|  |
| --- |
|  |
| CA研制方案 |
|  |
|  |
| XXXX单位 |
| 二○XX年XX月 |

目录

[XXXX研制方案 4](#_Toc452043530)

[1 概述 4](#_Toc452043531)

[1.1 研制背景 4](#_Toc452043532)

[1.2 任务来源和依据 4](#_Toc452043533)

[1.3 研制目标 4](#_Toc452043534)

[2 主要功能与技术指标 4](#_Toc452043535)

[2.1 主要功能 4](#_Toc452043536)

[2.2 技术指标 4](#_Toc452043537)

[3 方案设计 4](#_Toc452043538)

[3.1 系统结构及组成 4](#_Toc452043539)

[3.2 工作原理 5](#_Toc452043540)

[3.3 硬件设计 5](#_Toc452043541)

[3.4 软件设计 5](#_Toc452043542)

[3.5 协议设计 5](#_Toc452043543)

[3.6 系统接口与信息交换关系（可选） 5](#_Toc452043544)

[4 密码密钥配用 6](#_Toc452043545)

[4.1 安全保护及管理保障需求分析 6](#_Toc452043546)

[4.2 密码配用（密码产品或模块请分别参考对应模板） 7](#_Toc452043547)

[4.3 密钥配用 7](#_Toc452043548)

[4.4 密码协议 8](#_Toc452043549)

[5 安全保密措施设计 9](#_Toc452043550)

[5.1 密码资源保护机制 9](#_Toc452043551)

[5.2 故障诊断和处置机制 10](#_Toc452043552)

[5.3 逻辑安全防护机制 10](#_Toc452043553)

[5.4 物理安全防护机制 10](#_Toc452043554)

[5.5 前向安全防护机制 10](#_Toc452043555)

[6 电子元器件国产化应用 10](#_Toc452043556)

[7 通用性设计 10](#_Toc452043557)

[7.1. 可靠性设计 11](#_Toc452043558)

[7.2. 维修性设计 11](#_Toc452043559)

[7.3. 保障性设计 11](#_Toc452043560)

[7.4. 安全性设计 11](#_Toc452043561)

[7.5. 测试性设计 11](#_Toc452043562)

[7.6. 环境适应性设计 11](#_Toc452043563)

[8 质量控制与标准化管理 11](#_Toc452043564)

[8.1. 质量控制要求 11](#_Toc452043565)

[8.2. 标准化管理要求 11](#_Toc452043566)

[9 关键技术分析 12](#_Toc452043567)

[10 任务分工 12](#_Toc452043568)

[11 进度安排 12](#_Toc452043569)

CA研制方案

# 概述

## 研制背景

随着计算机网络技术的迅速发展和信息化建设的大力推广，越来越多的传统办公和业务处理模式开始走向电子化和网络化，从而极大地提高了效率、节约了成本。与传统的面对面的手工处理方式相比，基于网络的电子化业务处理系统必须解决以下问题：

1. 如何在网络上识别用户的真实身份；
2. 如何保证网络上传送的业务数据不被篡改；
3. 如何保证网络上传送的业务数据的机密性；
4. 如何使网络上的用户行为不可否认；

基于公开密钥算法的数字签名技术和加密技术，为解决上述问题提供了理论依据和技术可行性；同时，《中华人民共和国电子签名法》的颁布和实施为数字签名的使用提供了法律依据，使得数字签名与传统的手工签字和盖章具有了同等的法律效力。PKI（Public Key Infrastructure）是使用公开密钥密码技术来提供和实施安全服务的基础设施，其中CA（Certificate Authority）系统是PKI体系的核心，主要实现数字证书的发放和密钥管理等功能。

数字证书由权威公正的CA中心签发，是网络用户的身份证明。使用数字证书，结合数字签名、数字信封等密码技术，可以实现对网上用户的身份认证，保障网上信息传送的真实性、完整性、保密性和不可否认性。

数字证书目前已广泛应用于安全电子邮件、网上商城、网上办公、网上签约、网上银行、网上证券、网上税务等行业和业务领域。

基于上述现状，本系统需要解决数据的签名问题和法律效力问题，从而提高本的便捷性和管理效率。鉴于数字证书、数字签名的广泛应用和相关法律的保障，本单位规划建设CA及数字签名认证系统，主要需求如下：

1. 建设CA系统或采用第三方CA，为本用户申请数字证书；
2. 在现有本系统中加入对数据的签名功能，存储数据签名并提供对签名的认证功能；

## 任务来源和依据

为了解决网上用户的身份证明问题，需要为用户颁发数字证书。数字证书由CA中心签发，目前在实际应用中主要存在两种类型的CA：

1. 独立的第三方CA
   1. 跨区域的CA，如：中国电信的CTCA、中国人民银行的CFCA；
   2. 地域性的CA，如：广东电子商务认证中心CNCA、上海电子商务认证中心SHECA，以及其他各省电子商务认证中心；
2. 各类应用系统自己建设的CA

如：招商银行、建设银行等建设的用于服务各自网上银行的CA；海关、税务等建设的服务各自网上报税系统的CA；

这两种类型的CA在实际使用过程中各有优劣，以下将进行分析和比较：

1.2.1.CA建设与使用的分析

 采用独立权威的第三方CA与自建CA的比较

|  |  |  |
| --- | --- | --- |
|  | 独立权威的第三方CA | 自建CA |
| 建设/使用成本 | 一般由第三方CA按用户收取年费，建设投入和证书使用成本较高 | 一次投入建设成本，建成后可为用户免费发放证书，成本较低 |
| 证书的有效性检查 | 证书的有效性检查:由第三方CA提供的CRL（证书撤销列表，实效性较差）或者OCSP（在线证书状态查询，依赖于外部系统，易形成性能瓶颈） | 与业务系统紧密结合，对证书有效性的控制和检查实时准确 |
| 定制化/灵活性 | 按第三方规定的流程申请证书，受制于第三方的系统，还需要将用户证书导入业务系统中，灵活性较差 | 可以与业务系统紧密结合，满足业务系统的定制化需求，灵活性高 |
| 故障响应时间 | 故障解决依赖于第三方 | 响应及时 |

