(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 106576046 B
(45)授权公告日 2020.09.15

(21)申请号 201580034810.1

(22)申请日 2015.06.22

(65)同一申请的已公布的文献号
申请公布号 CN 106576046 A

(43)申请公布日 2017.04.19

(30)优先权数据
62/017,045 2014.06.25 US
14/704,963 2015.05.05 US

(85)PCT国际申请进入国家阶段日
2016.12.26

(86)PCT国际申请的申请数据
PCT/US2015/036937 2015.06.22

(87)PCT国际申请的公布数据
WO2015/200196 EN 2015.12.30

(73)专利权人 美国亚德诺半导体公司
地址 美国马萨诸塞州

(72)发明人 J·沃尔什
J·R·瓦尔拉本斯泰因

(74)专利代理机构 中国贸促会专利商标事务所有限公司 11038
代理人 邹丹

(51)Int.Cl.
H04L 9/32(2006.01)

审查员 郭风顺

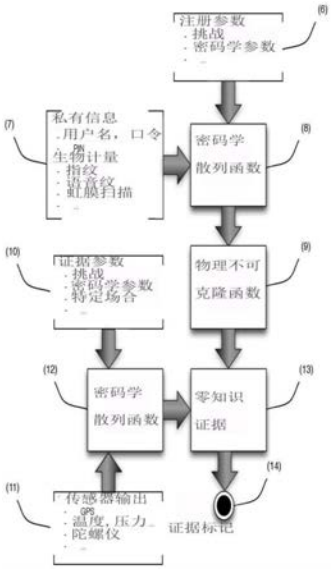
权利要求书2页 说明书12页 附图2页

(54)发明名称

将元数据与硬件固有属性绑定的系统和设备

(57)摘要

通过获得认证相关的元数据以及将其与诸如物理不可克隆函数的与信任根有关的信息组合而将诸如从生物计量传感器的输出得到的信息的元数据与硬件固有属性绑定的系统、设备和方法。元数据可以从诸如生物计量传感器的传感器取得，信任根可以是物理不可克隆函数，元数据和信任根信息的组合可以采用散列函数，并且来自该散列过程的输出可用作信任根的输入。组合的信息能够用于交互式或非交互式认证。



CN 106576046 B

1.一种认证设备,所述认证设备被配置成将认证相关的元数据与硬件固有属性绑定,所述认证设备包括:

a)物理不可克隆函数PUF电路,具有PUF电路输入和PUF电路输出,所述PUF电路被构造成响应于接收到输入值而生成表征所述PUF电路和所述输入值的输出值;以及

b)处理器,其连接到所述PUF电路,所述处理器被配置成:

i)将与所述认证设备相关联的设备注册参数和与所述设备相关联的认证相关的元数据组合以产生绑定值;

ii)将所述绑定值传送到所述PUF电路并接收来自所述PUF电路的输出值,其中所述输出值是根据所述PUF电路、被组合以产生所述绑定值的所述设备注册参数和所述认证相关的元数据来生成的;以及

iii)创建认证证据标记作为所述输出值的函数。

2.如权利要求1所述的设备,其中所述处理器还被配置成,响应于来自外部验证实体的认证请求,执行认证零知识证据。

3.如权利要求2所述的设备,其中所述处理器被配置成执行的认证零知识证据是其非交互式的且包括包含非敏感元数据的特定场合。

4.如权利要求2所述的设备,其中所述处理器被配置成执行的认证零知识证据是交互式的。

5.如权利要求1所述的设备,其中所述处理器被配置成利用密码散列函数将设备注册参数与认证相关的元数据组合。

6.如权利要求1所述的设备,其中所述处理器被配置成利用迭代密码散列函数将设备注册参数与认证相关的元数据组合。

7.如权利要求1所述的设备,其中所述处理器被配置成借助算法来与设备注册参数组合的所述认证相关的元数据仅包含敏感元数据。

8.如权利要求1所述的设备,其中所述处理器被配置成创建的认证证据标记是公共身份标记。

9.如权利要求1所述的设备,其中所述处理器被配置成与设备注册参数组合的所述认证相关的元数据仅包含敏感元数据,并且其中所述处理器被配置成创建的认证证据标记是公共身份标记。

10.如权利要求1所述的设备,其中所述处理器被配置成与认证相关的元数据组合的所述设备注册参数包括与密码数学架构有关的值。

11.如权利要求10所述的设备,其中所述处理器被进一步配置成执行椭圆曲线密码学。

12.如权利要求11所述的设备,其中与密码数学架构有关的值包含挑战值、椭圆曲线基点和模数。

13.如权利要求1所述的设备,其中所述处理器被进一步配置成,响应于来自外部验证实体的认证请求而执行认证零知识证据,而不将任何敏感元数据传达给外部验证实体。

14.如权利要求1所述的设备,其中所述PUF被构造成,响应于特定挑战值的输入而生成表征所述PUF和所述特定挑战值的输出值。

15.如权利要求14所述的设备,其中所述处理器被进一步配置成,响应于来自外部验证实体的认证请求而执行认证零知识证据,而不将任何敏感元数据传达给所述外部验证实

体。

16.如权利要求14所述的设备，其中所述处理器被配置成组合包含挑战值的设备注册参数。

17.如权利要求15所述的设备，其中所述处理器被配置成组合包含挑战值的设备注册参数。

18.如权利要求14所述的设备，其中所述处理器还被配置为，响应于来自外部验证实体的认证请求，执行认证零知识证据。

19.如权利要求14所述的设备，其中所述处理器被配置为利用密码散列函数将设备注册参数与认证相关的元数据组合。

20.如权利要求14所述的设备，其中所述处理器被配置成创建的认证证据标记是公共身份标记。

将元数据与硬件固有属性绑定的系统和设备

[0001] 相关申请的交叉引用

[0002] 本申请要求于2014年6月25日提交的序号为62/017,045的美国临时专利申请以及于2015年5月5日提交的序号为14/704,963的美国专利申请的优先权的利于,这两个申请中的每一个申请均通过引用合并于此。

背景技术

[0003] 认证协议通常依赖于实体所持有的私有信息从而确立身份。在传统系统中,私有数据可以包含用户名和口令对、个人认证号(PIN)或密码密钥。多因素认证协议通常需要两种或更多种标识信息,诸如实体所知的信息(例如,用户名和口令)、实体所具有的某事物(例如,智能卡或标记)以及代表了实体为何物的信息(例如,指纹)。

