(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 112398649 A
(43)申请公布日 2021.02.23

(21)申请号 202011272804.9

(22)申请日 2020.11.13

(71)申请人 浪潮电子信息产业股份有限公司
地址 250101 山东省济南市高新区浪潮路1036号S05南五楼

(72)发明人 李寿斌

(74)专利代理机构 济南诚智商标专利事务所有限公司 37105
代理人 王敏

(51)Int.Cl.
H04L 9/08(2006.01)
H04L 9/06(2006.01)
H04L 9/32(2006.01)
H04L 29/06(2006.01)

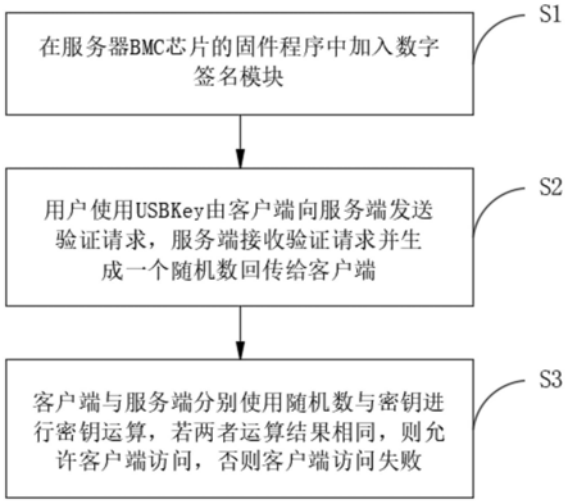
权利要求书1页 说明书4页 附图2页

(54)发明名称

一种利用USBKey和CA进行服务器加密的方法及系统

(57)摘要

本发明实施例公开了一种利用USBKey和CA进行服务器加密的方法及系统，涉及服务器安全技术领域。所述方法包括：在服务器BMC芯片的固件程序中加入有数字签名模块；用户使用USBKey由客户端向服务端发送验证请求，服务端接收验证请求并生成一个随机数回传给客户端；接下来客户端与服务端分别使用随机数与密钥进行密钥运算，若两者运算结果相同，则允许客户端访问，否则，客户端访问失败。本发明方案采用软件和硬件结合的方法，把数字证书保存在USBKey中，将USBKey插入计算机后，只有通过pin码验证后才能使用数字证书通过身份验证，退出USBKey后数字证书自动从计算机中清除，保护了访问用户身份的安全，提高了网络访问的安全性。



CN 112398649 A

1.一种利用USBKey和CA进行服务器加密的方法,其特征在于,包括以下步骤:

在服务器BMC芯片的固件程序中加入数字签名模块;

用户使用USBKey由客户端向服务端发送验证请求,服务端接收验证请求并生成一个随机数回传给客户端;

客户端与服务端分别使用随机数与密钥进行密钥运算,若两者运算结果相同,则允许客户端访问,否则,客户端访问失败。

2.根据权利要求1所述的一种利用USBKey和CA进行服务器加密的方法,其特征在于,所述数字签名模块嵌入在BMC芯片的RAM中,数字证书保存在所述USBKey中。

3.根据权利要求2所述的一种利用USBKey和CA进行服务器加密的方法,其特征在于,所述客户端使用随机数与密钥进行密钥运算,包括下述步骤:

客户端接收服务端回传的随机数;

将接收的随机数通过USB接口传递给USBKey;

USBKey使用所述随机数与存储在USBKey中的密钥进行MD5-HMAC运算,并将运算结果作为认证数据传递给服务器。

4.根据权利要求3所述的一种利用USBKey和CA进行服务器加密的方法,其特征在于,所述服务端使用随机数与密钥进行密钥运算,包括下述步骤:

服务端使用其生成的所述随机数与存储在服务器数据库中的客户密钥进行MD5-HMAC运算,得到运算结果;

将运算结果与客户端传递的认证数据进行比对。

5.一种利用USBKey和CA进行服务器加密的系统,包括客户端和服务端,其特征在于,所述服务端BMC芯片的固件程序中加入有数字签名模块;

