uma ligação forte entre usuário e um ou mais atributos de privilégio [CHA 03a].

Como na ICP, a IGP também é baseada em um sistema de confiança, onde duas partes fazem uso de uma terceira parte confiável para fornecer informações de privilégios uma da outra. A estrutura de IGP é uma hierarquia de autoridades similar a da ICP, mas no lugar das ACs a IGP tem Autoridades de Atributo (AA). Sendo a autoridade de atributo inicial, similar à âncora de confiança da ICP, chamada de Fonte de Autoridade (FA). A figura 3.1 ilustra uma

IGP.

Antes de aceitar um CA como válido, um verificador precisar verificar a validade do mesmo. Esta verificação do CA ocorre de modo similar a do CD na ICP, para verificar a validade de um CA o verificador precisa validar todas as assinaturas do caminho de delegação até a FA.

Os componentes básicos da IGP conforme o *framework* X.509 são encontrados na tabela 3.1 e descritos a seguir;

Tabela 3.1: Componentes básicos de uma IGP.

Componente	**Descrição**
Fonte de Autoridade (FA)	Entidade raiz confiável para emissão do atributo
Autoridade de Atributo (AA)	Entidade que tem poder de conceder um atributo
Lista de Certificados de Atributo Revogados (LCAR)	Lista dos certificados revogados antes do seu período de expiração
Certificado de Atributo (CA)	Apresenta qualidades associadas ao seu titular

3.1.1 Fonte de Autoridade

A fonte de autoridade também é uma autoridade de atributo, mas na hierarquia da IGP ela é a raiz de confiança. A FA é uma autoridade de atributo, na qual o verificador de privilégio de um recurso em particular confia como sendo a autoridade máxima para designar um conjunto de privilégios [CHA 03b].

É a FA quem delega seus privilégios a outras autoridades de atributo para que estas possam assinar certificados de atributos, por sua vez conferindo privilégios às entidades fazendo uso dos certificados. Um verificador de CA conhece e confia na FA, quando este verifica um CA ele valida todas as assinaturas no caminho de delegação até encontrar a assinatura da FA.

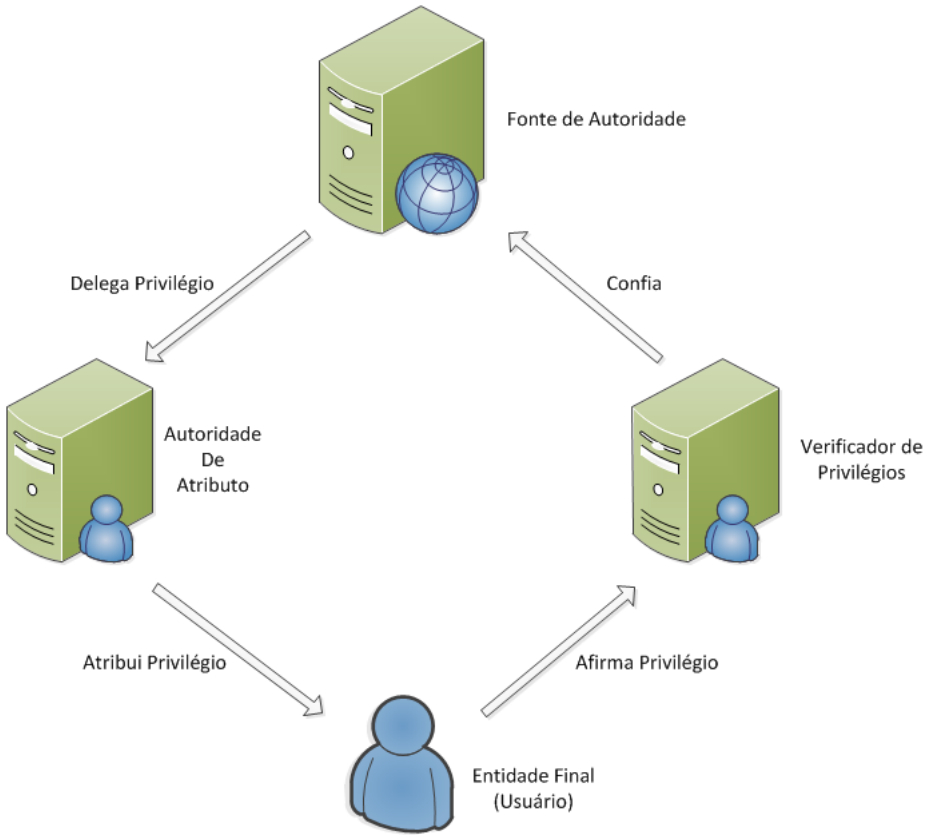

Figura 3.1: Estrutura de uma IGP.

3.1.2 Autoridade de Atributo

A Autoridade de Atributo (AA) é uma autoridade que associa atributos à uma entidade através da emissão de certificados de atributos [UNI 08].

A hierarquia da IGP começa com a fonte de autoridade. A partir dela, os privilégios são delegados às demais entidades da infraestrutura, que podem ser usuários ou autoridades de atributo. Se a entidade a quem FA delega pode emitir certificados de atributos, ela é chamada de AA [CHA 04]. Uma AA pode delegar seus privilégios para outras AAs ou usuários. No entanto, uma AA pode ou não ser capaz de criar novos privilégios.

Uma AA é a autoridade responsável por emitir, gerenciar e revogar certificados de atributos. É a AA quem assina o CA concedendo os atributos listados no certificado ao seu titular. A responsabilidade dos atributos listados no certificado é da AA, é ela quem garante a veracidade da informação por todo o período de validade do certificado. Também é responsabilidade da AA revogar um certificado caso um de seus atributos deixe de ser válido, isso deve ser feito através da Lista de Certificados de Atributos Revogados (LCAR), OCSP ou outro meio

pré-determinado.

3.1.3 Lista de Certificados de Atributos Revogados

Uma lista de revogação contendo referências à certificados de atributos que não são mais considerados válidos por sua autoridade emissora, é chamada de Lista de Certificados de Atributos Revogados (LCAR) [UNI 08]. A lista de certificados de atributo revogados é similar à lista de certificados revogados da infraestrutura de chaves públicas.

Contudo, na estrutura da IGP não é necessário que exista a LCAR, ela é opcional devido ao fato que os atributos contidos nos certificados de atributos podem ter períodos de validade muito curtos. O que resultaria em problemas no desempenho de uso e armazenamento da LCAR. Nestes casos, dependendo da política adotada, a publicação do status de revogação pode ser ignorada ou adotada de outras formas, como por exemplo, fazendo uso do Protocolo Online de Status de Certificados (OCSP).

3.1.4 Estrutura de um Certificado de Atributos X.509

O certificado de atributo faz a ligação do *holder*, o seu titular, com os atributos (informações de privilégios), sem ter o alto custo operacional de um CD. É uma estrutura de dados digital, que contém um conjunto de atributos de uma entidade e algumas informações de identificação, assinada digitalmente com a chave privada da AA [UNI 08].

```
AttributeCertificateInfo ::= SEQUENCE {
  version                 AttCertVersion, -- version is v2
  holder                  Holder,
  issuer                  AttCertIssuer,
  signature               AlgorithmIdentifier,
  serialNumber            CertificateSerialNumber,
  attrCertValidityPeriod  AttCertValidityPeriod,
  attributes              SEQUENCE OF Attribute,
  issuerUniqueID          UniqueIdentifier OPTIONAL,
  extensions              Extensions OPTIONAL
}
```

Figura 3.2: Estrutura ASN.1 do CA. Fonte [FAR 10]

Um CA é composto por três partes, a que contém todas as informações relevantes, chamada AttributeCertificateInfo conforme a figura 3.2. As outras duas partes, para o algo-

ritmo de assinatura e a assinatura propriamente dita. A estrutura do AttributeCertificateInfo é detalhada a seguir;

- *Version*: Especifica a versão utilizada;
- *Holder*: Identifica o titular do CA;
- *Issuer*: Identifica o emissor do CA;
- *Signature*: Identifica o algoritmo de assinatura;
- *SerialNumber*: Número de série único para cada AA;
- *AttCertValidityPeriod*: O período durante o qual o certificado é válido;
- *Attributes*: Um conjunto não vazio de atributos formados por: OID e valor;
- *IssuerUniqueID*: Opcional caso a AA tenha um identificador único;
- *Extensions*: Extensões opcionais para CA.

Todas essas informações devem estar presentes no certificado, com exceção das opcionais. O titular e o emissor podem ser identificados no CA através das seguintes opções:

- *BaseCertificateID*: Através do seu certificado digital de chave pública, o campo vai conter a informação sobre qual a autoridade certificadora que emitiu o certificado e seu número de série. Sendo assim possível identificar unicamente o certificado;
- *EntityName*: Um nome utilizado para identificar a entidade;
- *ObjectDigestInfo*: Resumo criptográfico da identificação da entidade, do seu CD por exemplo, desta forma sua autenticação tem que ser feita diretamente;

É recomendado que somente uma dessas opções seja utilizada para evitar ambiguidades. No perfil da RFC 5755 somente a opção *baseCertificateID* deve ser utilizada para identificar o emissor.

Caso seja utilizado *baseCertificateID*, a AA precisa verificar a validade do certificado a ser utilizado antes de assinar o CA, uma vez que toda informação contida no CA é de responsabilidade da AA.

3.1.4.1 Atributos

Os atributos do certificado são informações de privilégios referentes ao titular do CA. Por exemplo, dentro da estrutura empresarial o atributo presidente indica que uma pessoa tem livre acesso a todos os setores e informações e outro atributo gerente indica que pode realizar contratações. Outro exemplo, a Ordem dos Advogados do Brasil pode emitir um certificado com um atributo que indica que o seu titular é um advogado habilitado e registrado, possibilitando assim, que ele tenha as plenas condições de exercer sua função.

A estrutura dos atributos como definida na RFC 5755 pode ser vista na figura 3.3.

```
Attribute ::= SEQUENCE {
  type      AttributeType,
  values    SET OF AttributeValue
    -- at least one value is required
}

AttributeType ::= OBJECT IDENTIFIER

AttributeValue ::= ANY DEFINED BY AttributeType
```

Figura 3.3: Estrutura ASN.1 dos Atributos. Fonte [FAR 10]

Segue abaixo uma lista de tipos de atributos permitidos no perfil descrito pela RFC 5755:

- *Service Authentication Information*: Este atributo identifica o titular para algum serviço/servidor específico no sistema alvo, por exemplo, nome de usuário e senha, neste caso o conteúdo do atributo seria cifrado;
- *Access Identity*: Utilizado para prover informações sobre o titular ao verificador do CA, para autorizar ações do titular no sistema do verificador;
- *Charging Identity*: Este atributo identifica a identidade de cobrança, que normalmente é diferente do titular. Por exemplo, identifica a companhia do titular que pode ser cobrada por um serviço;
- *Group*: Carrega informação sobre grupos aos quais o titular pertence;
- *Role*: Indica papéis/funções que foram atribuídas ao titular;

- *Clearance*: Contém informação sobre liberação de acesso do titular, tem um campo indicando a qual política de segurança a informação diz respeito.

Cada tipo tem uma estrutura específica que pode ser encontrada na RFC 5755. Alguns tipos usam a mesma estrutura, a *IetfAttrSyntax*, ela permite separar a Autoridade de Política de Atributo da emissora do CA. A estrutura identifica a responsável pela política no próprio atributo.

3.1.4.2 Extensões

Uma extensão é um estrutura opcional, que permite incluir no certificado informações que não são suportadas pelo formato básico do certificado [HOU 01]. As seguintes extensões podem ser utilizadas em conformidade com o perfil da RFC 5755:

- *Audit Identity*: Utilizado em circunstâncias onde a identidade de um indivíduo não pode ser obtida diretamente através do rastreamento de operações, o auxílio da AA é necessário para sua identificação;

- *AC Targeting*: Faz um direcionamento do uso do CA para um grupo específico de servidores/serviços. Um verificador honesto que encontre esta extensão e não esteja presente na lista deve rejeitar o CA;

- *Authority Key Identifier*: Auxilia o verificador a conferir a assinatura da AA no CA. Não é necessária nos casos onde o campo *issuer* utiliza a opção *baseCertificateID*, onde a ligação com o certificado digital de chave pública é explicita;

- *Authority Information Access*: Usada para indicar uma URL HTTP com um serviço OCSP, com finalidade de facilitar a verificação do status de revogação do certificado;

- *CRL Distribution Points*: Esta extensão serve para indicar o local de publicação da lista de certificados revogados, deve conter uma URL HTTP ou LDAP;

- *No Revocation Available*: Indica que nenhuma informação sobre o status de revogação deste CA será publicada pela AA.