[0004] 元数据包括与认证中所涉及到的实体的身份或状态有关的辅助信息。元数据的示例包括生物计量数据、传感器输出、全球定位数据、口令或PIN以及可用于构造实体的身份或状态的特性的类似的辅助信息。生物计量数据包括足够独特以便用作身份证据的用户的物理特性(例如,指纹、视网膜、虹膜、语音和静脉模式)的测量。

[0005] 然而,依赖于传感器输出的系统会易受伪造传感器输出的攻击;虽然生物计量系统使用有力的属性进行认证,但是它们会面临与敏感生物计量数据的暴露和/或丢失有关的挑战。由于传感器将测得的物理特性变换成二进制串,所以没有进一步措施,系统不能将从传感器返回的串与没有传感器的对手所提供的串区分开,二进制串由计算机系统存储(注册),然后与随后在认证请求时传感器所生成的二进制串比较。因此,例如,对手会试图观察特定用户的生物计量传感器的输出且通过将暗中获得的生物计量数据提供给系统来“克隆”用户。对手可以类似地试图通过读取系统中所存储的生物计量数据来克隆用户。进一步,由于在生物计量系统中所使用的特征按定义趋向于基本不变,所以用户的生物计量数据的损害无法以如下方式来弥补:能够简单地更改丢失的用户口令。

[0006] 个体硬件设备所独有和固有的特性(例如,电线电阻、初始存储器状态、CPU指令定时)也可以被提取且用作认证协议的部分。这样的主要例子是物理不可克隆函数(PUF)。PUF函数f(c)将输入域(或挑战)c映射到输出范围(或应答)r,其中映射是基于计算f(•)的设备所独有的特性来定义的。f(•)的电路或硬件描述可以在所有设备上都相同,而从域到范围的映射将基于执行计算f(•)的电路的特定的硬件设备而是独特的。

[0007] 在各种设备认证方案中,物理不可克隆函数(PUF)已经被使用,而使得每个设备都具有本质上与设备所链接的独特身份。Rührmair等人(“Modeling attacks on Physical Unclonable Functions,”Proceedings of the17th ACM conference on Computer and communications security,CCS'10,第237-249页,ACM,2010)定义了PUF设备的三个不同的类:

[0008] • 弱PUF通常仅用于取得私钥。挑战空间会受限制,并且假设应答空间永不显露。典型的构造包括SRAM PUF(Holcomb等人,“Initial SRAM State as a Fingerprint and Source of True Random Numbers for RFID Tages,”In Proceedings of the

Conference on RFID Security,2007),Butterfly PUF(Kumar等人,"Extended abstract: The Butterfly PUF Protecting IP on Every FPGA,"IEEE International Workshop on Hardware-Oriented Security and Trust,第67-70页,2008),Arbiter PUF(Lee等人,"A technique to build a secret key in integrated circuits for identification and authentication applications,"IEEE Symposium on VLSI Circuits:Digest of Technical Papers,第176-179页,2004),Ring Oscillator PUF(Suh等人,"Physical Unclonable Functions for Device Authentication and Secrete Key Generation," Proceedings of the 44th annual Design Automation Conference,DAC'07,第9-14页,ACM,2007),以及Coating PUF(Tuyls等人,"Read-Proof Hardware from Protective Coatings,"Proceedings of the 8th international conference on Cryptographic Hardware and Embedded Systems,CHES'06,第369-383页,Springer,2006)PUF。

[0009] • 强PUF假设为(i)物理上不可能克隆,(ii)不可能在合理的时间内采集挑战应答对的完整集合(通常要花费数周量级),以及(iii)难以预测对随机挑战的应答。例如,Rührmair("Applications of High-Capacity Crossbar Memories in Cryptography,"IEEE Trans.Nanotechnol.,卷10,3期:489-498,2011)所描述的超高信息内容(SHIC)PUF可被视为强PUF。

[0010] • 受控PUF满足了强PUF的所有标准,并且另外实现了能够计算更先进功能的辅助控制单元以便以密码学方式增强协议。

[0011] PUF输出是有噪声的,因为尽管评估相同的输入其也略微变化。这通常是利用模糊提取来解决的,这是一种开发用来消除生物计量测量中的噪声的方法。(参见Juels等人,"Fuzzy Commitment Scheme",Proceedings of the 6th ACM conference on Computer and Communications Security,CCS'99,第28-36页,ACM,1999)。模糊提取可以部分地在具有PUF的设备内采用,例如在辅助控制单元内,使得输出对于固定输入是恒定的。模糊提取(或逆模糊提取)可以例如采用如Juels等人所描述的"安全略图"来存储待重构的敏感值 p_i^{priv} 和用于恢复 p_i^{priv} 的助手串 helper_i。输入串O的安全略图SS可以例如定义为 $SS(O;p_i^{\mathrm{priv}})=O\oplus ECC(p_i^{\mathrm{priv}})$,其中ECC是能够校正t个错误的长度为n的二进制(n,k,2t+1)纠错码,$p_i^{\mathrm{priv}}\leftarrow\{0,1\}^k$ 是k比特值。然后,给定助手串 helper_i 以及在O的最大汉明距离t内的输入O',对于纠错码ECC和O'使用译码方案D来复制原始值V为

[0012] $$D(\mathrm{helper}_i\oplus O')=D(O\oplus ECC(p_i^{\mathrm{priv}})\oplus O')=p_i^{\mathrm{priv}}$$。

[0013] 与设备d绑定的物理不可克隆函数 $\mathrm{P}_d:\{0,1\}^{\kappa_1}\rightarrow\{0,1\}^{\kappa_2}$ 优选地呈现以下性质:

[0014] 1.不可克隆性:$Pr[\mathrm{dist}(y,x)\leq t|x\leftarrow U_{\kappa_1},y\leftarrow \mathrm{P}(x),z\leftarrow \mathrm{P}']\leq \epsilon_1$,利用克隆PUF P'复制PUF P而使得它们的输出分布是t统计接近的概率小于某充分小的 $\in_1$。

[0015] 2.不可预测性:期望的是对手无法以大于可忽略概率(至少没有实际访问设备)预测设备对挑战c的PUF应答r,并且助手数据没有向对手显露出任何关于PUF应答的信息。假设全部实体都必然是概率多项式时间(PPT),即,仅能够高效地关于全局安全参数λ(其是指