所述客户端通过USBKey向服务端发送验证请求,并使用服务端回传的随机数与密钥进行密钥运算;

所述服务端接收客户端发送的验证请求,然后生成随机数回传给客户端,并使用随机数与密钥进行密钥运算。

6.根据权利要求5所述的一种利用USBKey和CA进行服务器加密的系统,其特征在于,所述客户端使用随机数与密钥进行密钥运算,包括:客户端接收服务端回传的随机数,并将随机数通过USB接口提供给USBKey;USBkey使用随机数与存储在USBkey中的密钥进行MD5-HMAC运算,并将运算结果作为认证数据传递给服务器。

7.根据权利要求6所述的一种利用USBKey和CA进行服务器加密的系统,其特征在于,所述服务端使用随机数与密钥进行密钥运算,包括:服务端使用其生成的随机数与存储在服务器数据库中的客户密钥进行MD5-HMAC运算,得到运算结果,并将运算结果与客户端传递的认证数据进行比对。

8.根据权利要求7所述的一种利用USBKey和CA进行服务器加密的系统,其特征在于,所述服务器将运算结果与客户端传递的认证数据进行比对,若两者运算结果相同,则允许客户端访问,否则,客户端访问失败。

一种利用USBKey和CA进行服务器加密的方法及系统

技术领域

[0001] 本发明实施例涉及服务器安全技术领域，具体来说涉及一种利用USBKey和CA进行服务器加密的方法及系统。

背景技术

[0002] 随着电子商务的迅速发展，信息安全已成为当前的焦点问题之一，尤其是网上支付和网络银行对信息安全的要求显得更为突出。USBKey采用双钥(公钥)加密的认证模式，是一种USB接口的硬件设备，多用于保护网上银行的资金安全。USBKey内置有单片机或智能卡芯片，有一定的存储空间，可以存储用户的私钥及数字证书。利用USBKey内置的公钥算法实现对用户身份的认证，由于用户私钥保存在密码锁中，理论上使用其它方式无法读取，因此保证了用户认证的安全性。随着PKI技术的日趋成熟，许多应用中开始使用数字证书进行身份认证与数字加密。数字证书是由权威公正的第三方机构即CA中心签发，以数字证书为核心的加密技术，可以对网络上传输的信息进行加密和解密、数字签名和签名验证，确保网上传递信息的机密性、完整性以及交易实体身份的真实性，签名信息的不可否认性保障了网络应用的安全性。

[0003] BMC主要用于采集单个服务器上的各种信息，同时提供给上层运维网管软件。其主要有两种手段，一种是BMC会提供各种各样的接口供上层网管查询，如web、命令行等人机接口，SNMP、IPMI、Restful等机机接口；另一种是主动上报，当检测到有故障产生时，BMC可通过SNMP trap消息、SMTP邮件消息、Redfish http json报文等手段上报给上层网管软件的服务端，以便运维人员及时识别处理故障。当前，国产服务器BMC登录是基于主板上BMC芯片密码的存储鉴权认证的，存在加密强度低及通过技术手段容易被篡改的问题。

发明内容

[0004] 本发明实施例提供了一种利用USBKey和CA进行服务器加密的方法及系统，使用USBKey远程访问BMC web界面，并利用USBKey内置的公钥算法实现对用户身份的认证，保证了用户认证的安全性。

[0005] 为实现上述目的，本发明公开了如下技术方案：

[0006] 本发明一方面提供一种利用USBKey和CA进行服务器加密的方法，所述方法包括以下步骤：

[0007] 在服务器BMC芯片的固件程序中加入数字签名模块；

[0008] 用户使用USBKey由客户端向服务端发送验证请求，服务端接收验证请求并生成一个随机数回传给客户端；

[0009] 客户端与服务端分别使用随机数与密钥进行密钥运算，若两者运算结果相同，则允许客户端访问，否则，客户端访问失败。

[0010] 基于上述方案，进一步的，所述数字签名模块嵌入在BMC芯片的RAM中，数字证书保存在所述USBKey中。

[0011] 进一步的,所述客户端使用随机数与密钥进行密钥运算,包括下述步骤:

[0012] 客户端接收服务端回传的随机数;

[0013] 将接收的随机数通过USB接口传递给USBKey;

[0014] USBKey使用所述随机数与存储在USBKey中的密钥进行MD5-HMAC运算,并将运算结果作为认证数据传递给服务器。

[0015] 进一步的,所述服务端使用随机数与密钥进行密钥运算,包括下述步骤:

[0016] 服务端使用其生成的所述随机数与存储在服务器数据库中的客户密钥进行MD5-HMAC运算,得到运算结果;

[0017] 将运算结果与客户端传递的认证数据进行比对。

[0018] 本发明另一方面提供一种利用USBKey和CA进行服务器加密的系统,系统包括客户端和服务端,所述服务端BMC芯片的固件程序中加入有数字签名模块;

[0019] 所述客户端通过USBKey向服务端发送验证请求,并使用服务端回传的随机数与密钥进行密钥运算;

[0020] 所述服务端接收客户端发送的验证请求,然后生成随机数回传给客户端,并使用随机数与密钥进行密钥运算。

[0021] 进一步的,所述客户端使用随机数与密钥进行密钥运算,具体包括:客户端接收服务端回传的随机数,并将随机数通过USB接口提供给USBKey;USBkey使用随机数与存储在USBkey中的密钥进行MD5-HMAC运算,并将运算结果作为认证数据传递给服务器。

[0022] 进一步的,所述服务端使用随机数与密钥进行密钥运算,具体包括:服务端使用其生成的随机数与存储在服务器数据库中的客户密钥进行MD5-HMAC运算,得到运算结果,并将运算结果与客户端传递的认证数据进行比对。

[0023] 基于上述系统,进一步的,所述服务器将运算结果与客户端传递的认证数据进行比对,若两者运算结果相同,则允许客户端访问,否则,客户端访问失败。

[0024] 发明内容中提供的效果仅仅是实施例的效果,而不是发明所有的全部效果,上述技术方案中的一个技术方案具有如下优点或有益效果:

[0025] 本申请实施例提供的一种利用USBKey和CA进行服务器加密的方法,在服务器BMC芯片的固件程序中加入有数字签名模块;用户使用USBKey由客户端向服务端发送验证请求,服务端接收验证请求并生成一个随机数回传给客户端;接下来客户端与服务端分别使用随机数与密钥进行密钥运算,若两者运算结果相同,则允许客户端访问,否则,客户端访问失败。本实施例方案采用软件和硬件结合的方法,把数字证书保存在USBKey中,将USBKey插入计算机后,只有通过pin码验证后才能使用数字证书通过身份验证,退出USBKey后数字证书自动从计算机中清除,提高了网络访问的安全性。另外,密钥运算分别在USB Key硬件和服务器中运行,不出现在客户端内存中,也不在网络上传输,由于MD5-HMAC算法是一个不可逆的算法,知道密钥和运算用随机数就可以得到运算结果,而知道随机数和运算结果却无法计算出密钥,从而保护了密钥的安全,进而保护了用户身份的安全。

[0026] 本申请实施例提供的一种利用USBKey和CA进行服务器加密的系统,能够实现上述的利用USBKey和CA进行服务器加密的方法,并取得上文所述的技术效果。

附图说明

[0027] 此处的附图被并入说明书中并构成说明书的一部分,示出了符合本申请的实施例,并与说明书一起用于解释本申请的原理。

[0028] 图1为本申请实施例提供的一种利用USBKey和CA进行服务器加密的方法流程示意图;

[0029] 图2为本申请实施例的服务器BMC芯片结构示意图;