备注：成本比较

权威的第三方CA的使用成本：10元/年/用户 × 10万用户 = 100万元/年

自建CA的系统建设成本 < 100万元，并且只是一次性的投入

综合比较而言，自建CA优于采用第三方CA，因此推荐自建CA。

1.2.2 证书存储方式的分析

 使用普通文件存储方式与USB智能卡存储方式的比较

|  |  |  |
| --- | --- | --- |
|  | 普通文件 | USB智能卡 |
| 成本 | 磁盘成本较低，优盘成本较高 | USB智能卡成本适中 |
| 安全性 | 私钥可以复制，易泄密 | 钥被固化在USB智能卡中，无法复制，安全性高 |
| 方便性 | 私使用磁盘或优盘等方式携带，使用时需要用户选择证书和私钥 | 携带方便，但需要安装USB智能卡驱动，使用时可自动读取用户证书 |

USB智能卡自带CPU，内置芯片操作系统（COS）；采用USB接口，易于使用和携带；支持RSA非对称算法和DES、3DES等对称算法；支持RSA公司的PKCS#11标准和微软的CSP标准；支持Windows98/NT/2000/XP/2003等操作系统。USB智能卡可支持国密算法SSF33，并通过国家密码管理委员会的检测。

1.2.3 签名数据类型的分析

系统需要进行电子签名并存储的数据主要有以下三类：

1. 上报表单的数据签名

用户在网上填写的各类表单需要由用户的私钥进行电子签名；

1. 上报数据文件的数据签名

用户上传的数据文件，其内容需要由用户的私钥进行电子签名；

1. 下载数据文件的数据签名

用户通过系统生成并下载的确认数据文件（如：PDF格式），其内容需要由用户的私钥进行电子签名并回传至系统存储。

## 研制目标

从满足使用需求、提升功能性能、提高技术水平等方面出发，确立研制目标。

# 主要功能与技术指标

## 主要功能

满足用户安全登陆认证，为用户颁发数字证书，对用户私钥进行安全管理

## 技术指标

1. 密钥中心独立设计，可以正常产生RSA1024、RSA2048、SM2 256类型的密钥对，并且方便以后扩展
2. CA中心可以正常颁发用户证书，吊销用户证书，并且验证证书的的有效性，提供各类接口供其他系统调用
3. 离线CA中心，可以颁发CA证书，证书的导入，证书链的导入，CSR产生，CA的证书离线密钥的管理，管理员五三认证的设计实现
4. RA中心，能够对用户证书申请，进行批准，把ca产生的用户CSR发送给用户

# 方案设计

## 系统结构及组成

该项目主要有四个项目组成：

1. KMS：密钥管理系统

主要负责用户密钥对的产生和安全保存，并提供用户密钥对信息查询接口，已经生成用户CSR文件，供CA系统使用

1. CAS：系统证书颁发系统

主要负责颁发用户证书，吊销用户证书，并管理用户证书，提供系统日志审计功能

1. 离线CA：CA证书的颁发管理系统

主要负责CA证书的生成，导入，证书链导入等功能

1. RAS：用户注册申请管理系统

用户负责对用户提交的证书申请记录批准或者拒绝



## **.逻辑结构**

## 用户通过用户中心提交用户证书申请请求，RA管理员在RA系统进行审批，审批通过的的请求发送到CA系统，CA系统根据用户的所在RA系统的RA管理员账户证书生成配置，调用kms产生的CSR，生成用户证书，依次发送到用户手中。

系统之间通讯，通讯基于dubbo协议，为了通讯的安全性，提前颁发通讯证书，RA系统配置后，每次调用CA接口，或者CA调用RA接口，对发送的数据进行加密，当CA系统接受到RA系统提交的申请后，使用相应的证书证书对应的私钥进行解密

硬件加密机或者智能卡连接使用JCE的机制，遵循PKCS11接口；USBKey通过浏览器插件，浏览器读取用户数字证书，或者将数字证书安装到key中，进行登陆验证

## 3.3工作原理

使用离线CA产生三级CA证书，将证书导出妥善保存，启动KMS系统预先先产生一些密钥对，供用户证书颁发使用，CA系统启动后初始化超级管理员，导入系统根证书，或者创建系统根证书，使用超级管理员登陆系统创建CA管理员，CA管理员登陆系统后上传二级证书，并创建RA管理员，启动RA系统，RA管理员登陆，上传RA证书，用户注册系统用户，提交证书申请，RA批准后去CA中心制作用户CSR，并将CSR安全发送给用户，用户用CSR自己输入密钥的保护密码，并提交到RA，RA对改条记录信息再次审核，无误后提交CA中心，颁发用户证书，供用户下载或者安装证书。