相关参数中的位数)执行需要多项式多次操作的计算，$\mathbf{Adv}_{\mathscr{A}}^{\mathrm{PUF-PRED}}(\kappa_2) := Pr[r = r']$，以$\kappa_2$表示对手$\mathscr{A}$猜测(Guess)PUF P对挑战(Challenge)c的正确应答r的概率优选地可忽略。这可以通过例如对手和PUF P设备之间的游戏来评估：$\{0,1\}^{\kappa_1} \mapsto \{0,1\}^{\kappa_2}$，其将来自长度为$\kappa_1$的挑战空间$\mathscr{C}_{\mathrm{P}}$的输入串映射到长度为$\kappa_2$的应答空间$\mathscr{R}_{\mathrm{P}}$，其中$\lambda$是协议的安全参数，统一地给定为$1^{\lambda}$。

[0016] PUF-PRED:PUF预测游戏

[0017]

	对手$\mathscr{A}$		PUF 设备 P	
(1)	$c_i \in \bar{\mathscr{C}}_{\mathrm{P}} \subset \mathscr{C}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	$\longrightarrow$		
		$\longleftarrow$	$r_i = \mathrm{P}(c_i) \in \bar{\mathscr{R}}_{\mathrm{P}}$ $\bar{\mathscr{R}}_{\mathrm{P}} \subset \mathscr{R}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	(2)
(3)	Challenge $\mathbf{c} \notin \bar{\mathscr{C}}_{\mathrm{P}}$	$\longrightarrow$		
(4)	$c'_i \in \bar{\mathscr{C}}'_{\mathrm{P}} \subset \mathscr{C}_{\mathrm{P}}$, $\mathbf{c} \notin \bar{\mathscr{C}}'_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	$\longrightarrow$		
		$\longleftarrow$	$r'_i = \mathrm{P}(c'_i) \in \bar{\mathscr{R}}'_{\mathrm{P}}$ $\bar{\mathscr{R}}'_{\mathrm{P}} \subset \mathscr{R}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	(5)
(6)	Guess $\mathbf{r}' \overset{?}{\leftarrow} \mathrm{P}(\mathbf{c})$	$\longrightarrow$		

[0018] 游戏如下进行：

[0019] 1.对手$\mathscr{A}$向PUF设备P发出多项式多个(关于安全参数λ) $c_i \in \bar{\mathscr{C}}_{\mathrm{P}}$，其中挑战集合$\bar{\mathscr{C}}_{\mathrm{P}}$是整个挑战空间$\mathscr{C}_{\mathrm{P}}$的真子集。

[0020] 2.PUF设备P将应答$\{r_i | r_i \leftarrow \mathrm{P}(c_i)\}$返回给$\mathscr{A}$。

[0021] 3.对手$\mathscr{A}$最终输出不是挑战查询原始集合$\bar{\mathscr{C}}_{\mathrm{P}}$内的挑战c。不允许对手关于所提交的挑战c来查询PUF设备P。

[0022] 4.对手$\mathscr{A}$会再次向PUF设备P发出多项式多次挑战新集合$c'_i \in \bar{\mathscr{C}}'_{\mathrm{P}}$。不允许对手关于所提交的挑战c来查询PUF设备P。

[0023] 5.PUF设备P返回应答$\{r'_i | r'_i \leftarrow \mathrm{P}(c'_i)\}$给$\mathscr{A}$。

[0024] 6.对手$\mathscr{A}$最终输出P对所提交的挑战c的应答的猜测r'。

[0025] 仅当猜测r'等于P对$\mathscr{A}$所提交的挑战c的实际应答$r \leftarrow \mathrm{P}(c)$时，对手才赢得游戏。(如所提到的，PUF的输出有噪声且将在任何固定输入上有略微变化，因此相对于模糊提取

器的输出来视为相等(例如,Dodis等人,“Fuzzy Extractors:How to Generate Strong Keys from Biometrics and Other Noisy Data,”SIAMJ.Comput.,卷38,no.1:97-139,2008))。

[0026] 3.鲁棒性:$Pr[\mathrm{dist}(y,z) > t | x \leftarrow U_{\kappa_1}, y \leftarrow \mathrm{P}(x), z \leftarrow \mathrm{P}(x)] \leq \epsilon_2$,即,固定PUF P在同一输入x上产生t距离应答的概率小于某充分小的$\in_2$。

[0027] 4.不可区分性:PUF设备的输出(通常是模糊提取器输出)优选地计算上不能与同长度ℓ的随机串区分开,使得PPT对手$\mathcal{A}$的优势$\mathbf{Adv}_{\mathcal{A}}^{\mathbf{PUF\text{-}IND}}(\ell)$至多可忽略地大于1/2。PUF的不可区分性可以例如通过这样的游戏来评估:其中对手$\mathcal{A}$被要求区分用于PUF P的模糊提取器的输出r和同长度ℓ的随机选定的串$s \in \{0,1\}^{\ell}$。

[0028] PUF-IND:PUF不可区分性游戏

[0029]

	对手$\mathcal{A}$		PUF 设备 P	
(1)	$c_i \in \mathcal{C}\mathcal{H} \subset \mathcal{C}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	$\longrightarrow$	$R_i \leftarrow \mathrm{rand} \in \{0,1\}^{\lambda}$ $H_i \leftarrow ECC(R_i) \oplus \mathrm{P}(c)$	
		$\longleftarrow$	$H_i \in \bar{\mathcal{R}}_{\mathrm{P}} \subset \mathcal{R}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	(2)
(3)	$c_i \in \bar{\mathcal{C}}\mathcal{H} \subset \mathcal{C}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	$\longrightarrow$		
		$\longleftarrow$	$R_i \in \bar{\mathcal{R}}_{\mathrm{P}} \subset \mathcal{R}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	(4)
(5)	Challenge $\mathbf{c} \notin \bar{\mathcal{C}}\mathcal{H}$	$\longrightarrow$	$b \in \{0,1\}$	
		$\longleftarrow$	$b(s \in \{0,1\}^{\ell}) +$ $(1-b)(R_i)$, $R_i = D(H_i \oplus \mathrm{P}(\mathbf{c}))$	(6)
(7)	$c_i' \in \mathcal{C}\mathcal{H} \subset \mathcal{C}_{\mathrm{P}}$, $\mathbf{c} \neq c_i'$, $0 \le i \le \mathrm{poly}(\lambda)$	$\longrightarrow$		
		$\longleftarrow$	$R_i' \in \bar{\mathcal{R}}_{\mathrm{P}} \subset \mathcal{R}_{\mathrm{P}}$, $0 \le i \le \mathrm{poly}(\lambda)$	(8)
(9)	Guess $\mathbf{b}' \stackrel{?}{=} b$	$\longrightarrow$		