3.2 ICP-Brasil

A Infraestrutura de Chaves Públicas Brasileira (ICP-Brasil) é a cadeia de confiança responsável pela emissão de certificados digitais no Brasil. A ICP-Brasil é gerida pelo Instituto Nacional de Tecnologia da Informação (ITI), órgão federal responsável por manter e auditar a ICP, estimular e desenvolver o uso da tecnologia de certificação digital no Brasil [dTdI b].

3.2.1 Certificados de Atributos na ICP-Brasil

No dia 5 de Julho de 2012, o Comitê Gestor da Infraestrutura de Chaves Públicas Brasileira aprovou a criação dos certificados de atributos no âmbito da ICP-Brasil [dTdI a]. No modelo aprovado não é criada uma nova infraestrutura baseada em autoridades exclusivamente para a emissão de certificados de atributos. Os certificados podem ser emitidos por qualquer entidade com um certificado digital ICP-Brasil de pessoa jurídica do tipo A3 ou A4, após assinados os certificados passam a ter validade jurídica.

O modelo adotado tem como referências o *framework* X.509 [UNI 08] e principalmente a RFC 5755 [FAR 10], com algumas particularidades. A documentação oficial que regulamenta o uso dos certificados de atributos na ICP-Brasil é composta pelo conjunto normativo DOC-ICP-16.

3.2.1.1 Perfil Brasileiro para Certificados de Atributos

No perfil brasileiro de certificados de atributos, a entidade responsável pela emissão dos certificados de atributos se chama Entidade Emissora de Atributo (EEA). Esta faz o mesmo papel que uma autoridade de atributo, é a entidade responsável pela emissão de certificados de atributos, verificação e gestão dos atributos.

Para ser uma EEA, a entidade deve ser uma pessoa jurídica com prerrogativa legal para emissão de determinado atributo e possuir um certificado digital ICP-Brasil A3 ou A4 do tipo pessoa-jurídica. É também de responsabilidade da EEA padronizar e identificar unicamente os seus atributos através de um OID. Cabe a cada entidade emissora de atributo organizar sua hierarquia de atributos e publicá-la da melhor forma possível.

A vinculação do certificado com o titular pode se dar de duas formas, caracterizadas na ICP-Brasil como dois tipos de certificados:

- Certificado de Atributos Autônomo (CAA): Equivalente à ligação *entityName*, não é pre-

ciso um certificado digital neste caso. No entanto, é necessário que o certificado tenha alguma informação que o relacione ao titular (e.g. RG, CPF, CNPJ). A presença do titular do certificado não é necessária, o que é muito conveniente para instituições que emitem declarações e/ou certidões pela internet [dTdI 12].

- Certificado de Atributos Vinculado (CAV): Equivalente à ligação *baseCertificateID*, tem um vínculo direto com um certificado digital ICP-Brasil. Para emitir o certificado, a EEA deve ter acesso ao CD do titular e fazer todas as verificações necessárias. O uso do CD gera a necessidade de validar ambos os certificados, o de atributos e o digital, para aceitar o CAV como válido. Esta opção fornece maior segurança ao CA [dTdI 12].

Capítulo 4

Trabalhos Relacionados

Esta seção apresenta *softwares* relacionados com os certificados de atributos. Estes *softwares* permitem gerenciar os certificados de atributos.

4.1 PERMIS

PrivilEge and Role Management Infrastructure Standards (PERMIS), é uma infraestrutura que provê aos seus usuários todos os meios necessários para gerenciar privilégios e políticas de autorização [CHA 03b]. O projeto foi concebido na Europa, como parte de um esforço conjunto entre Espanha, Itália e Reino Unido, para desenvolver uma infraestrutura de gestão de privilégios X.509 com controle de acesso baseado em papéis. Um dos resultados deste esforço foi a criação da ferramenta PERMIS.

PERMIS tem suporte ao modelo RBAC básico e RBAC hierárquico, existe uma hierarquia entre os papéis, assim quando um papel inferior recebe um atributo todos os superiores recebem o mesmo atributo. A estrutura do PERMIS também permite a utilização somente de atributos no lugar de papéis, neste caso as permissões para os atributos devem constar na política de autorização de maneira análoga ao uso de papéis.

O PERMIS tem dois módulos para gerência, um para privilégios e outro para políticas. O gerenciador de privilégios é na verdade um gerenciador de certificados de atributos que permite ler, criar e designar os privilégios aos usuários e armazenar os certificados de atributos.

O gerenciador de políticas auxilia na construção e edição de políticas de autorização e conta inclusive com suporte à delegação de privilégios. As políticas são salvas em XML, uma vez criadas as políticas elas podem ser opcionalmente encapsuladas em um certificado de

atributo X.509 [CHA 08].

PERMIS por padrão utiliza uma base LDAP para armazenar as políticas e certificados, mas permite ser expandido para utilizar outros repositórios. Ele também pode ser usado para controle de acesso com diversas aplicações como Apache, Shibboleth e Microsoft .NET [1].

As ferramentas e módulos que compõem o PERMIS estão disponíveis para *download* e separadas em categorias, para facilitar a aprendizagem de forma gradual. A maioria dos módulos conta com documentação descrevendo sua instalação/utilização e alguns inclusive com aplicações de teste. Alguns contam ainda com interface gráfica, que é de grande auxílio na utilização de componentes como o editor de políticas. A utilização do PERMIS no início é um pouco confusa devido à quantidade de configurações possíveis. Os sistemas operacionais listados como suportados são Windows XP, 2000 e Linux. Apesar disto, nem todos os módulos funcionaram corretamente em ambos os sistemas. A integração com a base LDAP também apresentou problemas e não foi possível ser realizada.

A ferramenta é implementada na linguagem de programação Java e baseada na biblioteca IAIK, além de contar também com uma versão aberta chamada Open PERMIS. O PERMIS apresenta poucas modificações aos padrões estabelecidos no *framework* X.509 [UNI 08] e a RFC 5755 [FAR 10].

4.2 strongSwan

O strongSwan é uma implementação IPsec aberta para a plataforma Linux, que tem como foco do projeto, mecanismos de autenticação segura utilizando certificados digitais X.509 [STR 13]. O projeto é baseado no FreeS/WAN, o qual foi descontinuado e, apesar de mudar o foco para utilização de certificados X.509, o projeto mantém os ideais de seu antecessor de um software livre para comunicação segura através da rede.

A aplicação tem suporte para gerar chaves RSA/ECDSA e certificados X.509, o strongSwan oferece suporte para verificação de status dos certificados através de LCR ou OCSP. Ainda, os certificados podem ser armazenados em *smartcards* via PKCS#11. O projeto conta também com suporte a protocolos *Internet Key Exchange* e certificados de atributos X.509. Os certificados de atributos são gerados pela ferramenta auxiliar *openac*, somente atributos de grupo estão sendo utilizados para controle de acesso e a ligação com o titular é feita através do

[1] Fonte: http://sec.cs.kent.ac.uk/permis/index.shtml

seu certificado digital.

Como o foco da ferramenta é fornecer segurança para redes de computadores, os aspectos de autenticação segura e sigilo da informação são muito importantes. Devido a este foco, a parte de autorização e o uso dos certificados de atributos é secundária na ferramenta. Apesar de funcionais, os certificados de atributos são muito restritos na implementação do strongSwan.

4.3 OpenPMI

OpenPMI é um projeto iniciado por professores da Universidade de Magala na Espanha para gerência de certificados de atributos X.509. A ferramenta surgiu em uma tentativa de criar uma aplicação fiel ao *framework* X.509 do ITU-T [UNI 08], fazendo uso da biblioteca OpenSSL [MON 04]. Os autores se inspiraram no PERMIS para tentar criar uma aplicação mais fiel aos padrões para os certificados de atributos estabelecidos.

Com o uso do OpenSSL, foi implementada uma nova biblioteca utilizando as linguagens de programação C e C++ para suporte de certificados de atributos X.509. A ferramenta funciona como sistema de gerência da AA, permite criar e revogar certificados e tem suporte à delegação de privilégios, ainda conta também com capacidade de publicar os certificados em uma base LDAP. Os atributos também são geridos e podem ser alterados e sofrer restrições quanto à delegação. Para a delegação, o cliente deve se conectar à AA e se autenticar para então requisitar a operação desejada. A forma de autenticação padrão é usuário e senha, uma outra alternativa suportada é a utilização de certificados digitais.

Vários problemas foram enfrentados para compilar e instalar a aplicação, além de problemas de compatibilidade. A utilização do OpenPMI com o OpenSSL é feita toda através do terminal, o que dificulta a utilização por parte de um usuário que não tenha prática com o modo texto. O editor de delegação tenta facilitar seu uso com a utilização de uma interface gráfica e modelando a política com grafos, mas acaba tornando mais difícil o entendimento do usuário uma vez que não fornece informações sobre o que os valores no grafo representam e como eles afetam a política de delegação de atributos.

A última versão disponível tem suporte ao Apache para executar o servidor, sendo necessário compilar o código. A aplicação tem suporte aos sistemas operacionais Microsoft Windows XP, Windows Vista, Windows 7 x64 e x32.

Capítulo 5

Desenvolvimento

5.1 Tecnologias Utilizadas

Abaixo são apresentadas as ferramentas utilizadas para o desenvolvimento do sistema.

5.1.1 Java Enterprise Edition

O Java EE oferece uma plataforma robusta e segura com suporte a diversas APIS e a *web services*. Fazendo uso da linguagem de programação Java, a plataforma oferece interoperabilidade e portabilidade para a aplicação. A arquitetura Java EE facilita o desenvolvimento web, não exige definições de interface, pois a própria linguagem java já possui interface, eliminando as complexidades de mapeamento. A versão utilizada é a Java EE 7[1].

5.1.2 Eclipse IDE

Eclipse é um ambiente de desenvolvimento *open source* para Java, muito robusto e popular entre os desenvolvedores. Permite facilidade ao desenvolvedor para editar, depurar e refatorar o código. Foi utilizado uma versão do Eclipse específica para Java EE de codinome Kepler[2].

[1]Disponível em: http://www.oracle.com/technetwork/java/javaee/downloads/index.html

[2]Disponível em: http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/keplersr1

5.1.3 Bouncy Castle API

The Legion of Bouncy Castle é uma iniciativa *open source* para desenvolvimento de um provedor de criptografia compatível com o Java Cryptography Architecture (JCA). A API do Bouncy Castle conta com vários pacotes essenciais ao desenvolvimento do trabalho como: provedor de criptografia JCA, biblioteca de leitura e escrita ASN.1, geradores de certificados X.509, CRLs e arquivos PKCS#12. A versão utilizada do Bouncy Castle é a 1.47[3].

5.1.4 Apache Tomcat

Apache é um servidor web *open source* que utiliza as tecnologias Java Servlet e JavaServer Pages, próprio para o uso com aplicações Java EE. O Apache Tomcat 7[4] foi a versão utilizada do servidor web.

5.1.5 Subversion

O Subversion (SVN) é um sistema de controle de versão de código aberto. Facilita o controle das versões do projeto e mantém uma cópia atualizada do projeto em um servidor, como um backup, podendo também obter a última cópia do projeto em qualquer computador com acesso à internet.

5.1.6 TRAC

TRAC é uma ferramenta *open source* de interface web para monitoramento de mudanças em projetos de desenvolvimento de *software*. Ele oferece controle de mudanças através de *tickets*, *timeline* do projeto, wiki para documentação e integração com o subversion.

5.1.7 ASN.1

Abstract Syntax Notation One (ASN.1) é uma notação formal para representação, codificação e transmissão de dados. Através de suas regras de codificação, o ASN.1 facilita a troca de informações entre aplicações por fornecer uma representação independente de linguagem de programação ou de máquina.

[3]Disponível em: http://www.bouncycastle.org/java.html

[4]Disponível em: http://tomcat.apache.org/download-70.cgi

5.1.8 Dumpasn1

Para visualizar os certificados no formato ASN.1 foi utilizada a ferramenta *dumpasn1*. Essa ferramenta possibilita codificar dados em qualquer formatação ASN.1 para uma forma textual mais inteligível a humanos.

5.1.9 XML

A Extensible Markup Language (XML) é um formato simples de texto para representar informações estruturadas. É um dos formatos mais utilizados atualmente para compartilhamento de informações, entre pessoas ou entre computadores.

