(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 116647361 A
(43)申请公布日 2023.08.25

(21)申请号 202310360587.6

(22)申请日 2023.03.31

(71)申请人 山东华芯半导体有限公司
地址 250101 山东省济南市高新区经十东路汉峪金谷A2-3第16层1601室

(72)发明人 张忠国 王晓玉 秦法林 姜向阳 范宣荣

(74)专利代理机构 济南泉城专利商标事务所 37218
专利代理师 李桂存

(51)Int.Cl.
H04L 9/40(2022.01)
H04L 67/02(2022.01)
H04L 9/32(2006.01)
H04L 9/08(2006.01)

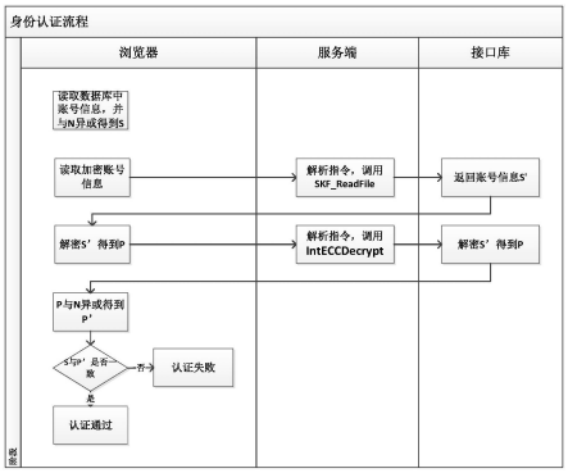
权利要求书1页 说明书3页 附图3页

(54)发明名称

一种基于智能密码钥匙的跨浏览器应用及身份认证方法和系统

(57)摘要

本发明提供了一种基于智能密码钥匙的跨浏览器应用及身份认证方法和系统，属于数据加密技术领域。包括以下步骤：密钥分发；密钥管理中心生成SM2加密密钥对，并调用导入ECC加密密钥对接口导入到加密设备中，并将该公钥导入到上层数据库；标签设置；设备初始化程序将加密设备上的标签信息，通过调用设置设备标签接口记录到设备中；根据COS设备的标签信息，从文件中依次读取标签对应的账号信息，将此账号信息与特定值异或后，加密存储到智能密码钥匙；浏览器端将从智能密码钥匙与数据库分别读取的账号信息对比以进行身份认证。本发明拓宽了智能密码钥匙的使用场景范围，提高了应用过程中的兼容性及安全性，能够更好地满足客户自定义的需求。



CN 116647361 A

1.一种基于智能密码钥匙的跨浏览器应用及身份认证方法,其特征在于,包括以下步骤:

步骤1:密钥分发;密钥管理中心生成SM2加密密钥对,并调用导入ECC加密密钥对接口导入到加密设备中,并将该公钥导入到上层数据库;

步骤2:标签设置;将加密设备上的标签信息,通过调用设置设备标签接口记录到设备中;

步骤3:设备批量初始化;将账号信息通过CSV文件存储,根据COS设备的标签信息,从文件中依次读取标签对应的账号信息,将此账号信息与特定值N异或后,将异或后的值经ECC外来公钥加密接口加密存储到智能密码钥匙;

步骤4:认证校验;浏览器端将从智能密码钥匙与数据库分别读取的账号信息对比以进行认证判断。

2.根据权利要求1所述的基于智能密码钥匙的跨浏览器应用及身份认证方法,其特征在于,所述浏览器端通过智能密码钥匙和数据库读取的账号信息对比进行身份认证具体方式如下:

首先从数据库读取账号信息S,所述账号信息包括账号和密码;然后从智能密码钥匙读取初始化时存储的账号信息S',将S'经智能密码钥匙内部私钥解密后得到P,将P与特定值N异或后得到P',最后将S与P'进行对比,若对比一致,则认证通过,如果不一致则认证不通过。

3.根据权利要求1所述的基于智能密码钥匙的跨浏览器应用及身份认证方法,其特征在于,所述智能密码钥匙接口中增加ECC内部私钥解密接口。

4.根据权利要求3所述的基于智能密码钥匙的跨浏览器应用及身份认证方法,其特征在于,所述增加ECC内部私钥解密接口功能如下:根据容器句柄获取到容器ID,从而找到内部存储的加密私钥,使用加密私钥对密文进行解密得到明文。