[0030]

[0031] 该游戏进行如下:

[0032] 1.对手$\mathcal{A}$对于任何挑战$c_i \in \mathcal{C}_{\mathrm{P}}$执行注册阶段。

[0033] 2.PUF设备返回相应的助手串H_i,其利用PUF的输出P(c)遮挡纠错后的敏感值ECC

(R_i)。将该挑战-助手对集合(c_i,H_i)标记为$\mathscr{CH}$。

[0034] 3.对手$\mathscr{A}$现在请求对于任何$c_i \in \mathscr{CH}$的PUF应答$r_i = P(c_i)$。将该步骤中的请求挑战集合标记为$\bar{\mathscr{C}}\mathscr{H}$。

[0035] 4.对于所有的请求$c_i \in \bar{\mathscr{C}}\mathscr{H}$,PUF设备返回集合$\{r_i | r_i \leftarrow P(c_i)\}$。

[0036] 5.对手$\mathscr{A}$选择挑战$c \notin \bar{\mathscr{C}}\mathscr{H}$,使得对于c,$\mathscr{A}$具有$H_i$而不是$R_i$。PUF设备随机地、均匀地选择位b∈{0,1}。

[0037] 6.如果b=0,则$\mathscr{A}$被给定$R_i = D(H_i \oplus P(c))$。否则,如果b=1,则$\mathscr{A}$被给定随机串$s \in \{0,1\}^l$。

[0038] 7.允许对手$\mathscr{A}$对于$c'_i \in \mathscr{CH}$查询PUF设备,只要不是$c'_i = c$。

[0039] 8.对于所有的请求$c'_i \neq c$,PUF设备返回集合$\{r'_i | r'_i \leftarrow P(c'_i)\}$。

[0040] 9.对手输出猜测位b',并且当b'=b时成功。

[0041] PUF的相关的评估由Hori等人的"Quantitative and Statistical Performance Evaluation of Arbiter Physical Unclonable Functions on FPGA," 2010International Conference on Reconfigurable Computing and FPGAs(ReConFig),第298-303页,2010;Maiti,A systematic Approach to Design an Efficient Physical Unclonable Function,dissertation,Virginia Tech,2012,以及其它。

[0042] 各种认证方案使用知识的零知识证据,其是一种假设给定陈述为真、同时除此事实之外不显露任何信息的方法。零知识证据是两方之间的交互:即,希望确立陈述的有效性的证明者$\mathscr{P}$,以及必须确信陈述为真的验证者$\mathscr{V}$。应当以压倒性的概率使验证者确信,真实的陈述确实是真的。利用知识的零知识证据,验证者可以不使用来自先前证据的消息来说服新的一方陈述的有效性,并且消息仅显露出单比特信息:证明者$\mathscr{P}$是否拥有秘密。存在两种一般类的零知识证据:交互式零知识证据,其中一系列消息在证明者$\mathscr{P}$与验证者$\mathscr{V}$之间交换,以及非交互式零知识证据,其中证明者传达单条消息$\mathscr{M}$,而不与$\mathscr{V}$交互,而使$\mathscr{V}$确信$\mathscr{P}$拥有秘密。许多(交互式)零知识证据系统要求多次迭代来确立陈述的有效性。也即,每个交互可以某概率而成功,即使证明者没有拥有秘密(或者陈述为假)。因此,如果当陈述为假时成功的概率是p,则协议运行n次直至$1-(p)^n$充分接近1。

[0043] Ivanov等人的美国专利8,577,091和Armstrong等人的美国专利8,566,579描述了认证系统,其中需要固有的硬件特性(例如,PUF输出)以及人类生物计量来成功地完成认证,但是均没有提供用于必然地将PUF与认证链接的方法或处置非敏感传感器输出的方法。

[0044] Frikken等人("Robust Authentication using Physically Unclonable Funcitons,"Information Security,Lecture Notes in Computer Science的卷5735,第262-277页,Springer,2009)教导了一种将元数据(例如,PIN)组合到PUF的输入的方法,但是没有提供到任意元数据(例如,生物计量数据)或非敏感元数据(例如,温度,压力)的扩展。

[0045] Rust(编辑)在“D1.1Report on use case and architecture requirements,”, Holistic Approaches for Integrity of ICT-Systems(2013)提到了将生物计量特征与基于单元的PUF合并的思想,但是没有对于实现的手段进行详述。

[0046] Erhart等人的美国专利申请公开20110002461描述了一种通过采用PUF来认证传感器输出的方法,其提取物理传感器硬件的独特特性。但是该方法没有直接将传感器的输出与硬件的认证链接,还要求敏感的生物计量传感器输出离开设备。

发明内容

[0047] 设备的固有身份是通过生成注册标记或公钥来构造的,该注册标记或公钥基于设备所独有的固有特性,诸如物理不可克隆函数(PUF)。认证系统使用设备的注册标记或公钥来验证设备的真实性,优选第通过零知识证据。敏感元数据优选第也并入注册标记或公钥中,这可以通过将元数据与硬件固有(例如,PUF)数据组合的散列函数的算法工具来实现。认证可以是交互式的或非交互式的。

附图说明

[0048] 图1是示出了在本发明的实施方案中的元数据绑定操作流的功能图;以及

[0049] 图2是示出了提供用于任意传感器输出的零知识证据生成的实施方案的功能图。

具体实施方式

[0050] 虽然本发明一般应用于元数据,但是描述了使用生物计量传感器的示范性的实施方案。还参考使用椭圆曲线密码学(包含相关联的术语和归约)的实施方案的实施例描述了本发明,但是发明构思和其中的教导同样应用于其它各种密码学方案,诸如采用不同问题的密码学方案,比如离散对数或因数化,并且本发明不限于本文所描述的可以利用或借助本发明所采用的各种附加特征。