5.2 Sistema Gerenciador de Certificados de Atributos

Nesta seção são apresentados os requisitos do sistema e diagramas de casos de uso, classes, modelagem do banco de dados e em seguida apresentada a interface e operação da aplicação que foi desenvolvida, o Sistema Gerenciador de Certificados de Atributos (SGCA). Serão apresentadas as duas versões do SGCA, Standalone e Web, sendo a versão Web apresentada em mais detalhes.

5.2.1 Requisistos

Nesta seção são apresentados os requisitos funcionais e não-funcionais definidos para o sistema.

5.2.1.1 Requisistos Funcionais

Os requisitos funcionais definem as funções fundamentais que devem existir no sistema.

RF01 - Emissão de Certificado de Atributos
O sistema deve ser capaz de emitir certificados de atributos, conforme definido na RFC 5755 e nas especificações do DOC-ICP-16.

RF02 - Conexão com LDAP

O sistema deve ter uma função que permita, caso selecionada, colocar os certificados de atributos emitidos em uma base LDAP.

RF03 - Atributos

Deve ser possível criar e editar atributos para utilização no sistema, para a utilização dos atributos o usuário seleciona o atributo com OID desejado e adiciona seu valor.

RF04 - Templates

O sistema deve disponibilizar uma opção para criar modelos contendo um ou mais OIDs de atributos, uma vez selecionado um modelo todos os atributos de OIDs correspondentes devem ser inclusos no certificado.

RF05 - Visualizar Certificado Antes da Emissão

Antes da emissão de um certificado ser completada o sistema deve exibir todos os dados a constarem no certificado para serem conferidos pelo usuário.

RF06 - Disponibilização do Certificado de Atributo Emitido

O sistema deve disponibilizar um certificado emitido através de um *download* direto ou LDAP (RF02).

RF07 - Controle de Acesso

O sistema deve contar com uma forma de controle de acesso, onde existem dois tipos de usuários: Administrador e Operador. O administrador pode efetuar todas as operações do sistema. O operador fica restrito a emitir certificados de atributos, emitir listas de certificados revogados e revogar certificados.

5.2.1.2 Requisitos Não-Funcionais

Os requisitos não-funcionais definem aspectos estruturais do sistema.

RNF01 - Linguagem de Programação Java

Deve ser utilizado para o desenvolvimento do sistema a linguagem de programação Java.

RNF02 - Utilização da Biblioteca Bouncy Castle

O uso desta biblioteca é necessário por ser uma biblioteca de código livre que possui suporte completo para criar certificados de atributos.

5.2.2 Diagramas

Todos os diagramas referentes ao sistema podem ser encontrados no Anexo C.

5.2.2.1 Diagrama de Casos de Uso

O sistema tem dois tipos de usuários que podem acessar o sistema, Administradores e Operadores. O diagrama de casos de uso da figura 7.1 demonstra as funcionalidades que devem estar disponíveis no sistema para um Administrador exercer sua função. Um Administrador no sistema tem livre acesso a todas as funcionalidades disponíveis no sistema. A figura 7.2 contém o diagrama de casos de uso para um usuário do tipo Operador. Um Operador no sistema tem acesso restrito a emissão e revogação de certificados e emissão de LCR.

5.2.2.2 Diagrama de Classes

O diagrama de classes, disponível na figura 7.3, abstrai algumas partes do sistema contendo apenas as principais classes para ser mais compreensível. A interface do programa se comunica com a classe *AppController* que realiza todas as chamadas para as funções necessárias para realizar as operações. A classe *LdapHandler* utiliza a biblioteca UnboundID LDAP para efetuar a comunicação com a base LDAP. A classe *CertificateAttribute* representa um atributo, com oid, valor e descrição. Os dados de um certificado de atributos são armazenados em um objeto da classe *CertificateDataObject*, que contém todos os campos do certificado e também o certificado em codificação DER. *PersistenceDatabase* é a classe encarregada das operações com banco de dados, para salvar e recuperar informações do banco.

5.2.2.3 Diagrama Entidade Relacional Estendido

O diagrama contendo a modelagem do banco de dados pode ser visto na figura 7.4. Os certificados de atributos emitidos ficam armazenados na tabela *Certificates*, uma vez revogado um certificado ele é adicionado a tabela *Revoked_Certificates*. As configurações da base LDAP ficam armazenadas na tabela *Ldap_Configuration*. A tabela *Auth_User* armazena os

dados dos usuários do sistema. Os atributos e templates são armazenados respectivamente nas tabelas *Attributes* e *Templates* existindo uma relação entre um template e os atributos contidos neste.

5.2.3 SGCA Standalone

Em um primeiro momento do desenvolvimento, a aplicação foi iniciada puramente com uso da linguagem Java e utilizando do seu pacote gráfico swing. Esta primeira versão serviu para experimentação e aprendizagem da API do Bouncy Castle. A aplicação resultante desta versão tem as funcionalidades básicas para emitir um certificado de atributos e criar atributos, estas funcionalidades são melhor descritas na próxima seção.

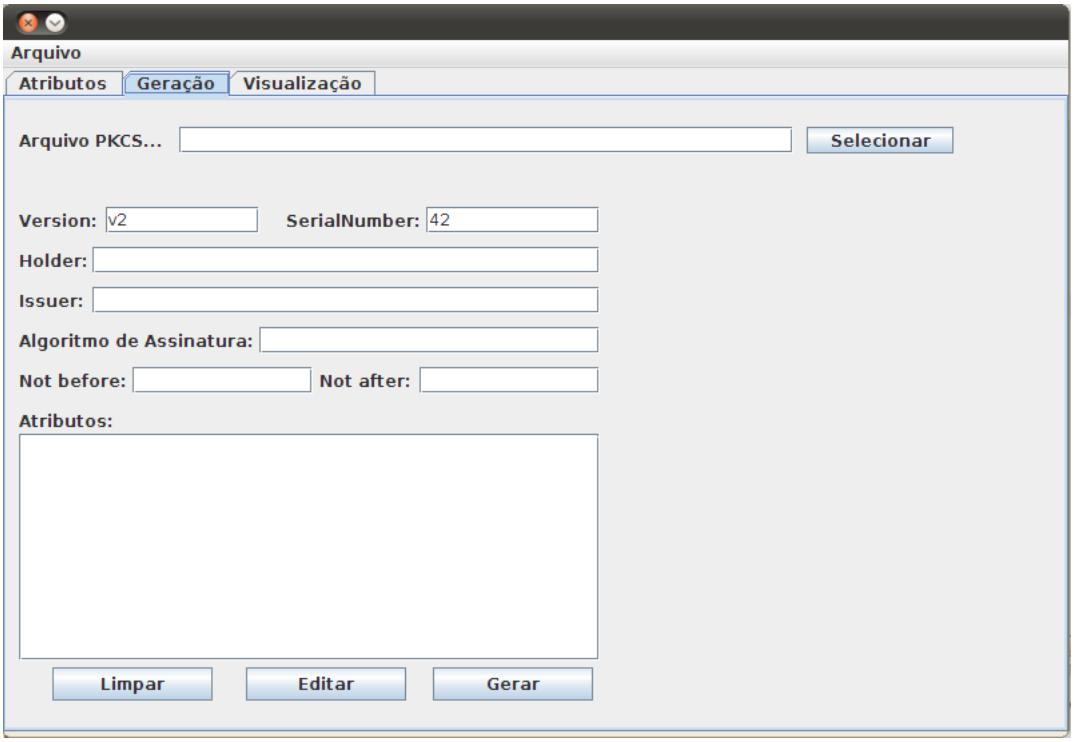

Figura 5.1: Primeira versão em Java swing.

5.2.4 SGCA Web

A segunda versão desenvolvida foi na forma de uma aplicação web. Os benefícios da utilização web são a possibilidade da aplicação poder ser utilizada apenas localmente ou disponível para intranet ou internet e ter acesso independentemente do sistema operacional utilizado. Esta versão conta com uma interface mais amigável para navegação e pode emitir

certificados de atributos no padrão ICP-Brasil, com suporte a extensões, atributos e templates, além de contar também com integração com banco de dados SQL e base LDAP. O SGCA foi instalado em um Ubuntu Server 12.04 com SQL Server e utilizando o servidor web Apache Tomcat. O acesso ao sistema como usuário foi feito utilizando os navegadores Firefox e Google Chrome.

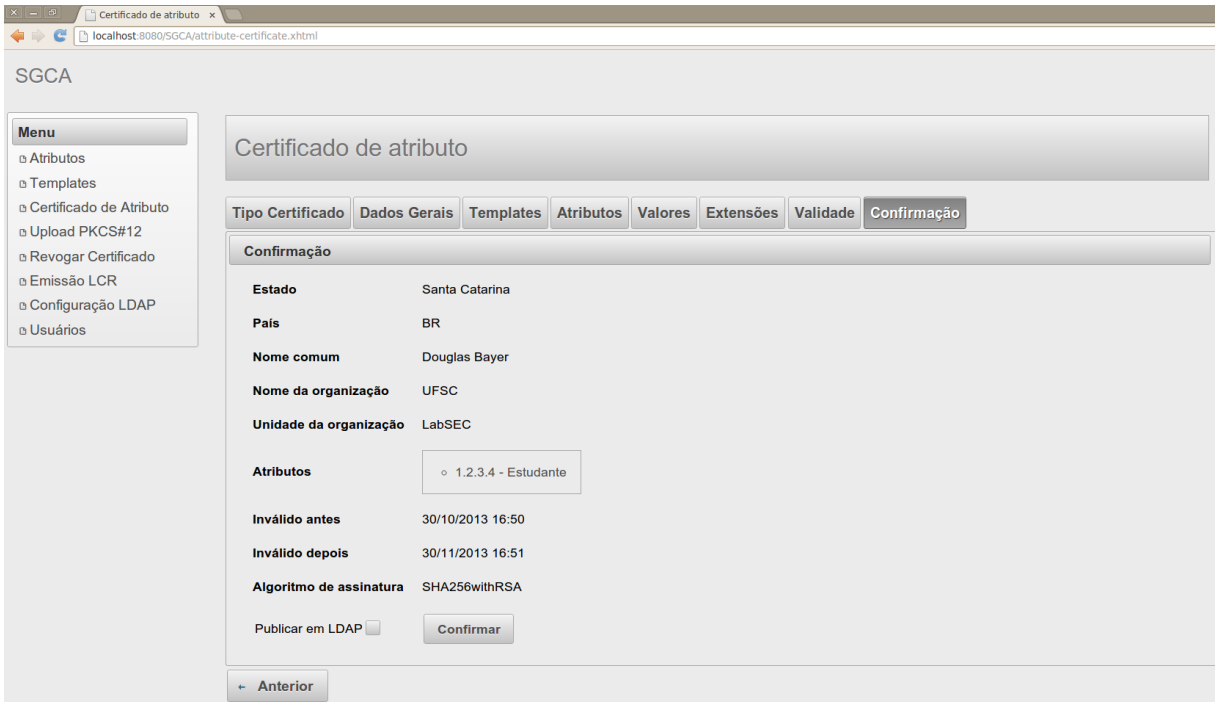

Figura 5.2: Visão do SGCA através do *browser* Google Chrome.

A seguir são descritas as funções que fazem parte da navegação do SGCA e demonstrações de seus componentes e suas interfaces gráficas.

5.2.5 Login

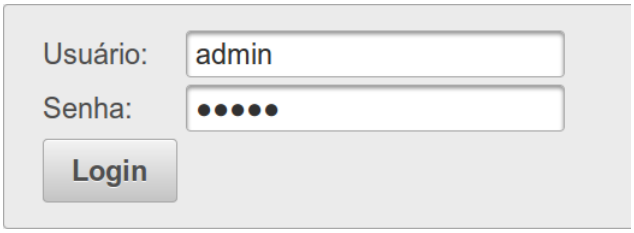

Figura 5.3: Tela de Login.

Sendo uma aplicação web, são necessários mecanismos de segurança para garantir que pessoas indevidas não tenham acesso à aplicação ou quaisquer informações trocadas com a mesma. Ao acessar a aplicação, a primeira coisa com que o usuário tem contato é a tela de login.

O login é obrigatório antes de acessar qualquer parte da aplicação, caso um usuário tente acessar alguma página que não seja a página de login, este é redirecionado automaticamente para a página de login. A página de login conta ainda com proteção contra tipos comuns de ataque como *Cross-site scripting* (XSS) e *SQL Injection*. A aplicação utiliza *Secure Sockets Layer* (SSL) para manter uma conexão criptografada com o usuário, mantendo sigilo de todas as informações trocadas.