5.一种基于智能密码钥匙的跨浏览器应用及身份认证系统,其特征在于,可以使用权利要求1-4任一所述的方法,包括:浏览器端、服务端、数据库和接口库;

所述浏览器端采用Browser/Server架构,并通过采用jQuery库对AJAX通信模块封装到浏览器端;

所述服务端包含服务进程模块,用于获取浏览器端请求并根据请求调用接口库完成设备调用,再将处理结果反馈到浏览器端;

所述数据库用于存储账号信息和密码;所述接口库提供浏览器请求调用的接口。

6.根据权利要求5所述的基于智能密码钥匙的跨浏览器应用及身份认证系统,其特征在于,所述浏览器端和服务端采用GET和POST方法进行请求-响应。

一种基于智能密码钥匙的跨浏览器应用及身份认证方法和系统

技术领域

[0001] 本发明涉及一种基于智能密码钥匙的跨浏览器应用及身份认证方法和系统，属于数据加密技术领域。

背景技术

[0002] 随着互联网及业务系统应用的逐步深入，人们对网上信息及业务系统的安全防护问题日益突出，其中尤其以身份假冒、电子欺骗、数据篡改、信息窃密等攻击最为突出。为了保证网络平台和系统平台的安全运行，利用基于密码技术的身份认证、访问控制、授权管理、数据加解密、行为防抵赖等技术措施，构建应用安全保障体系，已经成为安全防护工作的当务之急。

[0003] SKF接口是国密标准中智能密码钥匙的C语言应用开发接口标准，以C接口库的方式提供给上层应用程序调用。为支持浏览器调用，现有方案使用Node.js、Activex/Com控件等方案实现。其中，Node.js方案底层为基于Chrome V8引擎的Java Script环境，依赖的FFI模块可实现JavaScript调用本地C接口动态链接库，但运行及部署该环境需要依赖Python和npm环境，并且还需要区别32/64位程序调用，终端用户在使用时需要安装基础环境，无法大规模部署。另一种Activex/com控件方案，由于谷歌等多家浏览器宣布已不再支持Activex/com控件，对智能密码钥匙的调用大大受限。

发明内容

[0004] 本发明目的是提供了一种基于智能密码钥匙的跨浏览器应用及身份认证方法和系统，支持HTTP、HTTPS两种协议传输拓宽了智能密码钥匙的使用场景范围，提高了应用过程中的兼容性及安全性，能够更好地满足客户自定义的需求。

[0005] 本发明为实现上述目的，通过以下技术方案实现：

[0006] 一种基于智能密码钥匙的跨浏览器应用及身份认证方法，包括以下步骤：

[0007] 步骤1：密钥分发；密钥管理中心生成SM2加密密钥对，并调用导入ECC加密密钥对接口导入到加密设备中，并将该公钥导入到上层数据库；

[0008] 步骤2：标签设置；将加密设备上的标签信息，通过调用设置设备标签接口记录到设备中；

[0009] 步骤3：设备批量初始化；将账号信息通过CSV文件存储，根据COS设备的标签信息，从文件中依次读取标签对应的账号信息，将此账号信息与特定值N异或后，将异或后的值经ECC外来公钥加密接口加密存储到智能密码钥匙；

[0010] 步骤4：认证校验；浏览器端将从智能密码钥匙与数据库分别读取的账号信息对比以进行认证判断。

[0011] 优选的，所述浏览器端通过智能密码钥匙和数据库读取的账号信息对比进行身份认证具体方式如下：

[0012] 首先从数据库读取账号信息S,所述账号信息包括账号和密码;然后从智能密码钥匙读取初始化时存储的账号信息S',将S'经智能密码钥匙内部私钥解密后得到P,将P与特定值N异或后得到P',最后将S与P'进行对比,若对比一致,则认证通过,如果不一致则认证不通过。

[0013] 优选的,所述智能密码钥匙接口中增加ECC内部私钥解密接口。

[0014] 优选的,所述增加ECC内部私钥解密接口功能如下:根据容器句柄获取到容器ID,从而找到内部存储的加密私钥,使用加密私钥对密文进行解密得到明文。

[0015] 一种基于智能密码钥匙的跨浏览器应用及身份认证系统,包括:浏览器端、服务端、数据库和接口库;

[0016] 所述浏览器端采用Browser/Server架构,并通过采用jQuery库对AJAX通信模块封装到浏览器端;

[0017] 所述服务端包含服务进程模块,用于获取浏览器端请求并根据请求调用接口库完成设备调用,再将处理结果反馈到浏览器端;