[0030] 图3为本申请实施例提供的一种利用USBKey和CA进行服务器加密的系统工作交互示意图。

具体实施方式

[0031] 为使本技术领域的人员更好地理解本发明中的技术方案,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,都应当属于本发明保护的范围。

[0032] 为了方便对本发明技术方案的理解,下面对本发明中涉及的缩略词和关键术语予以解释和说明。

[0033] CA:Certificate Authority,认证中心;

[0034] PKI:Public Key Infrastructure,公开密钥基础架构;

[0035] BMC:Baseboard Manager Controller,基板控制管理器。

[0036] 图1示出了本发明实施例一种利用USBKey和CA进行服务器加密的方法流程示意图。

[0037] 参照图1,本实施例的方法,包括以下步骤:

[0038] S1、在服务器BMC芯片的固件程序中加入数字签名模块;

[0039] 具体的,国产服务器的带外管理软件运行在服务器一款单独的ARM芯片上,这个ARM芯片就是BMC软件的CPU,同时芯片外围配置有自己的RAM及Flash等器件,本实施例的服务器BMC芯片的结构设计如图2所示。

[0040] S2、用户使用USBKey由客户端向服务端发送验证请求,服务端接收验证请求并生成一个随机数回传给客户端;

[0041] 具体的,所述数字签名模块嵌入在BMC芯片的RAM中,数字证书保存在所述USBKey中,用户使用USB Key远程访问BMC web界面,实现对服务器的部署操作。

[0042] S3、客户端与服务端分别使用随机数与密钥进行密钥运算,若两者运算结果相同,则允许客户端访问,否则,客户端访问失败;

[0043] 具体的,在本步骤中,所述客户端使用随机数与密钥进行密钥运算,包括下述步骤:

[0044] 客户端接收服务端回传的随机数;

[0045] 将接收的随机数通过USB接口传递给USBKey;

[0046] USBKey使用所述随机数与存储在USBKey中的密钥进行MD5-HMAC运算,并将运算结果作为认证数据传递给服务器。

[0047] 进一步的,所述服务端使用随机数与密钥进行密钥运算,包括下述步骤:

[0048] 服务端使用其生成的所述随机数与存储在服务器数据库中的客户密钥进行MD5-HMAC运算,得到运算结果;

[0049] 将运算结果与客户端传递的认证数据进行比对。

[0050] 如果服务器的运算结果与客户端传回的响应结果相同,则认为客户端是合法用户,允许客户端访问;否则,记录客户端登录信息日志。本实施例方法密钥运算分别在USBKey硬件和服务器中运行,由于MD5-HMAC算法是不可逆的算法,保护了密钥的安全,同时保护了用户身份的安全。

[0051] 图3示出了本发明实施例一种利用USBKey和CA进行服务器加密的系统工作交互示意图。

[0052] 参照图3,本实施例的系统包括客户端和服务端,所述服务端BMC芯片的固件程序中加入有数字签名模块;

[0053] 所述客户端通过USBKey向服务端发送验证请求,并使用服务端回传的随机数与密钥进行密钥运算;

[0054] 所述服务端接收客户端发送的验证请求,然后生成随机数回传给客户端,并使用随机数与密钥进行密钥运算。

[0055] 进一步的,如图中所示,所述客户端使用随机数与密钥进行密钥运算,具体包括:客户端接收服务端回传的随机数,并将随机数通过USB接口提供给USBKey;USBkey使用随机数与存储在USBkey中的密钥进行MD5-HMAC运算,并将运算结果作为认证数据传递给服务器。

[0056] 进一步的,所述服务端使用随机数与密钥进行密钥运算,具体包括:服务端使用其生成的随机数与存储在服务器数据库中的客户密钥进行MD5-HMAC运算,得到运算结果,并将运算结果与客户端传递的认证数据进行比对。

[0057] 如上所述的系统中,所述服务器将运算结果与客户端传递的认证数据进行比对,若两者运算结果相同,则鉴权通过允许客户端访问;否则,客户端访问失败,并记录客户端IP相关登录信息日志。