Os usuários do sistema e suas funções são melhor explicados na seção 5.2.14.

5.2.6 Menu

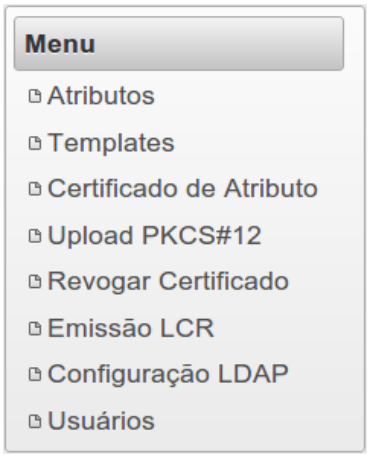

Figura 5.4: Menu de navegação.

O menu de navegação do SGCA encontra-se na parte mais à esquerda do navegador do usuário, figura 5.4. Este menu fica sempre visível e localizado nesta posição. Contém todas as referências para as seções de configurações e emissão de CA. As referências no menu são exibidas com base na função do usuário logado, as funções de usuário são melhor explicadas na seção 5.2.14. Os itens do menu são descritos a seguir.

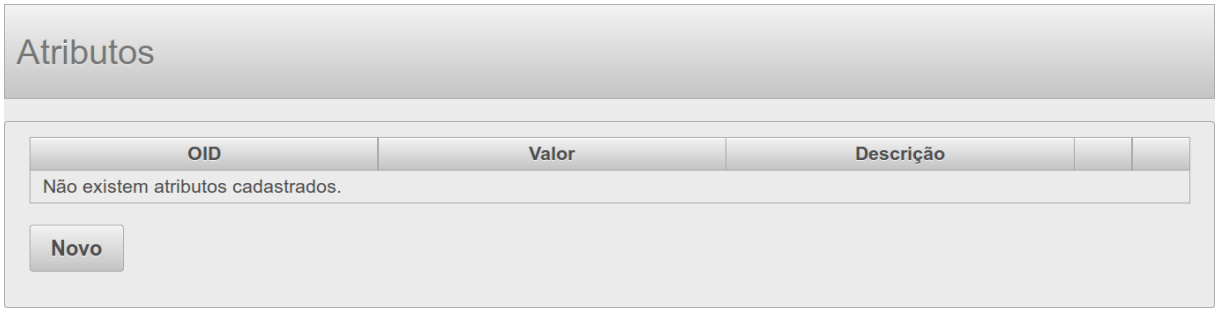

Figura 5.5: Seção de atributos quando nenhum atributo está cadastrado no sistema.

5.2.7 Atributos

Na seção de atributos são exibidos todos os atributos cadastrados e permite criar novos atributos.

A figura 5.5 demonstra o estado da tela de atributos quando nenhum atributo está cadastrado no sistema. Ao clicar no botão “Novo”, a tela para criação de atributos é exibida.

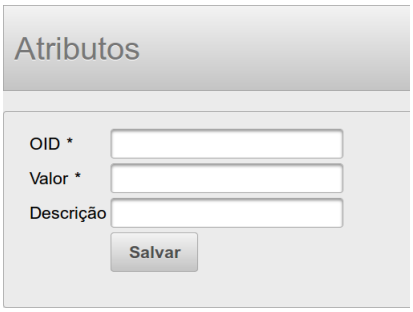

Figura 5.6: Tela de configuração de um novo atributo.

Um atributo é composto por um OID, um valor e uma descrição. O campo *OID* é o mais importante, pois identifica o atributo e determina como é utilizado. O preenchimento deste campo é obrigatório. O campo *valor* serve para conter uma descrição de como o real valor do atributo deve ser informado durante o processo de emissão, e.g., data nascimento no formato DD/MM/AAAA. O preenchimento deste campo é obrigatório.

O campo *descrição* serve para facilitar o reconhecimento do atributo pelo usuário, pode conter um nome do atributo ou descrição de casos de uso do mesmo, o uso deste campo é opcional.

Uma vez configurados novos atributos, estes são agora exibidos na tela de atributos, como pode ser visto na figura 5.7. Cada atributo pode ser modificado ou removido, através dos

Figura 5.7: Seção de atributos exibindo os atributos presentes no sistema.

botões editar e remover respectivamente.

O botão remover apaga o registro deste atributo da base de dados completamente. O botão editar permite alterar os campos valor e descrição dos atributos, mas não seu OID, conforme figura 5.8.

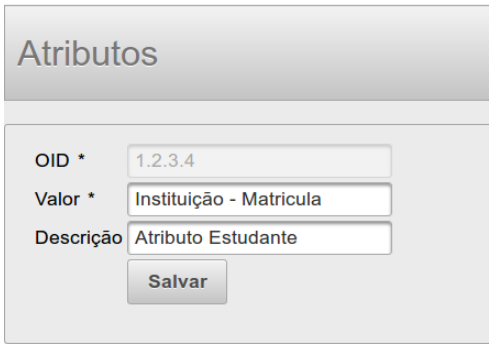

Figura 5.8: Tela de edição de atributos.

Conforme o DOC-ICP-16, uma entidade emissora de atributos deve publicar seus atributos da melhor forma possível. No SGCA, a publicação dos atributos é feita no formato XML, que é uma linguagem que permite fácil interpretação de dados. O arquivo é composto por uma lista de atributos que contém todos os atributos juntamente com seus respectivos OIDs, valores e descrições 5.9.

5.2.8 Templates

Os templates são uma adição do SGCA e eles não constam na documentação do X.509 ou DOC-ICP. Templates são conjuntos de atributos com intuito de facilitar sua seleção, eliminando a necessidade de buscar cada atributo individualmente.

```
−<attributes>
   <attribute oid="1.1.1.1" value="numero CRM" desc="Atributo Medico"/>
   <attribute oid="1.2.3.4" value="Instituicao - Matricula" desc="Atributo Estudante"/>
   <attribute oid="1.2.4.3" value="numero CPF" desc="Atributo CPF"/>
 </attributes>
```

Figura 5.9: Estrutura XML da lista de atributos.

Isto facilita na criação de papéis ou funções que tenham vários atributos, e.g., um médico possui as características CRM, RG, CPF, então um template "Médico" conteria estes mesmos atributos. Adicionando o template "Médico", o certificado do médico tem todos os atributos referentes automaticamente incluídos.

A tela de templates, ilustrada pela figura 5.10, é similar a dos atributos, mas listando os templates cadastrados no sistema. Cada template mostra o seu nome e o OID de cada atributo que contém. Os templates também podem ser modificados ou excluídos através dos botões gráficos.

Figura 5.10: Seção de templates exibindo os templates presentes no sistema.

O botão "Novo" exibe a tela para criação de novos templates, esta sendo bastante diferenciada da tela de criação de atributos, demonstrado pela figura 5.11. Um template é identificado unicamente por seu nome e pode receber um conjunto de quaisquer atributos já cadastrados no sistema.

5.2.9 PKCS#12

Esta seção realiza o envio de um arquivo PKCS#12, figura 5.12, o qual é utilizado para assinar os certificados de atributos. O certificado contido deve ser válido para assinatura e de acordo com DOC-ICP-16, ou seja, ser do tipo Pessoa-Jurídica A3 ou A4. Após selecionar o arquivo, devem ser fornecidas as senhas do backup e da chave. Uma vez enviado o arquivo,

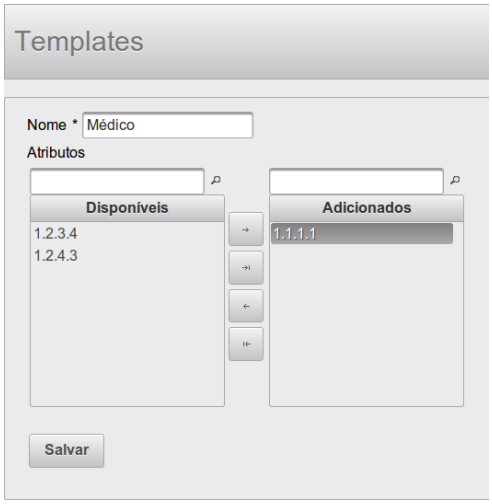

Figura 5.11: Tela de configuração de um novo template.

uma mensagem de êxito será exibida para o usuário.

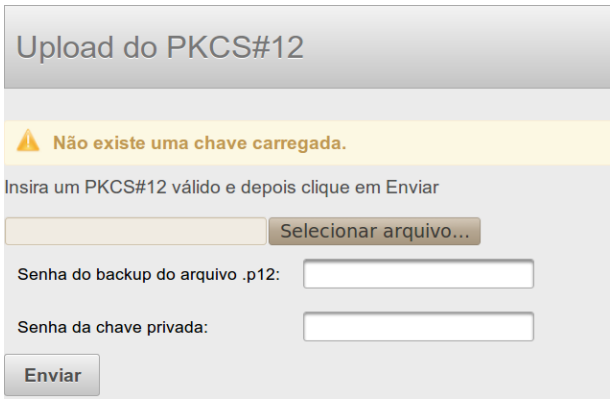

Figura 5.12: Seção de envio do arquivo PKCS#12.

5.2.10 LDAP

A seção de configuração do LDAP, ilustrada na figura 5.13, permite que o administrador do sistema configure um diretório LDAP para salvar os CAs emitidos.

O endereço do servidor deve ser fornecido na forma de uma URL http ou https, seguido por ":" número da porta "/". O campo de usuário deve ser preenchido com o *Distinguished Name* (DN) do mesmo, mas ignorando seu *Domain Component* (DC). O DC do diretório é informado em outro campo, onde o SGCA monta o DN quando necessário acesso ao LDAP. A senha é fornecida em dois campos que ocultam os caracteres inseridos e confirmam

se ambos são iguais.

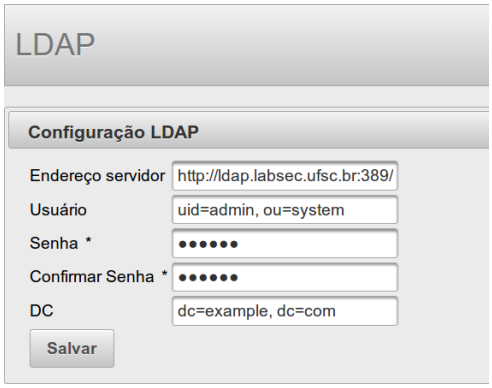

Figura 5.13: Seção de configuração do diretório LDAP.

5.2.11 Certificado de Atributos

Nesta seção é apresentado o processo de emissão de um certificado de atributos no SGCA. A emissão do certificado de atributos é realizada através de diversas telas representando diferentes etapas do processo de emissão, estas etapas são descritas nas próximas seções. Um usuário pode navegar entre estas etapas com o uso dos botões “Anterior” e “Próximo”, sendo possível avançar somente se a etapa atual estiver em conformidade com todos os requisitos. O processo de emissão consiste nas seguintes etapas:

5.2.11.1 Tipo Certificado

Em conformidade com o padrão ICP-Brasil para certificados de atributos definido no DOC-ICP-16, o CA deve pertencer a um dos seguintes tipos: Certificado de Atributos Vinculado (CAV) ou Certificado de Atributos Autônomo (CAA). Somente uma opção deve ser utilizada, caso nenhuma ou as duas sejam marcadas o sistema exibe uma mensagem de erro, indicando que somente uma deve ser utilizada. A tela de escolha do tipo de certificado é ilustrada na figura 5.14.

5.2.11.2 Dados Gerais

Para emissão de um certificado vinculado, o seu titular deve possuir um certificado digital ICP-Brasil válido. Conforme demonstra a figura 5.15, o certificado deve ser fornecido

Figura 5.14: Escolha do tipo de certificado, autônomo ou vinculado.

ao SGCA e este então realiza uma verificação da sua validade. O usuário não pode avançar para a próxima etapa até fornecer o certificado requerido.

Figura 5.15: Envio do certificado digital para vinculação ao de atributos.

Caso a opção escolhida anteriormente seja do Certificado de Atributos Autônomo, a tela de dados gerais muda de forma e assume novos campos, conforme figura 5.16. A emissão do certificado autônomo não requer nenhum mecanismo de autenticação forte como o certificado digital do CA vinculado. Para emitir um CAA, a entidade emissora deve fornecer os seguintes dados sobre o futuro titular do CA: estado de origem; país de origem; nome comum; organização; unidade organizacional. O campo de identificador único (*Unique Identifier*) é opcional.