[0018] 所述数据库用于存储账号信息和密码,所述接口库提供浏览器请求调用的接口。

[0019] 所述浏览器端和服务端采用GET和POST方法进行请求-响应。

[0020] 本发明的优点在于:本发明不仅实现了智能密码钥匙的网络应用,实现了更加安全的网络身份认证方案。拓宽了智能密码钥匙的使用场景范围,提高了应用过程中的兼容性及安全性,能够更好地满足客户自定义的需求。

附图说明

[0021] 附图用来提供对本发明的进一步理解,并且构成说明书的一部分,与本发明的实施例一起用于解释本发明,并不构成对本发明的限制。

[0022] 图1为本发明流程结构示意图。

[0023] 图2为本发明HTTPS协议下流程结构示意图。

[0024] 图3为本发明系统架构示意图。

具体实施方式

[0025] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0026] 一种基于智能密码钥匙的跨浏览器应用及身份认证方法和系统采用Browser/Server架构,通过AJAX+HTTP SERVER机制实现浏览器网页与本地程序之间的双向通信。具体实现为:浏览器端(Browser)采用Html+Java Script语言实现,将数据请求转为AJAX通信,通过HTTP将数据请求发送到本地服务端(Server),本地服务端将来自浏览器网页的下行报文,转发给C接口库,并将来自动态库的上行报文,转发给浏览器。

[0027] 实施例

[0028] HTTP协议存在信息被窃听、信息被篡改、信息被劫持等风险,而HTTPS协议则可通过身份验证、信息加密、完整性校验等方式来解决这些问题。为了数据的安全性,越来越多

的系统从HTTP升级到HTTPS协议，因此针对HTTPS的支持至关重要。HTTPS在HTTP的基础上加入了SSL协议，SSL协议依靠证书来验证服务端身份。通过对HTTPS支持，不仅为浏览器端和服务端之间的通信加密，确保数据传输安全；还能有效验证对方身份，防止假冒，具体步骤如下：

[0029] 步骤1：基于OpenSSL安装环境，生成RSA密钥对，去除密码保护，生成服务端证书请求文件csr，CA机构对请求文件签名并生成证书；服务端将SSL初始化并载入SSL算法，载入服务端证书和私钥；

[0030] 步骤2：浏览器打开网页，基于ctx产生一个新的SSL，并将连接用户的socket加入到SSL，以建立SSL连接；与服务端建立SSL接，并将获取服务端证书及随机数R的消息通过服务端转发到接口库；

[0031] 步骤3：所述接口库接收消息并生成随机数R，将证书和随机数R通过服务端发送到浏览器；

[0032] 步骤4：浏览器读取数据库中CA公钥，通过CA公钥解密证书获取服务端公钥，进行服务端签名；

[0033] 步骤5：服务端接收到浏览器的签名请求后通过接口库对随机数进行哈希得到哈希值h，并调用接口库签名函数对h进行签名；

[0034] 步骤6：接口库通过获取的哈希值h产生签名值R+S，并通过服务端将签名值发送到浏览器；

[0035] 步骤7：浏览器接收到签名值后对随机数R和签名值R+S进行验签对随机数R经过固定运算(如异或等)得到R’，R’作为会话密钥对浏览器与服务端之间的通信进行Des加解密，实现数据传输的一次一密。结合到本专利服务端与浏览器端交互的指令均是加密之后传输的，实现更安全的数据传输。

[0036] 最后应说明的是：以上所述仅为本发明的优选实施例而已，并不用于限制本发明，尽管参照前述实施例对本发明进行了详细的说明，对于本领域的技术人员来说，其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换。凡在本发明的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。