## 硬件设计

该项目主要硬件为，标准linux和windows系统用于安装系统，符合PKCS11的USBKey供用户存储用户证书，可以产生RSA和SM2的硬件加密机或者加密卡

## 软件设计

1. KMS：密钥管理系统，主要负责用户密钥对的安全管理功能
   1. 密钥管理模块

对用户的密钥的生成，查询，CSR生成，进行统一管理

* 1. 用户管理模块

管理KMS系统的管理员

* 1. 签名模块

对用户的CSR产生等提交签名服务

* 1. 日志审计模块
     1. 用户日志审计

记录kms管理用用户的登陆日志

* + 1. Key日志审计

记录密钥相关日志，如key的创建，查询，CSR生成记录等

1. CA：证书认证系统，主要负责证书的管理，用户的管理
   1. 证书模块，负责证书颁发，吊销
   2. 签名模块，签名服务
   3. 用户模块，管理系统用
   4. 日志审计模块，
      1. 证书
      2. 用户日志
      3. CRL日志
2. 离线CA模块
   1. CA证书管理模块，主要负责CA证书的申请，证书链导出，交换密钥文件导出，可信证书jks导出，证书链导入和导出
   2. 用户管理模块，负责离线CA管理员的管理
   3. 离线密钥管理模块，可以导入jks文件。或者自己产生符合用户要求的CA证书密钥对
   4. 日志审计模块，
      1. 用户日志，记录用户登录日志，注销日志
      2. 证书日志，记录CA证书生成，导入，等
3. RA：注册中心，主要负责用户的证书审核
   1. 证书模块，主要负责用户的批准审核工作
   2. 日志模块：
      1. 管理员日志，记录管理员批准记录
      2. 证书日志，记录证书的申请记录

## 协议设计

1. 系统各个子系统间交换使用dubbo协议，并且zookeeper进行负载。
2. 用户交换前端使用http1.1协议进行通讯

## 系统接口与信息交换关系（可选）

1. KMS系统
   1. 可用用户密钥查询接口

根据用户需要产生用户密钥对，私钥不可导出

* 1. 用户CSR产生接口、

根据用户主题和使用的密钥对生成证书请求信息

1. CA系统
   1. 日志查询接口
      1. 查询用户登录日志，证书生成日志，管理员初始化，证书导入日志
   2. 证书查询接口

根据用户的条件查询相关的已发布的签名证书

* 1. 颁发证书接口

使用用户的CSR为用户颁发x509格式的数字证书

* 1. 用户认证接口

使用该接口可以用用户名、手机、邮箱及数字证书验证证书的有效性

* 1. 证书吊销接口

吊销用户数字证书

* 1. CRL下载接口

下载已经发布的CRL文件

* 1. 用户查询接口

管理角色查询相关用户

* 1. 离线jks查询接口

查询CA证书的可用密钥别名集合

* 1. CA证书导入接口

CA管理员证书的导入接口

* 1. 用户注册接口

普通用户的注册过程

1. 离线CA系统
   1. 管理员认证接口

离线管理员通过用户名和密码认证

* 1. 自签名证书申请接口

根据密钥对和证书主题颁发自签名CA证书

* 1. 下级CA证书生成接口

根据CSR成功下级CA证书

* 1. CA证书CSR生成级下载接口

根据密钥对和主题产生CA证书

* 1. P7证书链导入和导出接口

导入P7格式的证书链，导入前请先正确配置jks格式的密钥库

* 1. P12导出接口

导出CA证书的密钥交换文件，请妥善保存

* 1. 可信CA jks导出接口

导出可信CA的jks文件，供web服务器使用

1. 硬件加密机接口
   1. 完全遵循PKCS11接口，具体请查看[PKCS11接口文档](http://wenku.baidu.com/link?url=EIqja8xvvl5SJ21sS9_i07RMALrmmRRmoI_pzjle6QTwnkAnDAzSjsizdTrvZqLc1MPxPAhZrB1Y_AfUe_PwqEijSCTU-QO5HA7KtgbwtyO)
2. USBKey设备接口
   1. 读取证书接口
      1. 调用接口返回用户的签名证书base64字符串
   2. 写证书链接口
      1. 写入P12用户私钥交互文件字符串
   3. 对数据进行签名接口
      1. 传入数据，返回签名接口
   4. 对数据进行验证接口
      1. 传入数据，返回签名是否成功
   5. 加密接口
      1. 传入数据，返回加密值
   6. 解密接口
      1. 传入数据，返回解密结果
3. 浏览器控件接口
   1. 系统证书查询接口
      1. 根据提供的的主题，序列号，csp返回证书列表
   2. P7证书链安装接口
      1. 安装p7格式的证书到系统或者usb设备
      2. 接口名:acceptPLCS7(P7证书链)
      3. 设置参数
         1. DeleteRequestCert:是否删除以前证书
         2. WriteCertToCSP:是否写证书到CSP
      4. 出参：成功返回true
   3. 重置接口
      1. 初始化
      2. 接口名:reset
   4. 加密服务提供者列表接口
      1. 返回系统当前的csp列表
      2. 接口名称：enumProviders
      3. 设置参数：ProviderType，csp类型
      4. 传出参数：csp集合
   5. PKCS10的CSR创建接口
      1. 根据用户的私钥和主题生成证书请求信息
      2. 接口名:createPKCS10(名称，版本)
      3. 设置传入参数：
         1. ProviderName:csp名称
         2. ProviderType:csp类型
         3. DeleteRequestCert:是否删除已经存在的证书
         4. UseExistingKeySet：是否使用已经存在的密钥集合
         5. ContainerName：密钥容器名称
         6. HashAlgorithm：哈希算法
         7. KeySpec：密钥生成使用的实现
         8. GenKeyFlags:生成密钥需要的参数
      4. 出参：用户的CSR
4. RA中心
   1. 证书信息提交接口