Estas informações podem não ser o suficiente para a identificação de um indivíduo, o DOC-ICP-16 recomenda adicionar outras informações de identificação como RG, CPF.

5.2.11.3 Templates

Nesta aba são apresentados os templates atualmente disponíveis no sistema, visível na figura 5.17. O usuário pode avançar para a próxima etapa sem selecionar nenhum template ou

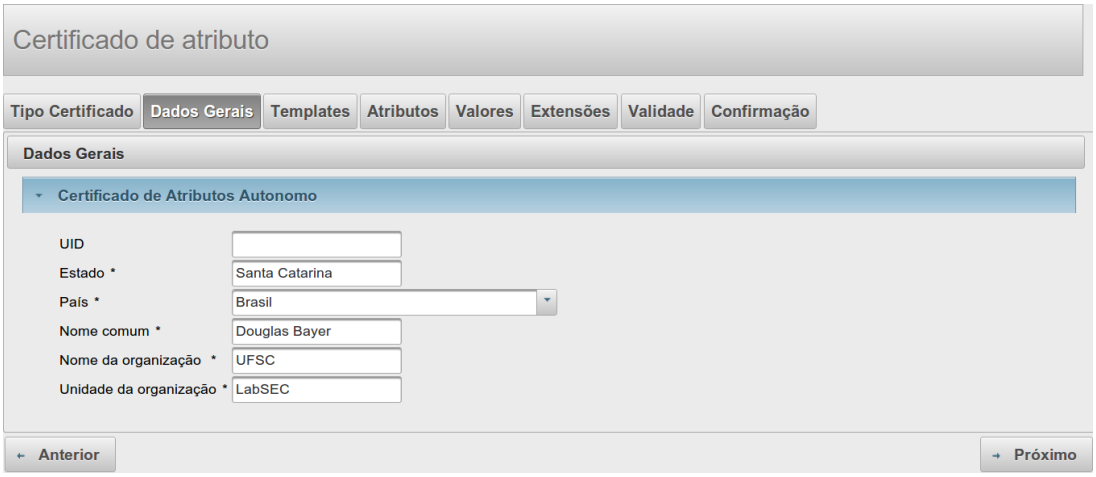

Figura 5.16: Informações necessárias para emitir um certificado autônomo.

pode selecionar quantos desejar. Uma vez selecionado um template, ele inclui automaticamente todos os atributos registrados sob aquele template no certificado de atributos. A utilização de um template não impossibilita o usuário de selecionar e excluir atributos individualmente na próxima etapa de emissão do certificado.

Figura 5.17: Aba para seleção de template.

5.2.11.4 Atributos

Esta aba possui a função de selecionar os atributos a serem incluídos no CA. O usuário deve escolher no mínimo um dos atributos disponíveis no sistema e nenhum dos atributos pode ser inserido mais de uma vez. A aba de atributos é ilustrada pela figura 5.18.

Figura 5.18: Aba para seleção de atributos.

5.2.11.5 Valores

Nesta etapa da emissão do certificado devem ser fornecidos os valores dos atributos. Cada um dos atributos selecionados previamente devem ter inclusos os seus valores e caso algum seja deixado em branco, uma mensagem de erro é exibida. Esta aba é demonstrada pela figura 5.19.

Figura 5.19: Aba para inclusão dos valores dos atributos.

5.2.11.6 Extensões

A aba de extensões contém todas as extensões definidas no DOC-ICP-16 que por sua vez, tem as mesmas extensões que são definidas pela RFC 5755. Estas extensões são as seguintes: *CRL Distribution Point*, *AC Target*, *No Revocation Available*, *Authority Key Identifier*, *Authority Info Access* e *Audit Identity*. As extensões são todas opcionais.

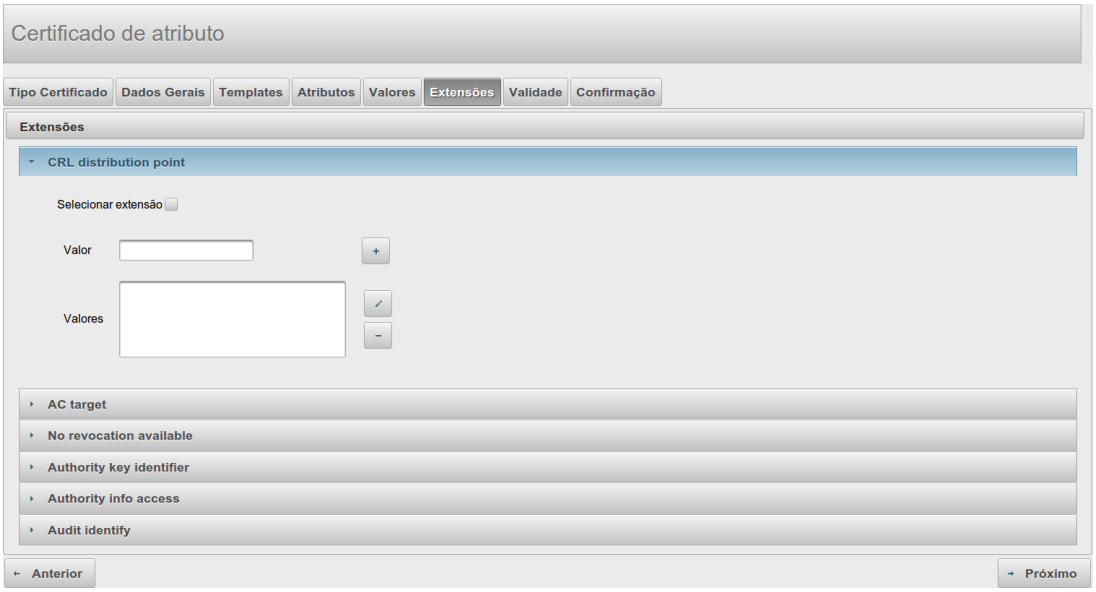

Figura 5.20: Aba para inclusão de extensões no certificado.

Aquelas extensões que forem selecionadas devem conter os valores apropriados, segundo as normas citadas. As extensões multi valoradas possuem um campo apropriado para inserção de vários valores, bem como sua edição e remoção. A aba de extensões é ilustrada pela figura 5.20.

5.2.11.7 Validade

Nesta etapa deve ser informado o período de validade do certificado e selecionado o algoritmo de assinatura. O período é informado através de dois campos "Inválido Antes" e "Inválido Após", ao clicar em um destes campos, uma aba contendo um calendário é exibida. Neste calendário, escolhem-se o ano, mês, dia, hora e minuto separadamente ou pode-se utilizar o botão "Agora" que define todos estes campos com os valores atuais do sistema, demonstrado na figura 5.21.

Ainda na etapa de validade é feita a escolha do algoritmo de assinatura, figura 5.22, dentre os seguintes disponíveis: *SHA256withECDSA*, *SHA256withRSA*, *SHA512withRSA*.

5.2.11.8 Confirmação

Esta é a etapa final do processo de emissão de certificado de atributos. Nesta etapa, são exibidos uma lista de informações e dados sobre o titular do CA, tal como: atributos inclusos, extensões, período de validade, algoritmo de assinatura. Nesta etapa, também é exibida a opção de publicar o certificado em base LDAP previamente configurada no sistema.

Figura 5.21: Calendário utilizado para indicar o período de validade.

Figura 5.22: Escolha do algoritmo de assinatura.

Caso algum dos dados não esteja de acordo, o usuário deve fazer o uso do botão "Voltar" até a etapa cuja informação está incorreta para sua alteração e poder, então, prosseguir novamente com a emissão e confirmação do certificado.

No advento de algum erro no processo de confirmação, uma mensagem em vermelho é exibida constatando erro e sua possível causa. Caso contrário, a tela de sucesso é exibida, como na figura 5.24.

A tela de sucesso tem um botão "Baixar" para efetuar o *download* do certificado em formato DER.

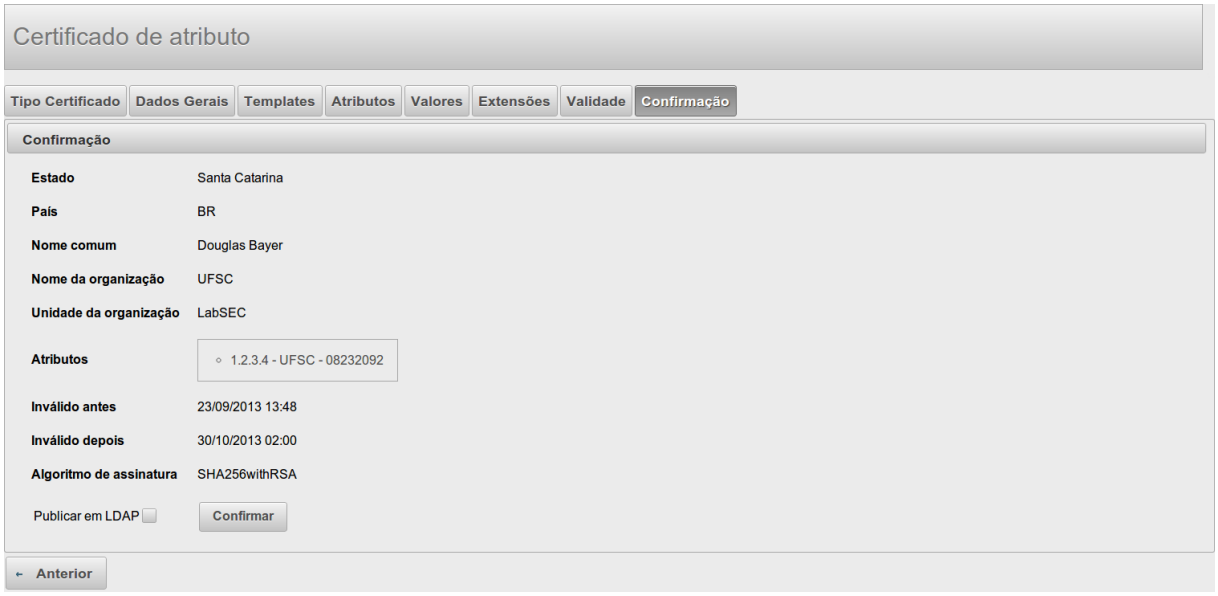

Figura 5.23: Confirmação dos dados contidos no CA.

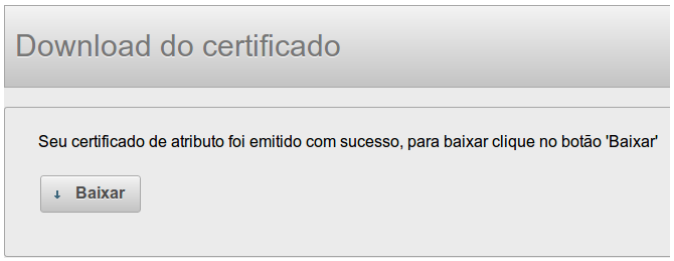

Figura 5.24: Certificado de atributo emitido com sucesso.

5.2.12 Revogar CA

Na seção para revogação de certificados de atributos, a primeira aba exibida mostra a lista dos certificados emitidos e que ainda são válidos, identificados pelo número de série e a informação do titular. Os certificados a serem revogados devem ser selecionados e o usuário deve avançar para a próxima etapa.

A aba seguinte é a de confirmação do procedimento, em que todos os certificados selecionados na tela anterior são exibidos para confirmação do administrador.

Após confirmar o procedimento, uma mensagem de sucesso será exibida caso não ocorra nenhum erro no processo. Se um erro venha a ocorrer, uma mensagem em vermelho contendo o erro será exibida.

Figura 5.25: Aba para seleção de certificados a revogar.

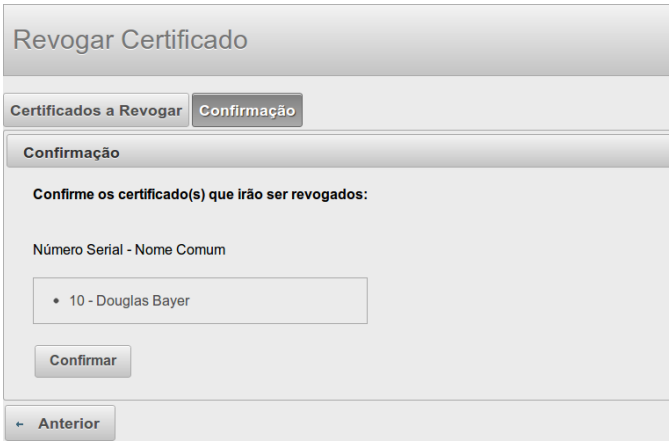

Figura 5.26: Aba de confirmação da revogação dos certificados.