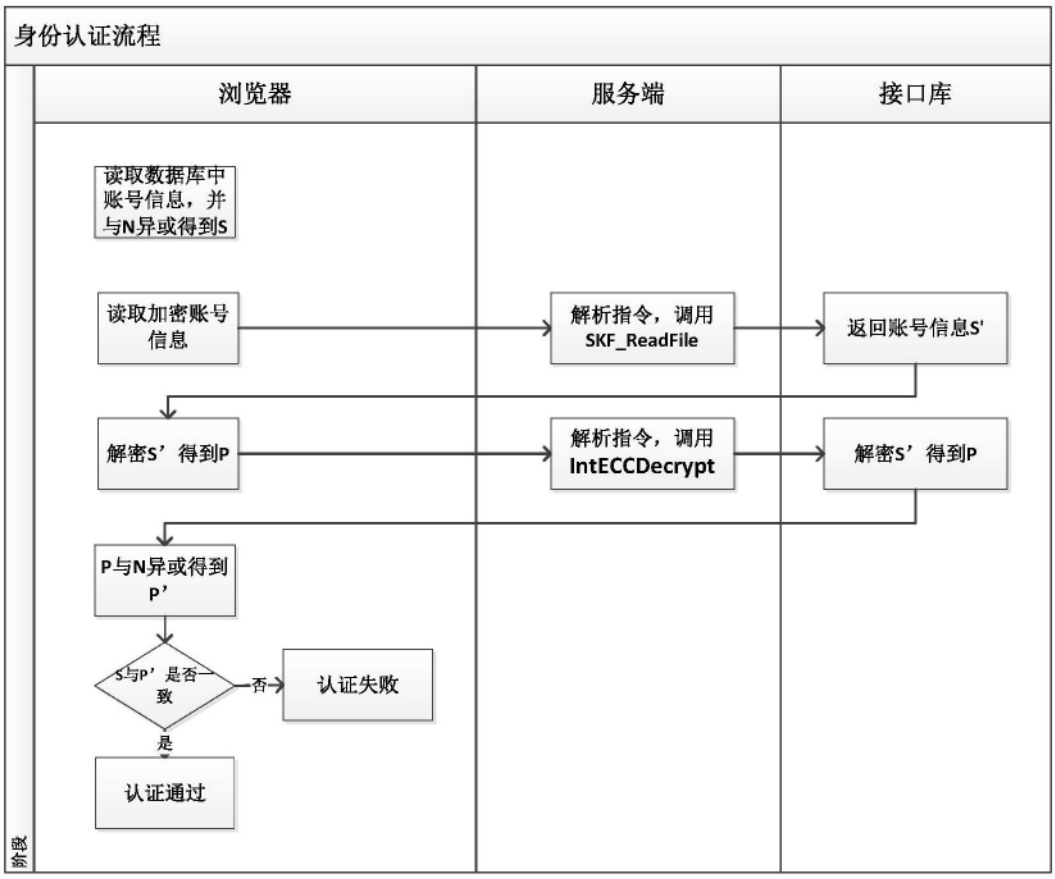

身份认证流程
浏览器
服务端
接口库
读取数据库中账号信息，并与N异或得到S
读取加密账号信息
解析指令，调用SKF_ReadFile
返回账号信息S'
解密S’得到P
解析指令，调用IntECCDecrypt
解密S’得到P
P与N异或得到P’
S与P’是否一致
否
认证失败
是
认证通过
阶段