[0058] 本申请实施例提供的一种利用USBKey和CA进行服务器加密的系统中未详述的内容,可参照上述实施例中提供利用USBKey和CA进行服务器加密的方法,在此不再赘述。

[0059] 以上所述仅为本发明的较佳实施例而已,并不用以限定本发明,对于本技术领域的技术人员来说,在不脱离本发明原理的前提下所作的任何修改、改进和等同替换等,均包含在本发明的保护范围内。

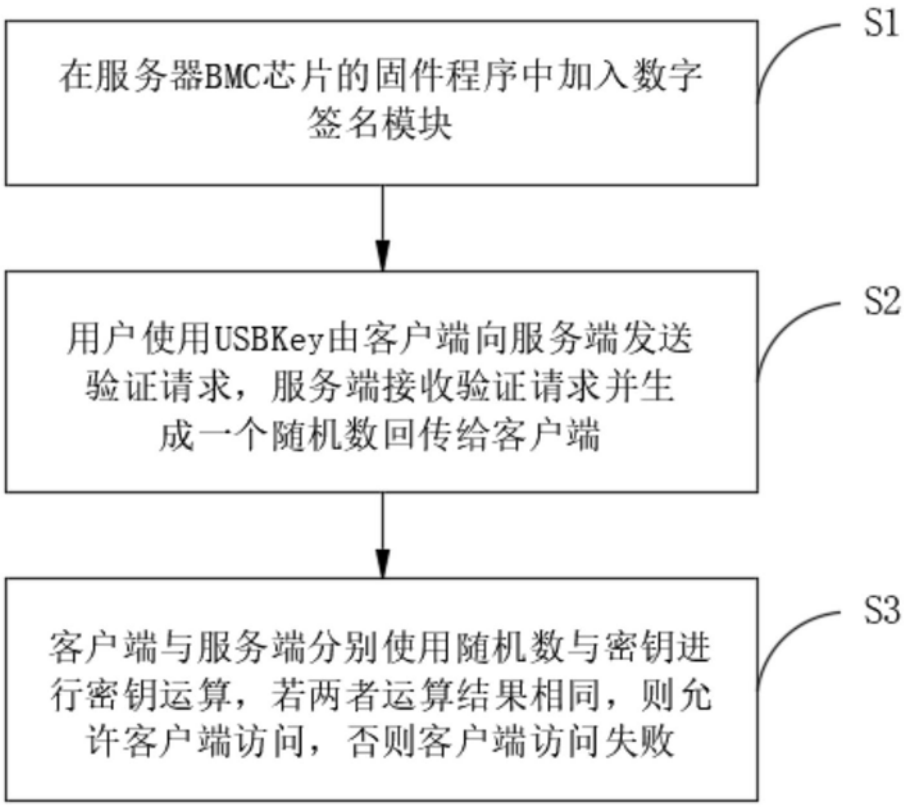

在服务器BMC芯片的固件程序中加入数字签名模块
S1
用户使用USBKey由客户端向服务端发送验证请求，服务端接收验证请求并生成一个随机数回传给客户端
S2
客户端与服务端分别使用随机数与密钥进行密钥运算，若两者运算结果相同，则允许客户端访问，否则客户端访问失败
S3