5.2.13 Emissão LCR

Nesta seção é realizada a emissão da LCR para os certificados de atributos previamente revogados. A tela mostra todos os certificados, identificados pelo número de série e a informação do titular, que farão parte desta LCR e aguarda por confirmação do usuário, como demonstra figura 5.27.

No caso de confirmação e nenhum erro ocorrer é exibida a tela de sucesso, ilustrada na figura 5.28, com o botão “Baixar” para realizar o *download* do arquivo da LCR. No caso de ocorrer um erro, uma mensagem em vermelho com o erro é exibida.

5.2.14 Usuários

Na seção de usuários é possível visualizar os usuários cadastrados no sistema e cadastrar novos usuários. Um usuário é uma entidade que vai utilizar o sistema para as funções

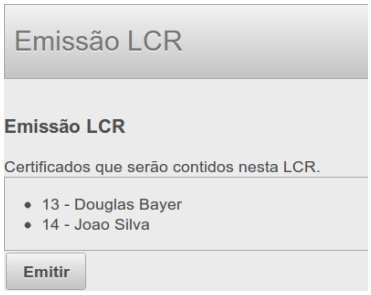

Figura 5.27: Confirmação da Lista de Certificados de Atributos Revogados.

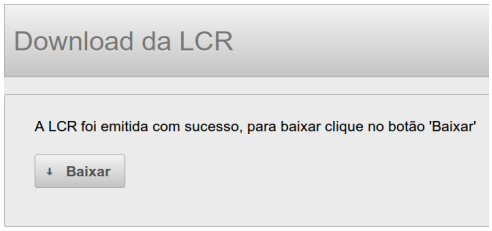

Figura 5.28: Lista de Certificados de Atributos Revogados emitida com sucesso.

de gerenciar CAs. Um usuário é identificado junto ao sistema através de seu nome de usuário, que é utilizado para determinar sua identidade e função perante o sistema. A figura 5.29 ilustra a visão inicial da página de usuários, demonstrando os atuais cadastros do sistema.

Figura 5.29: Usuários atualmente cadastrados no sistema.

O cadastro de usuários é feito através do botão “Novo”, que exibe uma nova tela para obtenção das informações necessárias. A tela de cadastro tem os seguintes campos: nome de usuário, senha, confirmação da senha e função no sistema. A função no sistema é selecionada a partir de um campo de escolha entre “administrador” e “operador”, a função determina a quais páginas do sistema o usuário tem acesso e consequentemente quais tarefas pode executar. A função de “operador” permite ao usuário acesso às páginas de emissão de certificado de atributos, revogação de certificado e emissão de LCR. A função de “administrador” permite

acesso irrestrito.

As demais funções para gerência dos usuários são de edição e remoção de usuários. A tela de edição exibe o nome do usuário e sua atual função no sistema e permite alterar sua senha e função, o nome de usuário não pode ser alterado.

Capítulo 6

Análise

Neste capítulo são abordados as dificuldades encontradas durante o processo de implementação e os testes realizados com o SGCA.

6.1 Testes Realizados

Durante o desenvolvimento, testes automatizados foram realizados para os métodos básicos através do *framework* de testes do Java, JUnit. Estes testes são executados novamente a cada alteração no código, garantindo assim que as alterações realizadas não causem comportamento inesperado dos métodos.

Para as funcionalidades de salvar atributos em XML, emitir o CA, emitir a LCR e publicar um CA em base LDAP, foram realizados testes para assegurar seu devido funcionamento. A funcionalidade de criar um arquivo XML contendo os atributos foi testada por meio da criação de vários arquivos com atributos e a aplicação tinha de reconhecer os arquivos gerados. Foi utilizado o *SimpleXML framework* para criar e manipular os arquivos XML. A aplicação correspondeu às expectativas sendo capaz de gerar e reconhecer arquivos XML contendo os atributos selecionados.

A emissão de certificados de atributos foi testada comparando a estrutura ASN.1 do certificado emitido pelo SGCA com as estruturas definidas nas normas. O certificado era emitido no formato DER pelo SGCA e então era utilizada a ferramenta *dumpasn1* para obter sua estrutura no formato ASN.1. A estrutura ASN.1 obtida foi comparada com as estruturas definidas nas normas. Como resultado, os CAs emitidos pelos SGCA estavam de acordo com as normas desejadas e sendo reconhecidos por algumas aplicações, como PERMIS. Um exemplo

de um certificado de atributo emitido pelo SGCA pode ser visualizado em formato ASN.1 no Anexo A 7.1. Para emissão deste CA foi selecionado o tipo certificado de atributo autônomo. Os dados fornecidos sobre o titular para os campos estado, país, nome comum, organização e unidade organizacional, são respectivamente: Santa Catarina, BR, Douglas Bayer, UFSC, LabSEC. Não foi selecionado nenhum template no processo de emissão deste certificado. Foi adicionado ao certificado o atributo de testes *Estudante*, OID 1.2.3.4, e o valor vinculado ao atributo foi *UFSC 082032092*. Nenhuma extensão foi adicionada ao certificado. O período de validade selecionado foi entre 30/09/2013 e 30/10/2013. O algoritmo de assinatura selecionado foi *sha256WithRSAEncryption*.

Os testes de revogação realizados consistiram em verificar a estrutura da LCR emitida, seu conteúdo e o status do certificado perante o sistema. Os certificados eram revogados no sistema, conferidas as atualizações de status e então emitida a LCR. Uma vez emitida a LCR, foi utilizada a ferramenta *dumpasn1* para obter a estrutura ASN.1 da LCR e verificar sua estrutura e conteúdo. Ao revogar um certificado o sistema atualiza seu estado corretamente e este passa a constar na lista dos certificados a serem emitidos na próxima LCR. Um exemplo de LCR emitida pelo SGCA contendo dois CAs revogados, identificados pelos números de série 13 e 14, pode ser encontrada em formato ASN.1 no Anexo B 7.1.

A integração com uma base LDAP foi testada utilizando o servidor diretório ApacheDS[1] e a ferramenta de gerência Apache Directory Studio[2]. Os testes foram realizados através da instalação do ApacheDS com suas configurações padrões, ou seja, não houve nenhuma mudança na estrutura da base LDAP depois que foi instalada. Durante o processo de emissão do CA, a opção de publicar o CA em diretório LDAP era marcada. Através do Apache Directory Studio foi possível visualizar a estrutura e o conteúdo do diretório. Uma nova entrada com o titular do certificado era criada e o certificado era armazenado nesta entrada. O certificado era reconhecido como um X.509, mas a ferramenta não exibe o seu conteúdo diretamente. Foi realizado o *download* do CA a partir do LDAP e utilizado a ferramenta *dumpasn1* para analisar se houveram quaisquer alterações na estrutura do certificado. Na análise não foram observadas quaisquer alterações do certificado, sendo este idêntico ao CA obtido diretamente do SGCA.

[1]Disponível em: http://directory.apache.org/apacheds

[2]Disponível em: http://directory.apache.org/studio

6.2 Dificuldades Encontradas

O conjunto de bibliotecas nativas do Java não possui suporte para certificados de atributos, apenas para certificados de chaves públicas. Então foi necessário utilizar a biblioteca Bouncy Castle. Devido à falta de documentação para esta biblioteca, houve a necessidade de uma dedicação maior para o seu entendimento. Adicionalmente, diversos testes tiveram que ser realizados até chegar a uma resposta final.

Apesar da existência de várias normas especificando os certificados de atributos, estes ainda são pouco utilizados e faltam exemplos práticos. As diferentes aplicações encontradas que utilizam certificados de atributos não são capazes de reconhecer os certificados emitidos uma pela outra. Esta falta de interoperabilidade mostra que, mesmo com as normas, cada implementação foi feita com uma interpretação diferente das mesmas ou seguindo diferentes padrões.

A falta de uma aplicação de código aberto e confiável, dificultou o processo de testes dos certificados de atributos. Não existindo um referencial de uma aplicação que seja reconhecida como correta, os certificados emitidos tinham de ser comparados somente com as normas.

Capítulo 7

Considerações Finais

O padrão X.509, definido pelo *International Telecommunication Union Telecommunication Standardization Sector* (ITU-T), estabelece a estrutura dos certificados de atributos e uma infraestrutura baseada em certificados de atributos. No entanto, os certificados de atributos no âmbito da ICP-Brasil fazem uso parcial desta infraestrutura. O modelo estabelecido utiliza a ICP-Brasil existente, permitindo que os titulares de certificados do tipo A3 e A4 de pessoa jurídica possam emitir CAs, i.e., sejam entidades emissoras de atributos.

A implementação do sistema foi feita levando em consideração o padrão do X.509, mas com objetivo de estar em conformidade com o padrão ICP-Brasil. O sistema foi desenvolvido segundo o conjunto normativo DOC-ICP-16, restringindo o tipo de vinculação do certificado aos especificados, fazendo uso da mesma nomenclatura e fornecendo os atributos em formato XML para auxiliar no requisito de publicação dos mesmos.

O formato web da aplicação reduz a necessidade de processamento por parte do cliente e permite que diversos tipos de sistemas possam emitir certificados de atributos. Esta facilidade de acesso à aplicação e o fato de não ser necessário realizar quaisquer instalações no sistema do usuário estimulam o uso da ferramenta e da tecnologia dos certificados de atributos.

A interface gráfica do SGCA foi desenvolvida em português, visando a utilização nacional e separa cada função em seções diferentes acessadas por um menu de modo a facilitar cada etapa de gerência dos certificados. O sistema exibe na interface diversas mensagens para auxilar seu usuário. Mensagens são exibidas se uma função executada foi bem sucedida ou não, exibindo onde ocorreu o erro ou sua possível causa.

Nos testes realizados, certificados de atributos no padrão ICP-Brasil foram emitidos e revogados com sucesso. A integração com a base LDAP foi bem sucedida, sendo possível

publicar o certificado de atributos e posteriormente recuperar o certificado e verificar sua integridade.

7.1 Trabalhos Futuros

Como trabalhos futuros são sugeridas algumas melhorias. A interface gráfica do SGCA atual libera muitas mensagens para o usuário de forma reativa, após o usuário cometer um erro é informado qual a forma adequada. Mudanças podem ser feitas para que esta informação seja exibida previamente para o usuário, deixando as mensagens de erro somente para indicar ocorrências inesperadas.

O sistema atualmente conta com métodos básicos para autenticação e segurança. O sistema pode ser aprimorado adicionando outros métodos de autenticação, e.g., através do certificado de chave pública do usuário. Também podem ser feitas algumas melhorias de segurança, como realizar a assinatura a partir de um Módulo de Segurança de Hardware onde se encontra a chave da EEA. Outra melhoria possível é a integração do sistema com *tokens* e *smartcards*, permitindo recuperar e armazenar certificados de atributos nestes.

Referências

[CHA 03a] CHADWICK, D.; OTENKO, A.; BALL, E. Role-based access control with x.509 attribute certificates. **Internet Computing, IEEE**, [S.l.], v.7, n.2, p.62 – 69, mar/apr, 2003.

[CHA 03b] CHADWICK, D.; OTENKO, A. The PERMIS X.509 role based privilege management infrastructure. **Future Generation Computer Systems**, [S.l.], v.19, n.2, p.277–289, February, 2003.

[CHA 04] CHADWICK, D. The x.509 privilege management infrastructure. **Security and Privacy in Advanced Networking Technologies**, University of Salford, Salford M5 4WT, England, 2004.

[CHA 08] CHADWICK, D. et al. Permis: a modular authorization infrastructure. **Concurr. Comput. : Pract. Exper.**, Chichester, UK, v.20, n.11, p.1341–1357, Agosto, 2008.

[COO 08] COOPER, D. et al. **Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile**. RFC 5280 (Proposed Standard).

[DIF 76] DIFFIE, W.; HELLMAN, M. New directions in cryptography. **Information Theory, IEEE Transactions on**, [S.l.], v.22, n.6, p.644 – 654, nov, 1976.

[dTdI a] DE TECNOLOGIA DA INFORMAçãO, I. N. **Certificado digital passa a ter validade de até cinco anos**. Disponível em <http://www.iti.gov.br/index.php/noticias/indice-de-noticias/3811-certificado-digital-passa-a-ter-validade-de-ate-cinco-anos.