提交用户证书申请信息

* 1. 证书批准接口

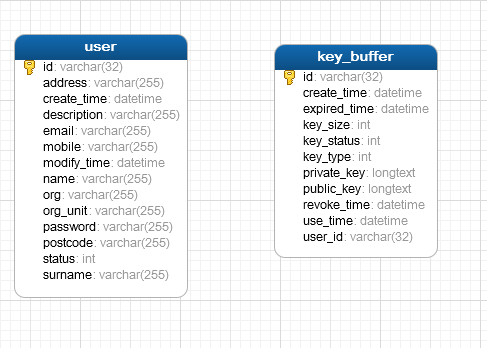
RA管理员批准证书请求

* 1. 证书CSR查询接口

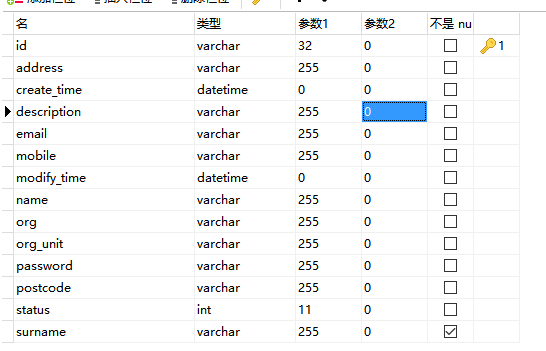
查询用户的CSR，并把CSR发送给用户

# 3.7 数据库设计

1. KMS系统数据库
   1. ER图



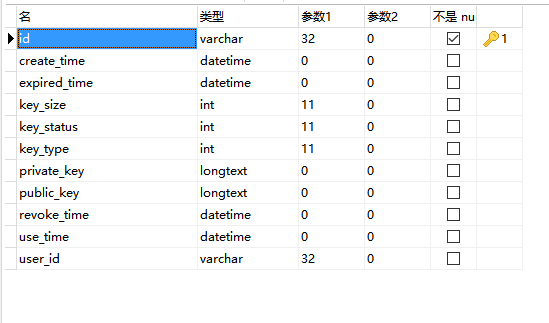
* 1. 数据表说明
     1. 用户表
        1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段名 | 描述 | 其他 |
| Id | id主键 |  |
| Address | 地址 |  |
| Create\_time | 创建时间 |  |
| Description | 描述 |  |
| Email | 邮箱 |  |
| Mobile | 手机号 |  |
| Modify\_time | 修改时间 |  |
| Name | 名称 |  |
| Org | 单位 |  |
| Password | 密码 |  |
| Postcode | 备用 |  |
| Status | 状态 | 0是正常，1是禁用 |
| surname | 昵称 |  |

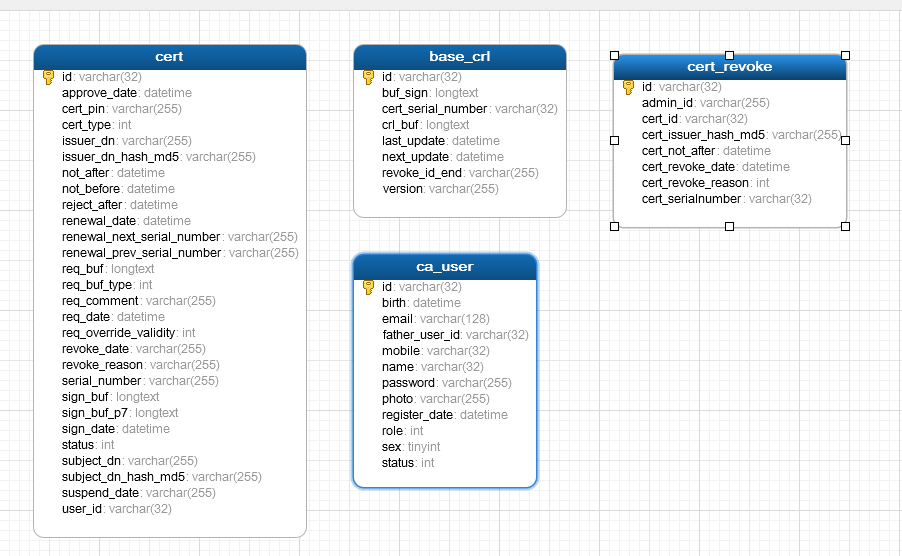
* + 1. 密钥表
       1. 表设计



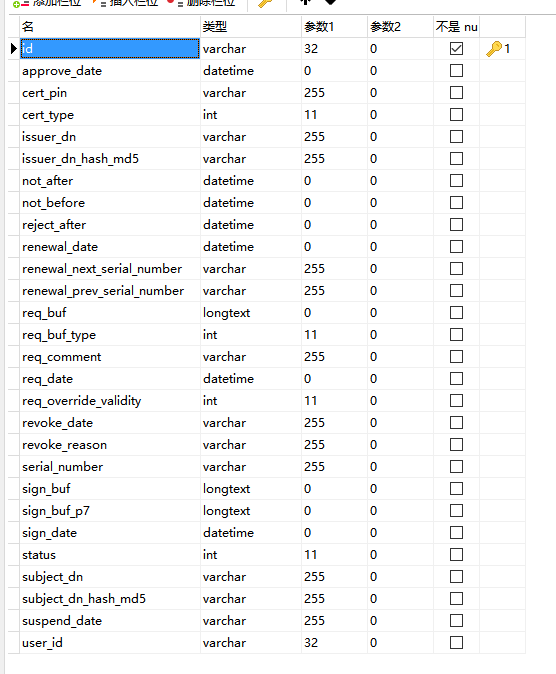
* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段名 | 描述 | 其他 |
| Id | id主键 |  |
| Create\_time | 创建时间 |  |
| Expired\_time | 过期时间 |  |
| Key\_size | 密钥大小 |  |
| Key\_status | 密钥状态 | 1.准备使用3.正在使用6.销毁 |
| Key\_type | 密钥类型 | 1.RSA,2.SM2 |
| Private\_key | 加密私钥 |  |
| Public\_key | 公钥 |  |
| Revoke\_time | 吊销时间 |  |
| Use\_time | 使用使用 |  |
| User\_id | 用户id |  |