[0051] 为了构造设备的固有身份,生成设备的身份的公共表示(此处称为注册标记或公钥)。在该设备注册过程中,从设备采集密码学注册标记。用于注册和认证的椭圆曲线数学架构可以使用,但是其它适合的架构(例如,离散对数架构,在这方面美国专利8,918,647通过引用方式合并于此)将提供相同的功能。响应于服务器的挑战查询c_i(或多个查询),从PUF设备采集密码学注册标记(或标记系列){(c_i,P_i,$A_i$mod p)}。在一个实施方案中,设备随机地从空间$\{0,1\}^{\lambda}$中均匀地选择私钥p_i^{priv},其中λ是安全参数(例如,模数p中的位数)并且计算$A_i = p_i^{priv} \cdot G \bmod p$作为设备的公钥,其中G是椭圆曲线在$\mathbb{F}_p$上的阶p的基点。使用算法1,设备能够任选地利用PUF而无需与服务器交互来执行本地注册协议。这允许每个PUF电路生成本地公钥p_i^{pub}。这在引导自举更复杂的密钥设置算法时可能是有用的;但是在密钥设置在设备内内部地(而不是在一组不同设备之间外部地)执行的情况下引导自举可能是不必要的)。在另一实施方案中,设备可以接受来自外部服务器的椭圆曲线参数和挑战,服务器随后存储设备的挑战和助手数据。

[0052]

算法 1 注册

for Device d **do**
Select finite field $\mathbb{F}_p$ of order p
Select E, an elliptic curve over $\mathbb{F}_p$
Find $G \in E/\mathbb{F}_p$, a base point of order q
$c_i \in \mathbb{F}_p$, a group element
$x = H(c_i, G, p, q)$
$O = PUF(x)$
$\text{helper}_i = O \oplus \text{ECC}(p_i^{\text{priv}} \bmod q)$
$p_i^{\text{pub}} = A_i = p_i^{\text{priv}} \cdot G \bmod p$
Store $\{p_i^{\text{pub}}, c_i, \text{helper}_i\}$
end for

[0053] 在采用椭圆曲线密码学的实施方案的实施例中，下面的算法2和3能够任选地用来允许PUF使能的设备来存储和取回敏感值，而不将任何敏感信息存储在非易失性存储器内。算法2说明了使用PUF存储敏感值 $\boldsymbol{p_i^{priv}}$，算法3说明了 $\boldsymbol{p_i^{priv}}$ 的动态再生成。挑战c_i和助手数据helper_i可以是公共的，因为两者均没有显露出任何有关敏感值 $\boldsymbol{p_i^{priv}}$ 的内容。这些值可以本地地存储到设备，或者在外部存储到不同的设备中。如果在外部存储，则值将在运行算法之前被提供给设备。虽然本实施例使用了通过异或⊕对 $\boldsymbol{p_i^{priv}}$ 加密，但是 $\boldsymbol{p_i^{priv}}$ 还可以被用作其它加密算法(例如，AES)的密钥来使能存储和取回任意大小的值。

[0054]

算法 2 PUF-存储

Goal: Store value p_i^{priv}
for PUF Device d **do**
$x = H(c_i, G, p, q)$
$O = PUF(x)$
$\text{helper}_i = O \oplus \text{ECC}(p_i^{\text{priv}})$
Write $\{c_i, \text{helper}_i\}$
end for

[0055]

算法 3 PUF-取回

Goal: Retrieve value p_i^{priv}
for PUF Device d **do**
Read $\{c_i, \text{helper}_i\}$
$x \leftarrow H(c_i, G, p, q)$
$O' = PUF(x)$
$p_i^{\text{priv}} \leftarrow D(\text{helper}_i \oplus O') = D((ECC(p_i^{\text{priv}}) \oplus O) \oplus O')$
end for

[0056] 每当O和O’是t接近时，纠错码ECC能够传递给译码算法D，译码算法将恢复敏感值 $\boldsymbol{p_i^{priv}}$。

[0057] 在椭圆曲线实施方案中，在接收到来自设备的认证请求时，服务器能够进行Chaum等人(“An Improved Protocl for Demonstrating Posession of Discrete Logarithms and some Generalizations,”,Proceedings of the 6th annual international conference on Theory and applications of cryptographic techniques,EUROCRYPT’

87,第127-141,Springer,1988)具有设备d的零知识证据协议的椭圆曲线变体从而认证设备d,如算法4所示。

算法 4 认证

for PUF Device d **do**
  Server $s \leftarrow$ request
end for
for Server s **do**
  Device $d \leftarrow \{c_i, \text{helper}_i, G, p, q, N\}$ where N is a nonce
end for
for PUF Device d **do**
  $x \leftarrow H(c_i, G, p, q)$
  $p_i^{\text{priv}} \leftarrow D(\text{helper}_i \oplus \text{PUF}(x))$
  $A_i = p_i^{\text{priv}} \cdot G \mod p$
[0058] $r \leftarrow \texttt{random} \in \mathbb{F}_p$, a random group element
  $B \leftarrow r \cdot G \mod p$
  $c \leftarrow \text{Hash}(G, B, A_i, N)$
  $m \leftarrow r + c \cdot p_i^{\text{priv}} \mod q$
  Server $s \leftarrow \{B, m\}$
end for
for Server s **do**
  $c' \leftarrow \text{Hash}(G, B, A_i, N)$
  $P \leftarrow m \cdot G - c' \cdot A_i \mod p$
  Device $d \leftarrow \begin{cases} \text{accept} & : P = B \\ \text{deny} & : P \neq B \end{cases}$
end for

[0059] 算法4中的验证服务器为设备提供了特定于当前证据的特定场合值,从而防止窃听的对手使用来自有效设备的先前的证据来成功地完成认证协议且伪装为设备。非交互式零知识证据免除了该通信要求,并且允许在不与验证端点进行交互的情况下完成证据。实现非交互式构造要求证明设备以如下方式代表验证者生成特定场合:防止证明的终端设备操纵证据。