[dTdI b] DE TECNOLOGIA DA INFORMAçãO, I. N. **ICP-Brasil**. Disponível em <http://www.iti.gov.br/icp-brasil. Acesso em 19 junho 2013.

[dTdI 12] DE TECNOLOGIA DA INFORMAçãO, I. N. **DOC-ICP-16: Visão Geral Sobre Certificado de Atributo**. Agosto, 2012.

[FAR 10] FARRELL, S.; HOUSLEY, R.; TURNER, S. **An Internet Attribute Certificate Profile for Authorization**. RFC 5755 (Proposed Standard).

[HOU 01] HOUSLEY, R.; POLK, T. **Planning for PKI: Best Practices Guide for Deploying Public Key Infrastructure**. 1st. ed. New York, NY, USA: John Wiley & Sons, Inc., 2001.

[MIC 05] MICROSOFT. **Managing Authorization and Access Control**. Disponível em <http://technet.microsoft.com/en-us/library/bb457115.aspx. Acesso em 24 dezembro 2012.

[MON 04] MONTENEGRO, J. A.; MOYA, F. A practical approach of x.509 attribute certificate framework as support to obtain privilege delegation. In: Katsikas, S. K.; Gritzalis, S.; Lopez, J., editors, EUROPKI, 2004. Springer, 2004. v.3093 of **Lecture Notes in Computer Science**, p.160–172.

[MYE 99] MYERS, M. et al. **X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP**. RFC 2560 (Proposed Standard).

[OSB 97] OSBORN, S. Mandatory access control and role-based access control revisited. In: PROCEEDINGS OF THE SECOND ACM WORKSHOP ON ROLE-BASED ACCESS CONTROL, 1997. **Proceedings...** New York, NY, USA: ACM, 1997. RBAC '97, p.31–40.

[OSB 07] OSBORN, S. Role-based access control. In: Petković, M.; Jonker, W., editors, SECURITY, PRIVACY, AND TRUST IN MODERN DATA MANAGEMENT, Data-Centric Systems and Applications, p.55–70. Springer Berlin Heidelberg, 2007.

[SAM 01] SAMARATI, P.; DE VIMERCATI, S. Access control: Policies, models, and mechanisms. In: Focardi, R.; Gorrieri, R., editors, FOUNDATIONS OF SECURITY ANALYSIS AND DESIGN, v.2171 of **Lecture Notes in Computer Science**, p.137–196. Springer Berlin / Heidelberg, 2001.

[SAN 94] SANDHU, R.; SAMARATI, P. Access control: principle and practice. **Communications Magazine, IEEE**, [S.l.], v.32, n.9, p.40 –48, sep, 1994.

[SAN 96] SANDHU, R. et al. Role-based access control models. **Computer**, [S.l.], v.29, n.2, p.38 –47, feb, 1996.

[SAN 00] SANDHU, R.; FERRAIOLO, D.; KUHN, R. The nist model for role-based access control: towards a unified standard. In: PROCEEDINGS OF THE FIFTH ACM WORKSHOP ON ROLE-BASED ACCESS CONTROL, 2000. **Proceedings...** New York, NY, USA: ACM, 2000. RBAC '00, p.47–63.

[STR 13] STRONGSWAN. **strongSwan Wiki**. Disponível em <http://wiki.strongswan.org/projects/strongswan/wiki. Acesso em 23 junho 2013.

[UNI 08] UNION, I. T. Itu-t recommendation x.509 — iso/iec 9594-8: "information technology - open systems interconnection - the directory: Public-key and attribute certificate frameworks". 2008. 162+ p. Relatório técnico.

Anexos

Anexo A - Certificado de Atributos

Abaixo é apresentado a estrutura ASN.1 obtida através da ferramenta *dumpasn1*, a partir de um certificado de atributos gerado com o SGCA no formato DER.

```
  0  446: SEQUENCE {
  4  295:   SEQUENCE {
  8    1:     INTEGER 1
 11  100:     SEQUENCE {
 13   98:       [1] {
 15   96:         [4] {
 17   94:           SEQUENCE {
 19   22:             SET {
 21   20:               SEQUENCE {
 23    3:                 OBJECT IDENTIFIER commonName (2 5 4 3)
 28   13:                 UTF8String 'Douglas Bayer'
      :                 }
      :               }
 43   15:             SET {
 45   13:               SEQUENCE {
 47    3:                 OBJECT IDENTIFIER organizationalUnitName (2 5 4
      11)
 52    6:                 UTF8String 'LabSEC'
      :                 }
      :               }
 60   13:             SET {
 62   11:               SEQUENCE {
 64    3:                 OBJECT IDENTIFIER organizationName (2 5 4 10)
 69    4:                 UTF8String 'UFSC'
      :                 }
      :               }
 75   23:             SET {
 77   21:               SEQUENCE {
```

```
 79    3:                    OBJECT IDENTIFIER stateOrProvinceName (2 5 4 8)
 84   14:                    UTF8String 'Santa Catarina'
      :                    }
      :                  }
100   11:                SET {
102    9:                  SEQUENCE {
104    3:                    OBJECT IDENTIFIER countryName (2 5 4 6)
109    2:                    UTF8String 'BR'
      :                    }
      :                  }
      :                }
      :              }
      :            }
      :          }
113  108:      [0] {
115  106:        SEQUENCE {
117  104:          [4] {
119  102:            SEQUENCE {
121   18:              SET {
123   16:                SEQUENCE {
125    3:                  OBJECT IDENTIFIER commonName (2 5 4 3)
130    9:                  UTF8String 'Usuario 1'
      :                  }
      :                }
141   15:              SET {
143   13:                SEQUENCE {
145    3:                  OBJECT IDENTIFIER organizationalUnitName (2 5 4
         11)
150    6:                  UTF8String 'LabSEC'
      :                  }
      :                }
158   13:              SET {
160   11:                SEQUENCE {
162    3:                  OBJECT IDENTIFIER organizationName (2 5 4 10)
167    4:                  UTF8String 'UFSC'
      :                  }
      :                }
173   22:              SET {
175   20:                SEQUENCE {
```

```
 66 | 177   3:                   OBJECT IDENTIFIER localityName (2 5 4 7)
 67 | 182  13:                   UTF8String 'Florianopolis'
 68 |       :                   }
 69 |       :                 }
 70 | 197  11:               SET {
 71 | 199   9:                 SEQUENCE {
 72 | 201   3:                   OBJECT IDENTIFIER stateOrProvinceName (2 5 4 8)
 73 | 206   2:                   UTF8String 'SC'
 74 |       :                   }
 75 |       :                 }
 76 | 210  11:               SET {
 77 | 212   9:                 SEQUENCE {
 78 | 214   3:                   OBJECT IDENTIFIER countryName (2 5 4 6)
 79 | 219   2:                   PrintableString 'BR'
 80 |       :                   }
 81 |       :                 }
 82 |       :               }
 83 |       :             }
 84 |       :           }
 85 |       :         }
 86 | 223  13:       SEQUENCE {
 87 | 225   9:         OBJECT IDENTIFIER
 88 |       :           sha256WithRSAEncryption (1 2 840 113549 1 1 11)
 89 | 236   0:         NULL
 90 |       :         }
 91 | 238   1:       INTEGER 15
 92 | 241  34:       SEQUENCE {
 93 | 243  15:         GeneralizedTime 30/09/2013 21:47:00 GMT
 94 | 260  15:         GeneralizedTime 30/10/2013 20:47:00 GMT
 95 |       :         }
 96 | 277  24:       SEQUENCE {
 97 | 279  22:         SEQUENCE {
 98 | 281   3:           OBJECT IDENTIFIER '1 2 3 4'
 99 | 286  15:           SET {
100 | 288  13:             UTF8String 'UFSC_08232092'
101 |       :             }
102 |       :           }
103 |       :         }
104 |       :       }
```

```
105  303   13:   SEQUENCE {
106  305    9:     OBJECT IDENTIFIER sha256WithRSAEncryption (1 2 840 113549 1
                   1 11)
107  316    0:     NULL
108        :     }
109  318  129:   BIT STRING
110        :     CA 20 7D CA 4F E5 71 7F E2 5F 17 9D 71 4C 25 63
111        :     1D 09 3B C1 BC 65 E8 F9 97 37 D5 24 C2 41 5A 4B
112        :     1B ED EA 02 04 A8 2D 98 57 E9 5F FB 23 98 3C 2D
113        :     17 91 72 7A F9 35 99 21 F2 02 F6 AB D8 F3 80 9A
114        :     BF 43 BC 02 01 95 D5 5E 2F BD 96 2B DF 4A 8B 25
115        :     B5 8B B5 FE FD 70 9D 8D 0F F7 2A E4 15 2B 9E 6B
116        :     93 43 73 18 E9 4D 24 B2 57 A1 94 BE A7 42 87 E3
117        :     AB A7 79 CC 4C 77 E4 5F EE DD 70 FA 6E BD 7F 54
118        :   }
```

Anexo B - Lista de Certificados de Atributos Revogados

A seguir é apresentado a estrutura ASN.1 obtida através da ferramenta *dumpasn1*, a partir de uma lista de certificados revogados gerada com o SGCA.

```
  0  354: SEQUENCE {
  4  207:   SEQUENCE {
  7    1:     INTEGER 1
 10   10:     SEQUENCE {
 12    8:       OBJECT IDENTIFIER ecdsaWithSHA256 (1 2 840 10045 4 3 2)
       :       }
 22  102:     SEQUENCE {
 24   18:       SET {
 26   16:         SEQUENCE {
 28    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 33    9:           UTF8String 'Usuario 1'
       :           }
       :         }
 44   15:       SET {
 46   13:         SEQUENCE {
 48    3:           OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
 53    6:           UTF8String 'LabSEC'
       :           }
       :         }
 61   13:       SET {
 63   11:         SEQUENCE {
 65    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
 70    4:           UTF8String 'UFSC'
       :           }
       :         }
```

```
 26 |  76   22:         SET {
 27 |  78   20:           SEQUENCE {
 28 |  80    3:             OBJECT IDENTIFIER localityName (2 5 4 7)
 29 |  85   13:             UTF8String 'Florianopolis'
 30 |       :             }
 31 |       :           }
 32 | 100   11:         SET {
 33 | 102    9:           SEQUENCE {
 34 | 104    3:             OBJECT IDENTIFIER stateOrProvinceName (2 5 4 8)
 35 | 109    2:             UTF8String 'SC'
 36 |       :             }
 37 |       :           }
 38 | 113   11:         SET {
 39 | 115    9:           SEQUENCE {
 40 | 117    3:             OBJECT IDENTIFIER countryName (2 5 4 6)
 41 | 122    2:             PrintableString 'BR'
 42 |       :             }
 43 |       :           }
 44 |       :         }
 45 | 126   13:       UTCTime 28/09/2013 14:30:00 GMT
 46 | 141   13:       UTCTime 30/09/2013 14:30:00 GMT
 47 | 156   40:       SEQUENCE {
 48 | 158   18:         SEQUENCE {
 49 | 160    1:           INTEGER 13
 50 | 163   13:           UTCTime 28/09/2013 14:06:55 GMT
 51 |       :           }
 52 | 178   18:         SEQUENCE {
 53 | 180    1:           INTEGER 14
 54 | 183   13:           UTCTime 28/09/2013 14:06:55 GMT
 55 |       :           }
 56 |       :         }
 57 | 198   14:       [0] {
 58 | 200   12:         SEQUENCE {
 59 | 202   10:           SEQUENCE {
 60 | 204    3:             OBJECT IDENTIFIER cRLNumber (2 5 29 20)
 61 | 209    3:             OCTET STRING, encapsulates {
 62 | 211    1:               INTEGER 1
 63 |       :               }
 64 |       :             }
```

```
 65 |          :            }
 66 |          :          }
 67 |          :        }
 68 |  214    10:     SEQUENCE {
 69 |  216     8:       OBJECT IDENTIFIER ecdsaWithSHA256 (1 2 840 10045 4 3 2)
 70 |          :       }
 71 |  226   129:     BIT STRING
 72 |          :       B6 B2 7F DF F6 C1 FF AB 69 A6 B6 EF EB C7 31 86
 73 |          :       A4 B9 01 03 B3 53 C0 5B 55 3F 79 EF B5 BB 42 9B
 74 |          :       B0 A7 41 2F F6 71 BB D0 98 E1 6C B8 FF 54 95 CE
 75 |          :       35 89 FB 65 C7 12 17 22 A2 87 DF 87 A3 11 77 37
 76 |          :       70 8B A7 7F 41 F6 95 83 FC 35 13 7F EB 63 A3 B5
 77 |          :       B0 F4 CF 51 DA 8F 84 AC 1C AD F6 4D 92 BC BD B4
 78 |          :       D9 64 33 99 F8 FC D4 8C 5C 8C 39 C2 4E 01 DC 16
 79 |          :       AD 3C 1F 41 42 05 51 ED E2 C8 10 72 E6 47 F8 40
 80 |          :     }
```