1. CA系统数据库
   1. ER图



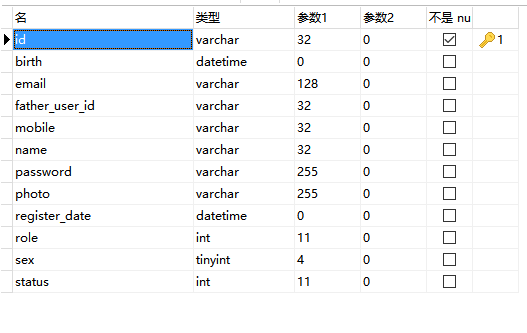
* 1. 数据表
     1. 证书表
        1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段名 | 字段描述 | 其他 |
| Id | id主键 |  |
| Approve\_date | 批准日期 |  |
| Cert\_pin | 证书密码 |  |
| Cert\_type | 证书类型 |  |
| Issuer\_dn | 颁发者主题 |  |
| Issuer\_dn\_hash\_md5 | 颁发者主题hash |  |
| Not\_after | 结束日期 |  |
| Not\_before | 开始日期 |  |
| Reject\_date | 拒绝日期 |  |
| Renew\_date | 更新日期 |  |
| Renew\_next\_serial\_number | 更新前序列号 |  |
| Renew\_prev\_serial\_number | 更新后序列号 |  |
| Req\_buf | Csr |  |
| Req\_buf\_type | Csr类型 | PKCS10:1 |
| Req\_comment | Csr注释 |  |
| Req\_date | 提交csr日期 |  |
| Req\_override\_validity | 过期时间 |  |
| Revoke\_date | 吊销时间 |  |
| Revoke\_reason | 吊销原因 |  |
| Serial\_number | 序列号 |  |
| Sign\_buf | 证书buf |  |
| Sign\_buf\_p7 | P7证书链 |  |
| Sign\_date | 签名日期 |  |
| Status | 状态 | ENROLL(0, "申请"), VALID(1, "有效"), REVOKE(2, "吊销"), SUSPEND(3, "挂起"); |

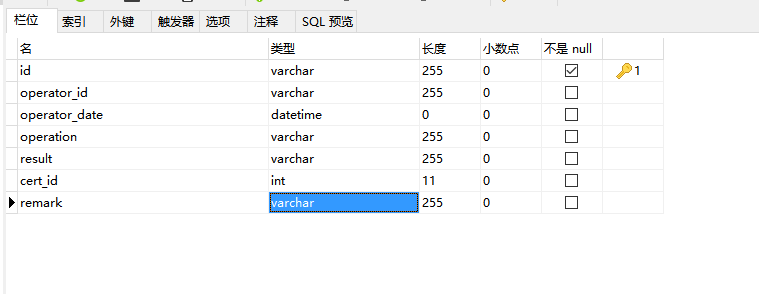
* + 1. 用户表
       1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段名 | 描述 | 其他 |
| Id | Id主键 |  |
| Birth | 出生日期 |  |
| Email | 邮箱 |  |
| Father\_user\_id | 父账户 |  |
| Mobile | 手机号 |  |
| Name | 名字 |  |
| Password | 密码 |  |
| Photo | 头像 |  |
| Register\_date | 注册日期 |  |
| Role | 角色 | USER(1, "普通用户"), RA\_ADMIN(2, "RA管理员"), CA\_ADMIN(3, "CA管理员"), OFFLINE\_ADMIN(4, "离线CA管理员"), SUPER(100, "超级管理员"); |
| Sex | 性别 |  |
| status | 状态 | VALID(1, "有效"), DISABLED(2, "禁用"); |

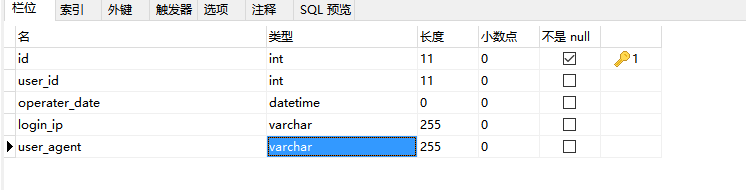
* + 1. 证书日志表
       1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段 | 描述 | 其他 |
| Id | 主键id |  |
| Operator\_id | 操作者id |  |
| Operator\_date | 操作日期 |  |
| Operation | 操作 |  |
| Result | 操作结果 |  |
| Cert\_id | 证书id |  |
| remark | 注释 |  |

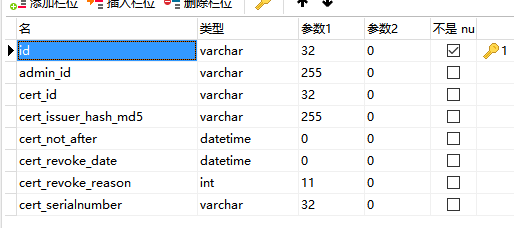
* + 1. 用户登陆日志
       1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段 | 描述 | 其他 |
| Id | Id |  |
| User\_id | 用户id |  |
| Operater\_date | 登陆日期 |  |
| Login\_ip | 登陆时ip |  |
| User\_agent | 用户http header头信息 |  |