图1

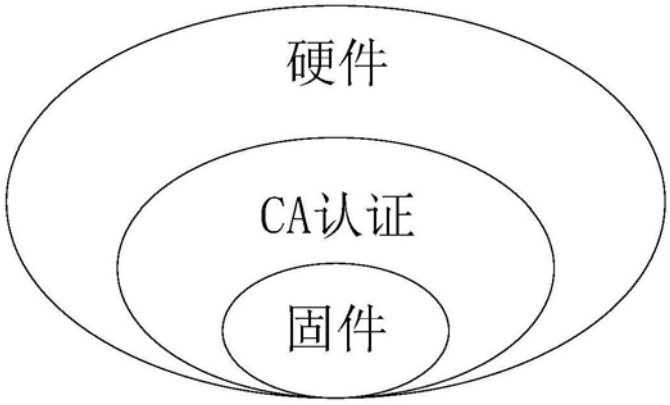

硬件
CA认证
固件

图2

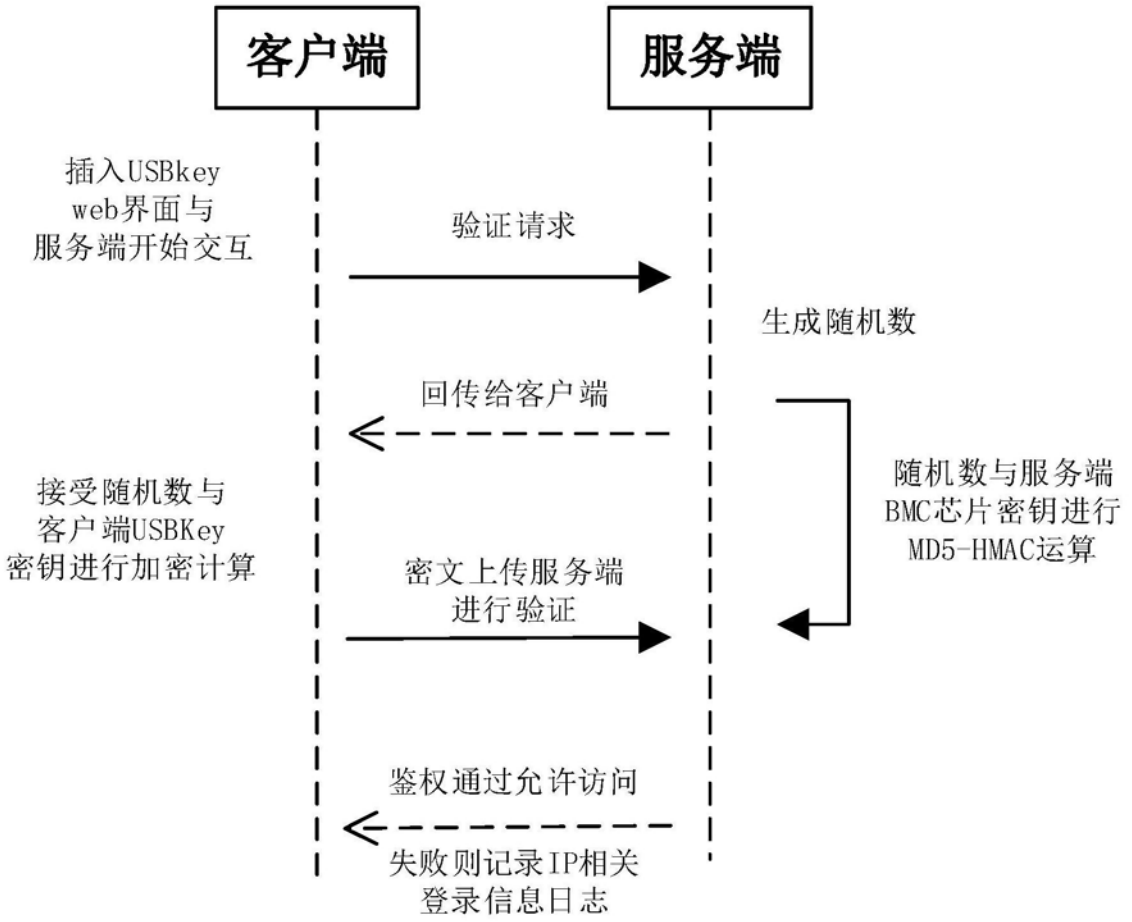

客户端
服务端
插入USBkey
web界面与
服务端开始交互
验证请求
生成随机数
回传给客户端
随机数与服务端
BMC芯片密钥进行
MD5-HMAC运算
接受随机数与
客户端USBKey
密钥进行加密计算
密文上传服务端
进行验证
鉴权通过允许访问
失败则记录IP相关
登录信息日志

图3