[0060] 一种构造非交互式零知识证据的方法是设备构造特定场合N为N←H(A||τ),其中A是设备的公钥,H(•)是密码学散列函数,τ是时间戳,x||y表示x和y的级联。时间戳确保通过证明设备所构造的先前的证据在未来不会被对手重放,而散列函数确保证明设备无法以对手的方式操纵特定场合。对时间戳的信赖实质上比对全局同步时钟的信赖更不繁重。也即,时间戳无需确切地匹配在到达证明者时的当前时间戳,这消除了影响证据的网络延时的可能。相反,验证的端点核验时间戳是合理地当前的(例如,第二粒度级)且单调地增加以防止重放攻击。对于PUF使能设备的示范性的非交互式零知识证据描述于算法5中。

算法 5 非交互式认证

for PUF Device d **do**

$p_i^{\text{priv}} \leftarrow \text{PUF-Retrieve}(c_i, \text{helper}_i)$

$A_i = p_i^{\text{priv}} \cdot G \mod p$

$r \leftarrow \text{random} \in \mathbb{F}_p$, a random group element

$B \leftarrow r \cdot G \mod p$

$N \leftarrow \text{Hash}(A \| \tau)$ where τ is the current timestamp

$c \leftarrow \text{Hash}(G, B, A_i, N)$

$m \leftarrow r + c \cdot p_i^{\text{priv}} \mod q$

[0061] Server $s \leftarrow \{B, m, \tau\}$

end for

for Server s **do**

$A_i = p_i^{\text{priv}} \cdot G \mod p$ (public key stored from device enrollment)

Verify τ is reasonably current (e.g., $\tau = \text{current time} - \epsilon$)

$N \leftarrow \text{Hash}(A \| \tau)$

$c' \leftarrow \text{Hash}(G, B, A_i, N)$

$P \leftarrow m \cdot G - c' \cdot A \mod p$

$$\text{Device } d \leftarrow \begin{cases} \text{accept} & : P = B \\ \text{deny} & : P \neq B \end{cases}$$

end for

[0062] 元数据绑定

[0063] 元数据绑定指的是将辅助元数据并入认证过程的过程。元数据是认证协议所应依赖的任意辅助信息。也即，没有正确的元数据，认证应当是失败的。元数据可表征为敏感的或非敏感的，其中敏感元数据不应离开设备(例如，口令，PIN，生物计量)，而非敏感元数据可以离开设备(例如，关于温度、压力的传感器输出)。

[0064] 敏感元数据优选地并入在注册期间所创建的公共身份标记中。例如，当没有提供敏感元数据时，设备注册输出仅表征设备的公共身份。然而，当在注册期间提供了敏感元数据(例如，生物计量，PIN等)时，公共身份表征设备和敏感元数据。本发明的一个实施方案永不要求敏感元数据离开设备，因为完成零知识证据协议，而无需验证者对敏感元数据具有访问权。

[0065] 优选地，非敏感元数据不并入注册过程，使得从注册输出的公共身份不依赖于非敏感元数据(例如，对于温度、压力等的传感器输出)。相反，非敏感元数据优选地并入零知识证据协议中，使得设备和/或用户真实性的证据仅在对应的非敏感元数据也提供给验证者的情况下才有效。这允许设备和/或用户具有单一公共身份，而被给予对非敏感元数据有访问权的验证者能够验证设备和/或用户的真实性以及元数据的源头。

[0066] 图1示出了元数据绑定的过程流。首先，注册参数1被取回且可以通过密码学散列函数3与敏感元数据2组合。密码学散列函数的输出被用作物理不可克隆函数4的输入，物理不可克隆函数4将注册参数和任选的元数据与硬件身份链接。最后，注册标记5被返回作为PUF输出的函数。

[0067] 一般地，散列函数被定义为$H(\bullet):\{0,1\}^* \rightarrow \{0,1\}^{\lambda}$，其中λ是固定常数。也即，散列函数H(•)(或者显式地写为Hash(•)取任意大小的输入，并且映射到有限输出域。对于密码学设定，散列函数必须满足附加的性质。在认证协议中绑定元数据的上下文中，散列函数应当优选地是单向的、耐冲突的，并且满足雪崩条件。单向意味着当给定输出H(x)时确定输

入x其在计算上是不可行的，确保输出H(x)不显露有关输入x的任何信息。耐冲突意味着提供不同的元数据集合y而使得H(x)＝H(y)是计算上不可行的，其中x是对于给定实体的正确元数据。雪崩条件意味着，$H(\bar{x})$的每个位以概率1/2与H(x)互补，其中x是任何散列输入，$\bar{x}$是单个位被互补的x，确保输出H(x)响应于输入x的微小变化而实质上变化，这允许对待检测元数据进行任何改变且强制失败的认证。

[0068] 一种绑定元数据$\mathcal{M}_i$的方式是将PUF输入x简单地重新定义为$H(c_i,\mathcal{M}_i,G,p,q)$，而不是H($c_i$,G,p,q)。因此，修正的注册算法6变成：

[0069]

算法 6 元数据注册

for Device d **do**
Select finite field $\mathbb{F}_p$ of order p
Select E, an elliptic curve over $\mathbb{F}_p$
Find $G \in E/\mathbb{F}_p$, a base point of order q
Retrieve Metadata $\mathcal{M}_i$
$c_i \in \mathbb{F}_p$, a group element
$x = H(c_i, \mathcal{M}_i, G, p, q)$
$O = PUF(x)$
$\text{helper}_i = O \oplus \text{ECC}(p_i^{\text{priv}} \bmod q)$
$p_i^{\text{pub}} = A_i = p_i^{\text{priv}} \cdot G \bmod p$
Store $\{p_i^{\text{pub}}, c_i, \text{helper}_i\}$
end for

[0070] 然而，其它各种值的置换(与所使用的数学架构有关)可以散列以产生并入了元数据的PUF输入，而且一个或多个值可以迭代地散列和/或散列值被嵌套(例如，$H(H(c_i\|\mathcal{M}_i),G,p,q)$，等等)。进一步，可以采用其它用于链接和/或组合参数的方法(例如，全部或无变换(all-or-nothing transformation))。