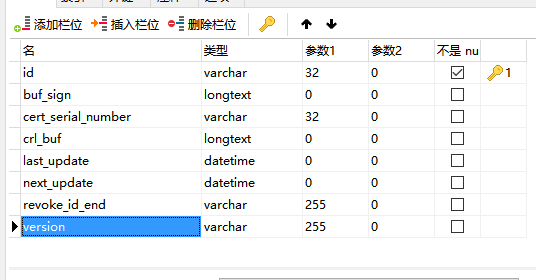
* + 1. 吊销证书记录表
       1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| Id | id主键 |  |
| Admin\_id | 吊销的管理员 |  |
| Cert\_id | 证书id |  |
| Cert\_issuer\_hash\_md5 | 证书颁发者hash |  |
| Cert\_not\_after | 证书开始日期 |  |
| Cert\_revoke\_date | 吊销日期 |  |
| Cert\_revoke\_reason | 吊销原因 | unused(1, "未知"),  keyCompromise(2, "密钥遭受损害"),  caCompromise(3, "CA 遭受损害"),  affiliationChanged(4, "从属关系变动"),  superseded(5, "证书被替代"),  cessationOfOperation(6, "停止使用"),  certificateHold(7, "证书暂停使用"); |

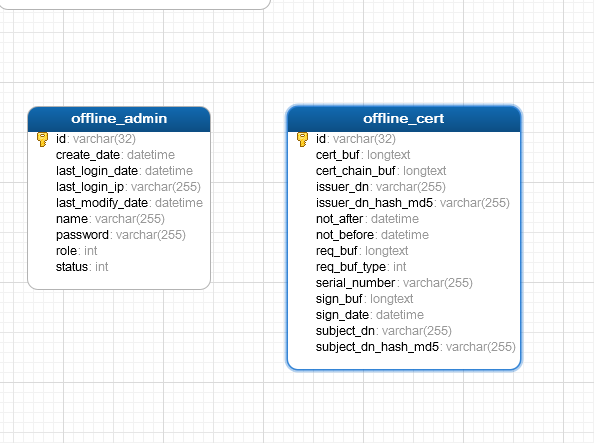
* + 1. CRL表
       1. 表设计



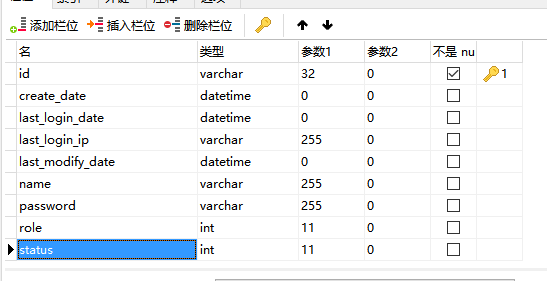
* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段 | 描述 | 其他 |
| Id | Id主键 |  |
| Buf\_sign | 签名数据 |  |
| Cert\_serial\_number | 证书序列号 |  |
| Crl\_buf | CRL base64编码字符 |  |
| Last\_update | 上一次生成 |  |
| Next\_update | 下一次生成 |  |
| Revoke\_id\_end | 列表中结束的证书id |  |
| version | 版本 |  |

1. 离线CA系统数据库
   1. ER图



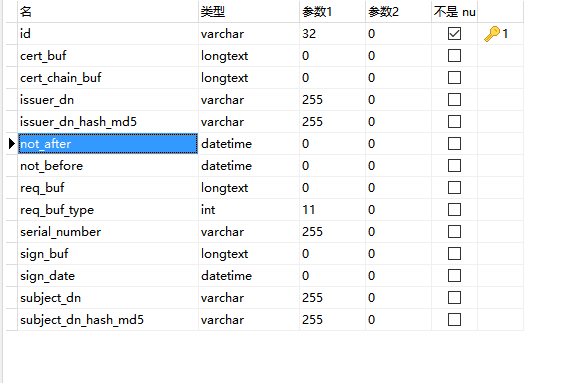
* 1. 数据表
     1. 管理员表
        1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段名 | 描述 | 其他 |
| Id | id主键 |  |
| Create\_date | 创建日期 |  |
| Last\_login\_date | 最后登陆日期 |  |
| Last\_login\_ip | 最后登陆ip |  |
| Last\_modify\_date | 最后修改日期 |  |
| Name | 登陆名 |  |
| Password | 密码 |  |
| Role | 角色 |  |
| status | 状态 |  |

* + 1. 离线Ca证书表
       1. 表设计



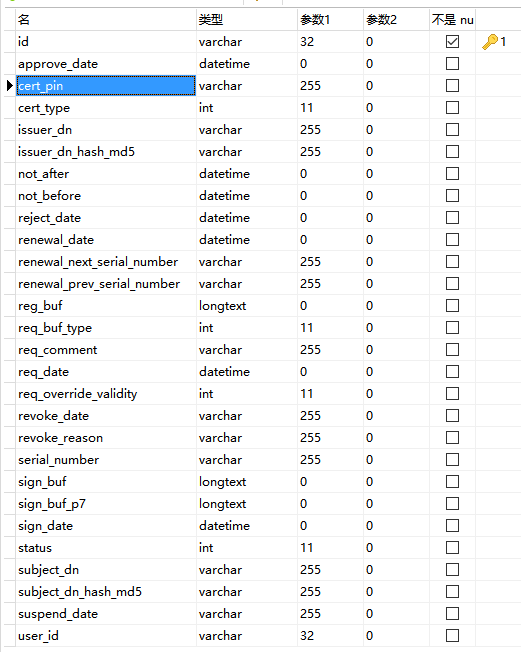
* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段 | 描述 | 其他 |
| Id | Id主键 |  |
| Cert\_buf | 证书base64编码字符串 |  |
| Cert\_chain\_buf | 证书链 |  |
| Issuer\_dn | 证书颁发者主题 |  |
| Issuer\_dn\_hash\_md5 | 证书颁发者主题hash |  |
| Not\_after | 结束日期 |  |
| Not\_before | 开始日期 |  |
| Req\_buf | Csr |  |
| Req\_buf\_type | Csr类型 |  |
| Serial\_number | 序列号 |  |
| Sign\_buf | 签名数据 |  |
| Sign\_date | 签名日期 |  |
| Subject\_dn | 主题 |  |
| Subject\_dn\_hash\_md5 | 主题hash |  |

