聚力·引领
软件安全学术论坛

数据安全区块链与隐私保护

通付盾创始人 汪德嘉

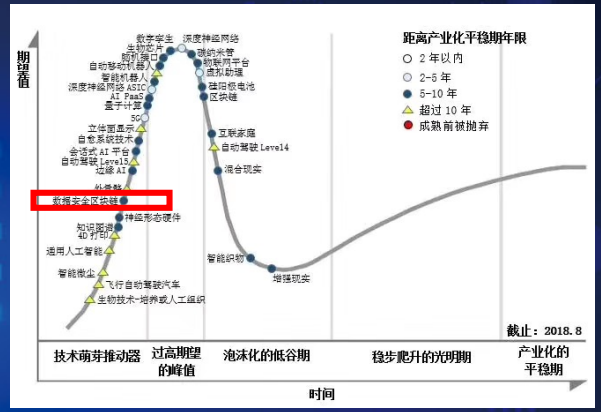
《网络安全法》

第十六条 国务院和省、自治区、直辖市人民政府应当统筹规划，加大投入，扶持重点网络安全技术产业和项目，支持网络安全技术的研究开发和应用，推广安全可信的网络产品和服务，保护网络技术知识产权，支持企业、研究机构和高等学校等参与国家网络安全技术创新项目。

《国家网络空间安全战略》提出的战略任务“夯实网络安全基础”，强调“尽快在核心技术上取得突破，加快安全可信的产品推广应用”。

网络安全等级保护制度2.0标准要求全面使用安全可信的产品和服务来保障关键基础设施安全。

数据安全保护与数字经济发展



大数据智能时代，数据已成为资产，开放带来便捷的同时，考验数据治理能力也带来新的数据安全隐患。打造数字经济，效率是基础、安全是底线，没有效率与安全就没有数字经济。数据的管理与使用需关注提高数据安全保护效率与能力。

第一个层面，利用区块链、MPC、零知识证明、同态加密、代理重签名等技术建立创新并高度安全可信的隐私保护机制，规避隐私泄漏问题。

第二个层面，建立可一个多层次的线上线下融合的多因子可信数字身份认证体系，并提供基于可信数字身份的系统登录、身份证明、数据资产确权等解决方案，给予用户和消费者最大的安全保障。

第三个层面，建设有效的数据治理机制和系统，提高数据融合、数据管理、数据应用能力。提高数据资产价值。

数据资产的特点

难转移

不同于数字资产，数字资产可以转账，但是数据资产不同，数据可以被复制，很难被转移，数字被复制后，同一份数据被双方占有。

持续产生

数据资产就像一座矿场，只要使用互联网，数据就会越来越丰富，数据价值就会越大。只要一个设备在工作，这台设备的数据资产就不断产生。

价值体现

不同于数字资产和实体资产，数据资产的价值在使用中才被体现，因为互联网的结构性问题，我们通常感知不到这部分资产。

用户为中心

随着数字经济的发展，现实世界被数字化，用户对隐私和数据的需求极大增加，用户需要更多的管理自己的数据，以用户为中心而不是服务为中心。

传统加密与安全计算

（1）数据安全保护的关键是确保隐私数据的“不可接触”和“不可解释”。

（2）数字身份认证在隐私保护方面针对“不可接触”问题，解决数据的使用授权，非通过身份认证的用户不能接触到数据。

（3）安全多方计算、零知识证明、代理重加密、同态加密针对数据“不可解释”问题，解决数据的加密保护。

（4）传统的加密方式（对称/非对称加密）把数据装进“黑盒子”，我们提到的加密方式是“白盒子”，解决数据使用权与所有权分离，在不泄露隐私的前提下使用数据。

（5）数据访问、数据使用、数据加密、数据交换等操作记录均通过数据安全区块链（KeyChain）存证，确保数据保护可信、可追溯。

零知识证明-理论

zkSNARKs，zero-knowledge Succint Non-interactive ARguments of Knowledge 的简称：

- Succinct：证明的数据量比较小
- Non-interactive：没有或者只有很少交互。
- ARguments：验证者只对计算能力有限的证明者有效。拥有足够计算能力的证明者可以伪造证明。这也叫 “计算可靠性”（相对的还有“完美可靠性”）。
- of Knowledge：对于证明者来说在不知道证据（Witness，比如一个哈希函数的输入或者一个确定Merkle-tree节点的路径）的情况下，构造出一组参数和证明是不可能的。