[0071] 由于散列函数的雪崩性质，元数据必须确切地相同从而使认证成功。然而，生物计量认证的示范性的实施方案经常导致噪声，其中尽管观察到相同的特性(例如，指纹、虹膜等)，但是扫描略微不同。因此，诸如模糊提取器的工具可以优选地被采用来确保生物计量可靠地返回恒定值。例如，对于元数据的恒定值$\mathcal{M}_i$可以被选定且链接到相关联的公共助手数据值$h_i^{\mathcal{M}}$。有噪声的生物计量扫描$\mathcal{S}$则能够用来计算$h_i^{\mathcal{M}} \leftarrow ECC(\mathcal{M}_i) \oplus \mathcal{S}$，其中ECC是纠错码，并且被给予对新的生物计量扫描$\bar{\mathcal{S}}$的访问权，也即，t接近$\mathcal{S}$，通过计算$\mathcal{M}_i \leftarrow D(h_i^{\mathcal{M}} \oplus \bar{\mathcal{S}})$来恢复恒定值$\mathcal{M}_i$，其中D是对应的错误译码算法。

[0072] 图2示出了构造零知识证据的过程流，演示在使用生物计量认证传感器(例如，指纹扫描仪)的实施方案中传感器完整性、用户认证和传感器输出验证。硬件可以是模块(例如，U.are.U 4500)的部分，或者仅限于传感器(例如，TCS4K Swipe Sensor)。优选地，PUF电路(例如，环形振荡器、SRAM、仲裁器等)将直接与传感器硬件集成，使得对传感器硬件的修改改动PUF映射。该设备的一个实施方案可以包括Xilinx Artix 7现场可编程门阵列(FPGA)平台，配备有例如215,000个逻辑单元，13兆字节的块随机存取存储器，以及700个数字信号处理(DSP)片。在采用例如椭圆曲线密码学的实施方案中，硬件数学引擎可以例示于板上DSP片中，PUF构造位于逻辑单元内，逻辑处理核包括到PUF的输入和输出且构造成控制

那些以及设备的外部输入和输出且执行算法(发送椭圆曲线和其它数学运算到数学引擎)，诸如上文所述的那些。

[0073] 首先，取回注册参数6，注册参数可以通过密码学散列函数8与敏感元数据7组合。密码学散列函数的输出被用作物理不可克隆函数9的输入，物理不可克隆函数9将注册参数和敏感元数据与硬件身份链接。接着，证据参数10和传感器输出11通过密码学散列函数12链接。物理不可克隆函数9的输出和密码学散列函数12的输出被合成以生成零知识证据13，其输出将说服验证者传感器的完整性、认证用户且验证传感器输出的证据标记14。

[0074] 算法7提供了指纹扫描如何与认证协议绑定而使得设备和指纹必须匹配那些原始注册的信息的实施例。非敏感元数据(例如，对于温度、压力等的传感器输出) $\mathcal{M}_i^{\text{pub}}$ 可以通过被并入特定场合N的构造且将 $\mathcal{M}_i^{\text{pub}}$ 提供给验证者而并入非交互式认证算法中。因此，如果 $\mathcal{M}_i^{\text{pub}}$ 匹配传感器的输出，则验证者仅能够构造特定场合N(并且因此，变量c')。

算法 7 设备&指纹认证

for User **do**
Scan Fingerprint
$\bar{FP} \leftarrow$ Scan
Read Fingerprint Helper Data $h_i^{\mathcal{M}}$
$\mathcal{M}_i^{FP} \leftarrow ECC(h_i^{\mathcal{M}} \oplus \bar{FP})$
end for
for PUF Device d **do**
$p_i^{\text{priv}} \leftarrow \text{PUF-Retrieve}(c_i, \mathcal{M}_i^{FP}, \text{helper}_i)$
$A_i = p_i^{\text{priv}} \cdot G \mod p$
$r \leftarrow \text{random} \in \mathbb{F}_n$, a random group element
$B \leftarrow r \cdot G \mod p$
[0075] $N \leftarrow \text{Hash}(A \| \mathcal{M}_i^{\text{pub}} \| \tau)$ where τ is the current timestamp
$c \leftarrow \text{Hash}(G, B, A, N)$
$m \leftarrow r + c \cdot p_i^{\text{priv}} \mod q$
Server $s \leftarrow \{B, m, \mathcal{M}_i^{\text{pub}}, \tau\}$
end for
for Server s **do**
$A_i = p_i^{\text{priv}} \cdot G \mod p$ (public key stored from device enrollment)
$N \leftarrow \text{Hash}(A \| \mathcal{M}_i^{\text{pub}} \| \tau)$
$c' \leftarrow \text{Hash}(G, B, A, N)$
$P \leftarrow m \cdot G - c' \cdot A \mod p$
$\text{Device } d \leftarrow \begin{cases} \text{accept} & : P = B \\ \text{deny} & : P \neq B \end{cases}$
end for

[0076] 首先，用户的指纹扫描 $\bar{FP}$ 与原始指纹扫描FP的助手数据 $h_i^{\mathcal{M}}$ 相结合使用来恢复元数据值 $\mathcal{M}_i$。接着，元数据值 $\mathcal{M}_i$ 被用作PUF的输入，使得PUF输出取决于元数据。为了绑定非敏感元数据 $\mathcal{M}_i^{\text{pub}}$ 到证据，其用来构造特定场合N，特定场合N取决于公共身份A以及当前时间戳τ(其防止重放攻击)。非敏感元数据 $\mathcal{M}_i^{\text{pub}}$ 随后提供给验证者，因为其现在有必要验

证证据。(如果非敏感元数据仅应当显露给验证者,则可以加密发送)。最后,设备构造非交互式零知识证据,这使得服务器能够验证设备和(敏感和非敏感)元数是否正确。

[0077] 还可以通过要求服务器将特定场合N发布给设备来构造交互式零知识证据。该示例性的构造图示在算法8中。

[0078]

算法 8 交互式设备&指纹认证

for Server s **do**
Send nonce $N \in \{0,1\}^{\lambda}$ to Device, where λ is the number of bits in the modulus p
end for
for User **do**
Scan Fingerprint
$\bar{FP} \leftarrow$ Scan
Read Fingerprint Helper Data $h_i^{\mathcal{M}}$
$\mathcal{M}_i^{FP} \leftarrow ECC(h_i^{\mathcal{M}} \oplus \bar{FP})$
end for
for PUF Device d **do**
$p_i^{\text{priv}} \leftarrow \text{PUF-Retrieve}(c_i, \mathcal{M}_i^{FP}, \text{helper}_i)$
$A_i = p_i^{\text{priv}} \cdot G \mod p$
$r \leftarrow \text{random} \in \mathbb{F}_p$, a random group element
$B \leftarrow r \cdot G \mod p$
$c \leftarrow \text{Hash}(G, B, A_i, N)$
$m \leftarrow r + c \cdot p_i^{\text{priv}} \mod q$
Server $s \leftarrow \{B, m, \mathcal{M}_i^{\text{pub}}\}$
end for
for Server s **do**
$A_i = p_i^{\text{priv}} \cdot G \mod p$ (public key stored from device enrollment)
$c' \leftarrow \text{Hash}(G, B, A_i, N)$
$P \leftarrow m \cdot G - c' \cdot A_i \mod p$
$\text{Device } d \leftarrow \begin{cases} \text{accept} & : P = B \\ \text{deny} & : P \neq B \end{cases}$
end for