1. RA系统数据库
   1. ER图



* 1. 数据表
     1. 证书申请记录表
        1. 表设计



* + - 1. 表描述

|  |  |  |
| --- | --- | --- |
| 字段名 | 字段描述 | 其他 |
| Id | id主键 |  |
| Approve\_date | 批准日期 |  |
| Cert\_pin | 证书密码 |  |
| Cert\_type | 证书类型 |  |
| Issuer\_dn | 颁发者主题 |  |
| Issuer\_dn\_hash\_md5 | 颁发者主题hash |  |
| Not\_after | 结束日期 |  |
| Not\_before | 开始日期 |  |
| Reject\_date | 拒绝日期 |  |
| Renew\_date | 更新日期 |  |
| Renew\_next\_serial\_number | 更新前序列号 |  |
| Renew\_prev\_serial\_number | 更新后序列号 |  |
| Req\_buf | Csr |  |
| Req\_buf\_type | Csr类型 | PKCS10:1 |
| Req\_comment | Csr注释 |  |
| Req\_date | 提交csr日期 |  |
| Req\_override\_validity | 过期时间 |  |
| Revoke\_date | 吊销时间 |  |
| Revoke\_reason | 吊销原因 |  |
| Serial\_number | 序列号 |  |
| Sign\_buf | 证书buf |  |
| Sign\_buf\_p7 | P7证书链 |  |
| Sign\_date | 签名日期 |  |
| Status | 状态 | ENROLL(0, "申请"), VALID(1, "有效"), REVOKE(2, "吊销"), SUSPEND(3, "挂起"); |

# 密码密钥配用

## 安全保护及管理保障需求分析

### 用户安全需求分析

1. 用户密码处理流程，用户输入密码后md5加密，密文通过网络发送



1. 系统安全架构，kms系统直接根CA通讯，CA与KMS同一个局域网内，CA通过防火墙，跟RA进行加密通讯，RA通过用户中心，使用https与用户进行安全通讯



### 密码产品安全防护需求分析

1. 加密机直接物理隔离，只有CA和KMS可以直接调用
2. CA与RA通讯使用加密的协议进行安全通讯
3. 用户跟用户中心使用https进行安全通讯认证

### 密码资源管理保障需求分析（可选）

1. CA负载服务对外只开放80端口，
2. RA服务对外只开放80端口
3. 用户密钥永远存在加密机中，出用户导出usbkey之外，其他功能都不可以导出

## 密码配用（密码产品或模块请分别参考对应模板）

### 密码算法配用

1. SM2算法
   1. 密钥长度256
   2. 主要用于通讯的加密和数字证书签名
2. RSA算法
   1. 密钥长度1024或者2048
   2. 主要用于通讯的加密和数字证书签名

### 密码算法工作模式配置

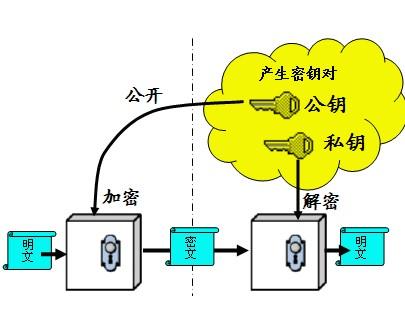
论证提出分组、杂凑等密码算法的工作模式，并以表格形式,给出密码算法、用途、工作模式、密码算法参数和所配置密码产品之间的对应关系。示例如下：

密码算法配置情况列表如下：

表1 配用密码算法

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 算法名称 | 使用密钥 | 用途 | 工作模式 | 所配置算法参数 |
| 非对称密码算法 | RSA密钥 | 对数据进行加密保护 | 模式 | [算法参数名称、新参数或已用参数，已配置于XX密码系统（装备）] |
| 非对称密码算法 | SM2密钥 | 对数据进行加密保护 | 模式 | [算法参数名称、新参数或已用参数，已配置于XX密码系统（装备）] |

### 工作模式(可选)



## 密钥配用

1. 数字证书签名，加密。封装
2. 安全通讯加密

### 密钥种类

1. SM2：256
2. RSA：1024,2048

### 密钥结构

1.用户私钥使用用户加密证书进行加密

### 密钥管理保障

1. 加密机：密钥不可导出
2. Usbkey：一次写入，不可篡改

## 密码协议

1、协议概述

公开密钥密码体制就是使用不同的加密密钥与解密密钥，是一种“由已知加密密钥推导出解密密钥在计算上是不可行的”密码体制。

在公开密钥密码体制中，加密密钥（即公开密钥）PK是公开信息，而解密密钥（即秘密密钥）SK是需要保密的。加密算法E和解密算法D也都是公开的。虽然解密密钥SK是由公开密钥PK决定的，但却不能根据PK计算出SK。

正是基于这种理论，1978年出现了著名的RSA算法，它通常是先生成一对RSA 密钥，其中之一是保密密钥，由用户保存；另一个为公开密钥，可对外公开，甚至可在网络服务器中注册。为提高保密强度，RSA密钥至少为500位长，一般推荐使用1024位。这就使加密的计算量很大。为减少计算量，在传送信息时，常采用传统加密方法与公开密钥加密方法相结合的方式，即信息采用改进的DES或IDEA对话密钥加密，然后使用RSA密钥加密对话密钥和信息摘要。对方收到信息后，用不同的密钥解密并可核对信息摘要。