zkSNARKs大体由四部分组成：

(1) 多项式问题的转化

需要证明的问题转化为多项式问题 $t(x)h(x) = w(x)v(x)$，证明者提交证明让验证者确认多项式成立。

(2) 随机取样验证

验证者随机取样秘密的验证数值s，将多项式乘法和多项式函数相等的验证减小到数值上的简单的乘法和等值验证：$t(s)h(s) = w(s)v(s)$。相对于验证多项式相等$t(s)h(s) = w(s)v(s)$，随机取样验证，简单，验证数据少。随机挑选验证，安全性肯定不及多项式等式验证，但如果确实足够随机，安全性还是相当高的。

(3) 同态编码/加密

使用具有一些同态性质的编码/加密函数E，但不是全同态加密，这允许在不知道s的情况下计算$E(t(s)), E(t(s)), E(t(s)), E(t(s))$。“同态”是函数的特殊性质，包括加法同态性和乘法同态性。加法同态性指$E(x+y)$可以由$E(x)$和$E(y)$计算出来，乘法同态性指$E(x \cdot y)$可以由$E(x)$和$E(y)$计算出来。

(4) 零知识

证明者和验证者之间除了“问题证明与否”知识外，不知道其他任何知识（不知道随机挑选值，不知道挑选值的多项式计算结果等等）。

零知识证明- QSP问题的SNARK证明

需要进行零知识证明的问题，肯定是 NP 问题，如果是 P 问题，不存在问题解的“寻找”，也就不存在证明。简单的说，zkSNARK 问题处理的都是 NP 问题。既然 NP 问题相互可以归约，首先需要确定一个 NP 问题，其他 NP 问题都可以归约到这个 NP 问题，再进行证明。也就是，证明了一个 NP 问题，就可以证明所有 NP 问题。QSP 问题是个 NP 问题，也特别适合 zkSNARK

QSP问题：已知多项式$v_0,\cdots,v_m,w_0,\cdots,w_m$，目标多项式$t$（不超过$d$阶）以及输入比特串$u$，证明者找到$a_1,\cdots,a_m,b_1,\cdots,b_m$（一定程度上取决于$u$）以及多项式$h$满足