Anexo C - Diagramas

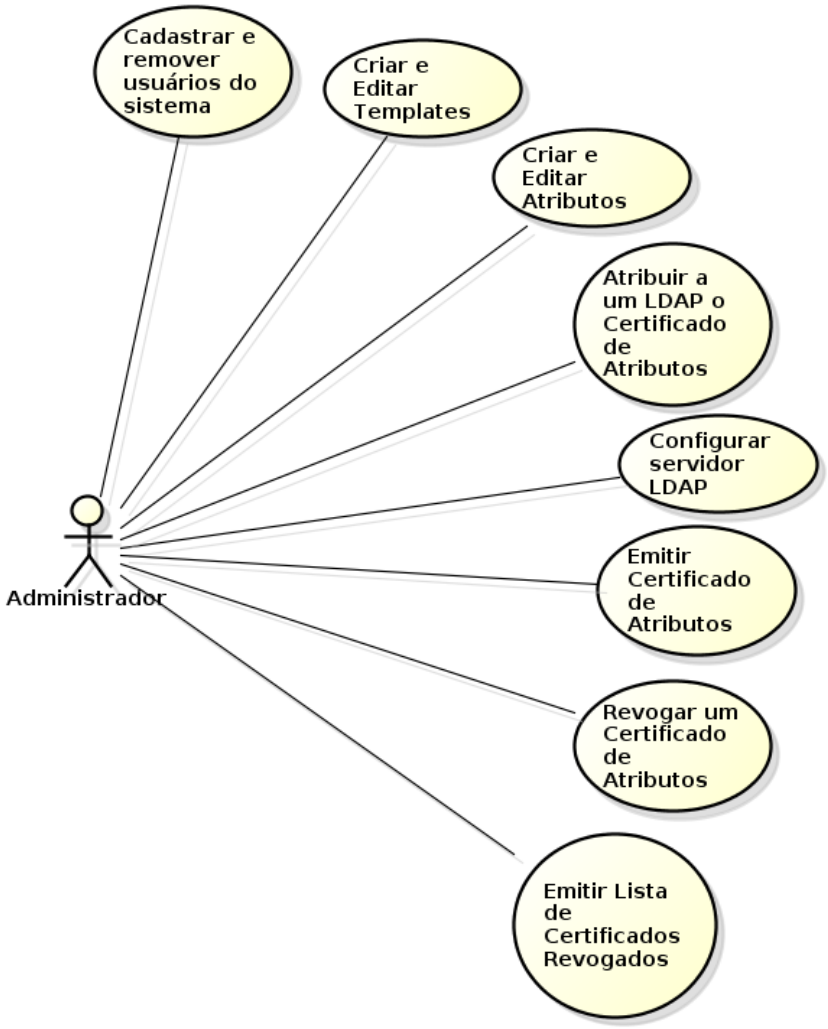

Figura 7.1: Diagrama de Casos de Uso do Administrador.

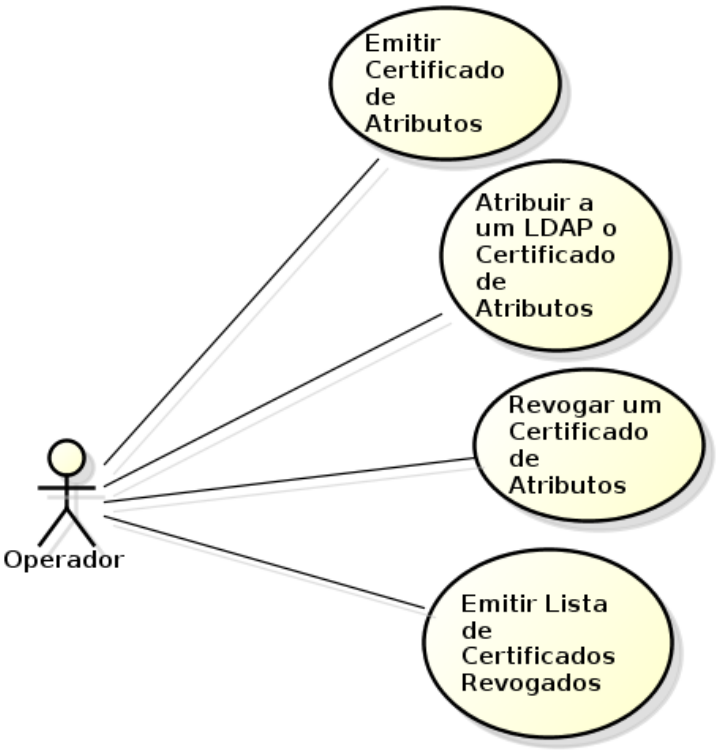

Figura 7.2: Diagrama de Casos de Uso do Operador.

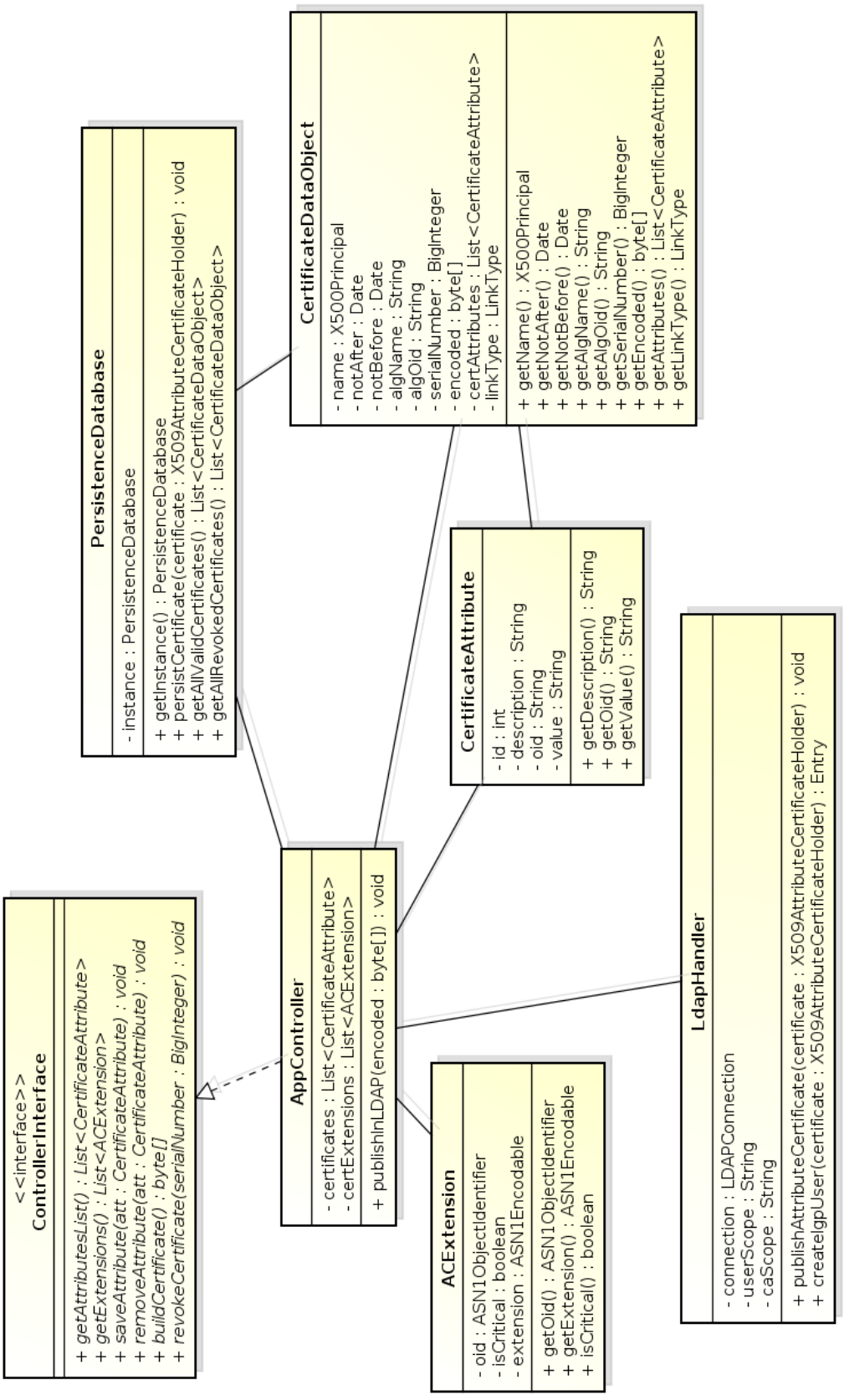

Figura 7.3: Diagrama de Classes.

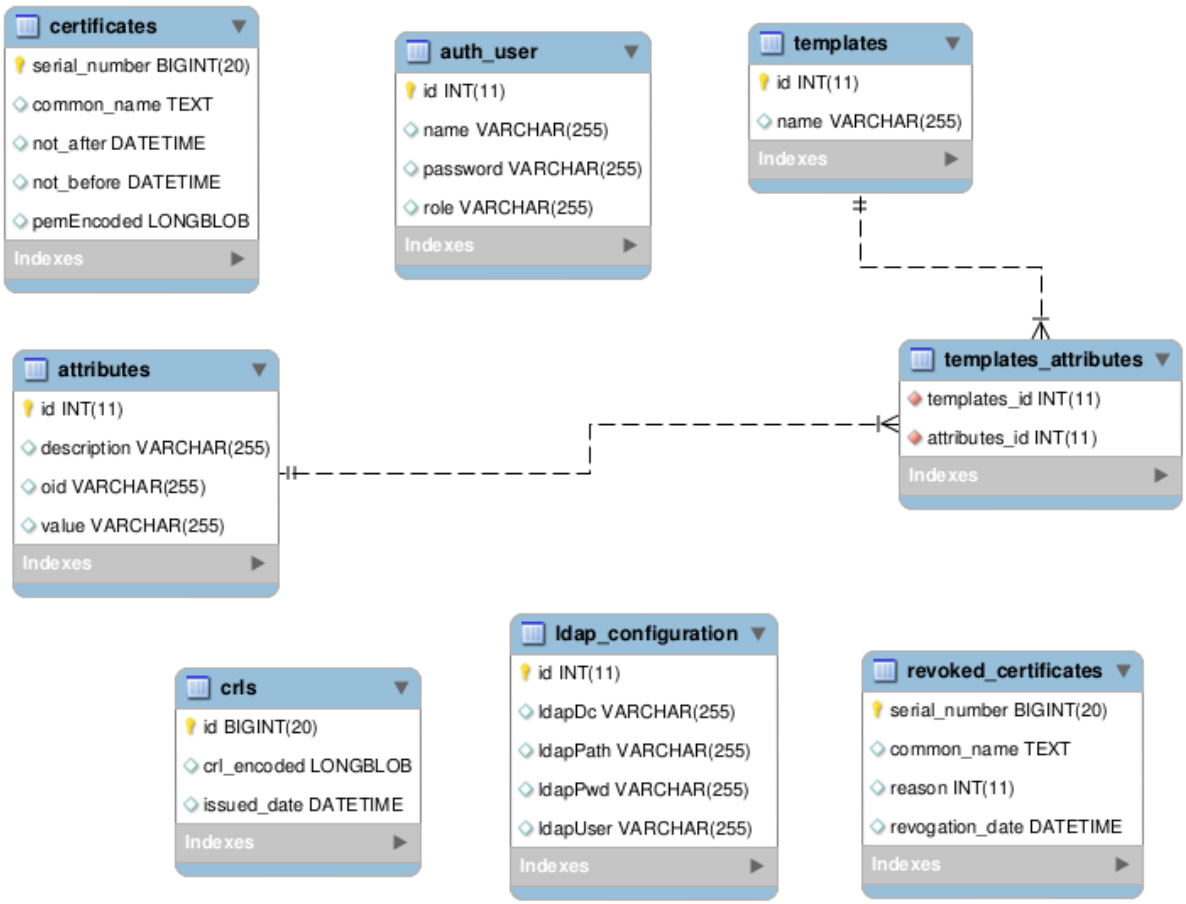

Figura 7.4: Modelagem do Banco de Dados.