[0079] 在本发明的实施方案中,(敏感和/或非敏感)元数据的添加是任选的。也即,可以包含非敏感元数据,同时排除敏感元数据。这仅要求公共身份标记不包含敏感元数据。类似地,可以包含敏感元数据,而排除非敏感元数据。这仅要求不利用非敏感元数据构造特定场合。

[0080] 由于本发明的一个实施方案依赖于椭圆曲线数学架构,所以本领域技术人员将认识到其可以扩展以支持基于密码学强制角色的访问控制(RBAC)。也即,可以数学方式来规定数据访问策略和设备证书,并且RBAC算法计算将策略$\mathscr{P}$和证书$\mathscr{C}$映射到{0,1}中的访问决策的函数$f(\mathscr{P},\mathscr{C}) \mapsto \{0,1\}$。这通常是通过构造双线性配对(例如,Weil或Tate配对)来实现,并且是本发明的自然扩展。

[0081] 虽然已经利用各种特征描述了前面的实施方案,本领域普通技术人员将认识到,认证协议无需限于零知识,可以基于其它用来确立身份的密码学构造。例如,设备可以使用其硬件身份来数字地签署分组的内容,并且在分组报头(例如,TCP选项报头,其中实施例报头包括{B=r·G mod p,m=r+Hash(G,B,A,N)·rand mod q,τ})内包含该签名,并且硬件身份可以应用于其它各种密码学认证技术,并且无需受所提供的实施例的零知识方案限制。

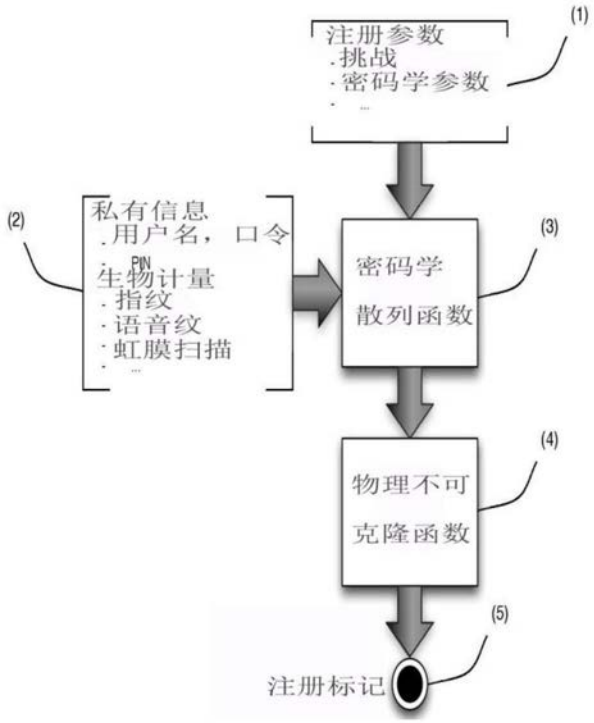

(1)
注册参数
·挑战
·密码学参数
·...
(2)
私有信息
·用户名，口令
·PIN
生物计量
·指纹
·语音纹
·虹膜扫描
·...
(3)
密码学
散列函数
(4)
物理不可
克隆函数
(5)
注册标记

图1

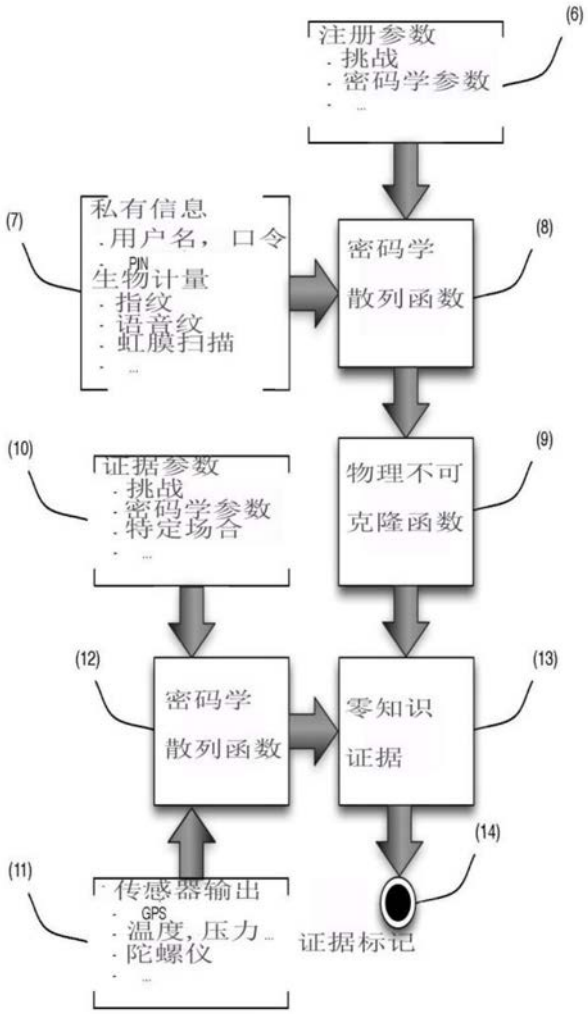

(6)
注册参数
· 挑战
· 密码学参数
· ...
(7)
私有信息
· 用户名，口令
PIN
生物计量
· 指纹
· 语音纹
· 虹膜扫描
· ...
密码学
散列函数
(8)
(9)
物理不可
克隆函数
(10)
证据参数
· 挑战
· 密码学参数
· 特定场合
· ...
(12)
密码学
散列函数
(13)
零知识
证据
(14)
证据标记
(11)
· 传感器输出
GPS
· 温度，压力 ...
· 陀螺仪
· ...

图2