图1

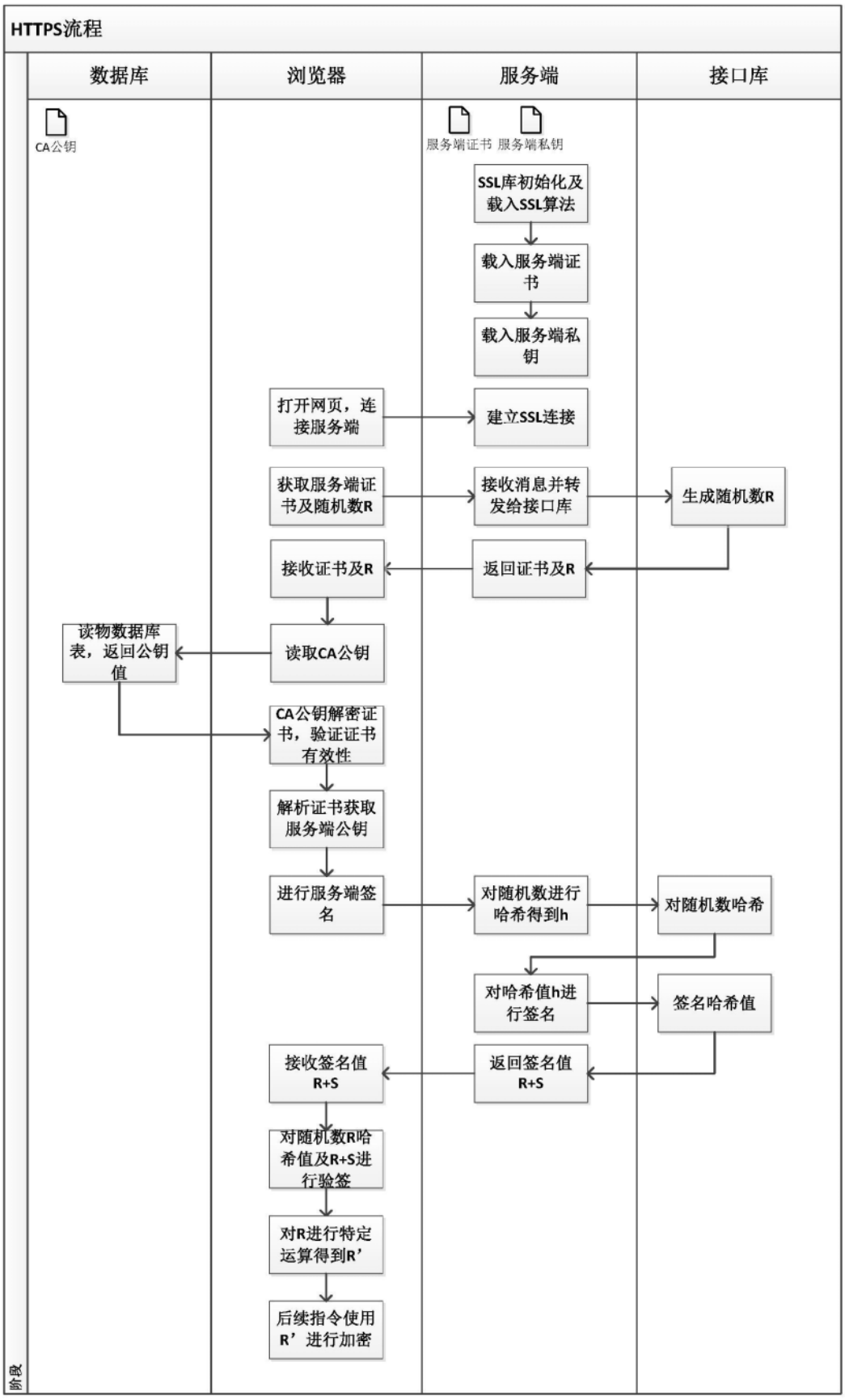

HTTPS流程
数据库
浏览器
服务端
接口库
CA公钥
服务端证书
服务端私钥
SSL库初始化及载入SSL算法
载入服务端证书
载入服务端私钥
打开网页，连接服务端
建立SSL连接
获取服务端证书及随机数R
接收消息并转发给接口库
生成随机数R
接收证书及R
返回证书及R
读物数据库表，返回公钥值
读取CA公钥
CA公钥解密证书，验证证书有效性
解析证书获取服务端公钥
进行服务端签名
对随机数进行哈希得到h
对随机数哈希
对哈希值h进行签名
签名哈希值
接收签名值R+S
返回签名值R+S
对随机数R哈希值及R+S进行验签
对R进行特定运算得到R'
后续指令使用R'进行加密
阶段

图2

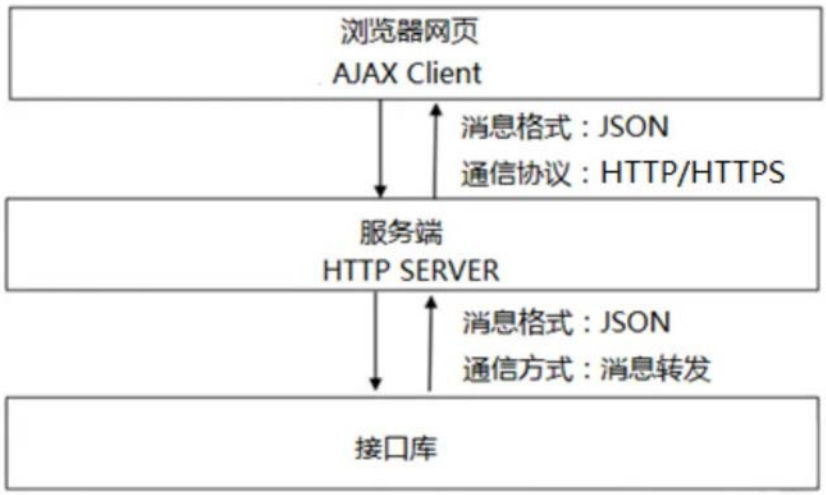

浏览器网页
AJAX Client
消息格式：JSON
通信协议：HTTP/HTTPS
服务端
HTTP SERVER
消息格式：JSON
通信方式：消息转发
接口库

图3