RSA算法是第一个能同时用于加密和数字签名的算法，也易于理解和操作。RSA是被研究得最广泛的公钥算法，从提出到现今的三十多年里，经历了各种攻击的考验，逐渐为人们接受，普遍认为是目前最优秀的公钥方案之一。

SET(Secure Electronic Transaction)协议中要求CA采用2048bits长的密钥，其他实体使用1024比特的密钥。RSA密钥长度随着保密级别提高，增加很快。下表列出了对同一安全级别所对应的密钥长度。

1. 协议交互流程
   1. 选择两个大素数，p 和q ，计算出n=qp，n称为RSA算法的模数。p，q 必须保密，一般要求p，q为安全素数，n的长度大于1024bit ，这主要是因为RSA算法的安全性依赖于因子分解大数问题。
   2. 计算n的欧拉数
      1. φ(n)=(p-1)(q-1)
      2. φ(n)定义为不超过n并与n互质的数的个数。
   3. 然后随机选择加密密钥e，从[0,φ(n)-1]中选择一个与φ(n)互质的数e作为公开的加密指数。
   4. 最后，利用Euclid 算法计算解密密钥d, 满足de≡1(mod φ(n))。其中n和d也要互质。数e和n是公钥，d是私钥。两个素数p和q不再需要，应该丢弃，不要让任何人知道。
   5. 得到所需要公开密钥和秘密密钥：
      1. 公开密钥（即加密密钥） PK=（e，n）
      2. 秘密密钥（即解密密钥） SK=（d，n）

3、协议安全性分析

RSA的安全性依赖于大数分解，但是否等同于大数分解一直未能得到理论上的证明，因为没有证明破解RSA就一定需要作大数分解。假设存在一种无须分解大数的算法，那它肯定可以修改成为大数分解算法。 RSA 的一些变种算法已被证明等价于大数分解。不管怎样，分解n是最显然的攻击方法。人们已能分解多个十进制位的大素数。因此，模数n必须选大一些，因具体适用情况而定。

# 安全保密措施设计

描述产品保密安全措施的设计。如开机自检、噪声检验、开机身份认证、密钥及参数自检、密码要素存储保护、密码销毁、安全报警、物理防护等安全保密措施的具体实现。

## 密码资源保护机制

分段阐述算法参数、密钥等密码资源的存储和使用区域、存储加密保护、紧急销毁等内容。存储加密保护措施必须注明所保护对象的具体内容，即具体到参数、密钥或其他需要保护的资源，还应说明采取的保护方式，若基于算法实现，则应给出采用的算法、参数和密钥。紧急销毁措施需要明确写明紧急销毁的销毁操作方式、具体的销毁方式（如覆写随机数等方式），以及销毁的内容，并明确密码产品是否还能提供密码服务等。

## 故障诊断和处置机制

分段阐述算法自检、参数自检、密钥自检、噪声源自检、功能自检等措施实现的原理和方式，以及发现故障后中止密码服务运行、报警、上报应用系统等处置方式。

## 逻辑安全防护机制

分段阐述密码产品采取的访问控制、安全审计、操作系统安全增强、软件防篡改、远程（本地进程）监控、数据完整性保护、防网络入侵等方面的逻辑安全防护措施，以及密码服务、密码产品操作配置、远程密码管理、本地密码管理、密码资源注入等接口设置。

## 物理安全防护机制

分段阐述密码产品为抵御物理攻击采取的安全防护措施，如芯片防探测和抗能量分析、开盖销毁、机械锁、电磁泄漏防护等。

## 前向安全防护机制

公普产品根据实际需要，进行前向安全设计，并说明采用的机制或方式。

# 电子元器件国产化应用

描述军用电子元器件国产化思路与原则，明确品种国产比例、数量国产比例以及国产数费比，简述选用进口电子元器件的必要性、安全性和可保障性。

# 通用性设计

描述项目研制过程中将采用的可靠性、维修性、保障性、安全性、测试性、环境适应性等设计要求。包括研制过程的可靠性控制措施、硬件可靠性控制措施、软件可靠性控制措施、维修性控制措施、保障性控制措施等。

## 可靠性设计

包括可靠性模型、可靠性分配、可靠性预计、硬件可靠性设计和软件可靠性设计等方面内容。

## 维修性设计

包括维修性设计要求、维修性设计措施等方面内容。

## 保障性设计

说明系统运行和维护时所需的技术保障、环境保障等要求以及相应的保障方案，包括保障内容、保障手段、保障资源等。

## 安全性设计

说明为保证系统安全、稳定使用，进行的相关设计方案，如操作权限设计、越权告警设计、容错设计等。

## 测试性设计

说明为保证软件可测试性，进行的相关设计方案。

## 环境适应性设计

说明为满足环境适应性指标要求，进行的相关设计方案，如结构设计、热设计等方面的措施。

# 质量控制与标准化管理

## 质量控制要求

描述项目研制过程中将采用的质量控制措施。包括成立质量管理机构，制定质量保证大纲，制定技术设计质量控制、联试试用质量控制、技术资料质量保证等方面的内容。

有外协工作的，要制定外协质量控制措施。

## 标准化管理要求

描述项目研制过程中的标准化控制措施。包括标准化工作要求、标准化工作内容与实施方案、标准化工作组织、标准化管理等方面的内容。

# 关键技术分析

提出本项目研制中要解决的关键技术，并分析实现的可行性。

# 任务分工

列出主要参与研制单位（人员）及各单位（人员）的负责的工作。

# 进度安排

按照项目的阶段划分，分别描述各阶段的起止时间、应完成的工作。