$$t\cdot h=(v_0+a_1v_1+\cdots+a_mv_m)(w_0+b_1w_1+\cdots+b_mw_m)$$

SNARK证明过程分为三部分：1）参数设置阶段，2）证明者提供证据阶段，3）验证者验证阶段。

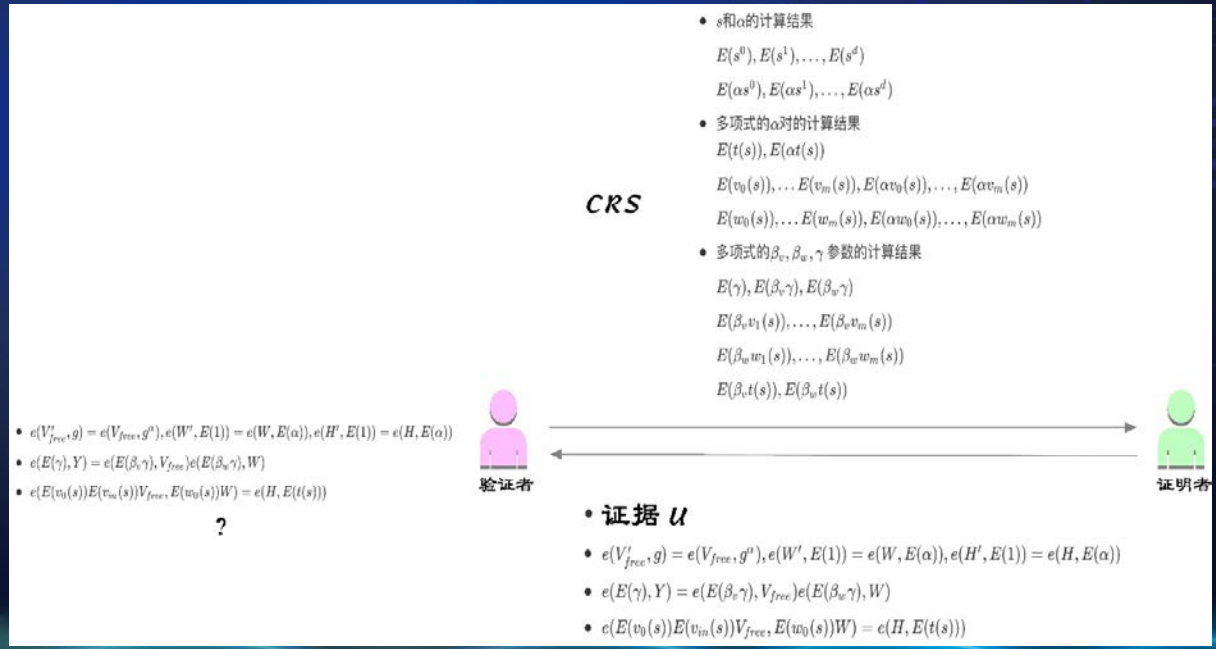
使用zkSNARK证明，由如下的几步组成：

1）问题转化：一个需要证明的NP问题转化为选定的NP问题（比如QSP问题）；

2）设置参数（setup）：设置参数的过程也是挑选随机数的过程，并提供CRS；

3）证明者获取证据u，通过CRS计算证据（proof）；

4）验证者验证证据以及响应的proof。



代理重加密-理论

代理重加密（Proxy Re-Encryption，PRE）：一个PRE方案可由五个算法组成：KeyGen，ReKeyGen，Encrypt，ReEncrypt，Decrypt.

(1) 密钥生成算法KeyGen(1^k) →(pk_i，sk_i)：输入安全参数1^k，$k \in \mathcal{K}$，KeyGen为用户i输出一对公私钥(pk_i，sk_i)

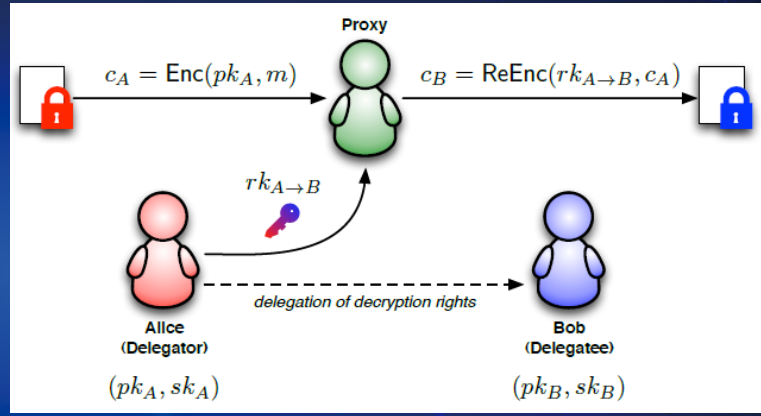
(2) 代理重加密密钥生成算法ReKeyGen(pk_A，sk_A，pk_B，sk_B) →$rk_{A\to B}$：输入Alice的公私钥对(pk_A，sk_A)和Bob的公私钥对(pk_B，sk_B)，ReKeyGen输出一个代理重加密密钥$rk_{A\to B}$。此时，Alice是委托者，Bob是被委托者。

(3) 加密算法Encrypt(pk_i，m) →c_i：输入用户i的公钥pk_i以及消息$m \in \mathcal{M}$，Encrypt算法输入消息m的密文$c_i \in \mathcal{C}_1$。

(4) 代理重加密算法ReEncrypt($rk_{A\to B}$，c_A)→c_B：输入一个代理重加密密钥$rk_{A\to B}$和Alice的密文c_A，ReEncrypt输出针对Bob的重加密密文$c_B \in \mathcal{C}_2$。

(5) 解密算法Decrypt(sk_i，c_i) →m：输入用户i的私钥sk_i和密文c_i，Decrypt输出消息m或者错误符号⊥表明密文c_i不合法。

在上述代理重加密定义中，一个拥有代理重加密密钥rkA→B的半可信代理者能够将Alice公钥下的密文$c_A \in \mathcal{C}_1$重加密为Bob公钥下针对同一明文$m \in \mathcal{M}$的密文$c_B \in \mathcal{C}_2$。然后，Bob 能够解密并获得消息$m \in \mathcal{M}$，同时，该代理者无法获得任何信息（如skA，skB和m）。



在上述 PRE 定义的ReKeyGen算法中，被委托者Bob的私钥sk_B是可选的。一般地，当 Bob的私钥sk_B不参与代理重加密密钥生成时，该代理重加密方案具有单向性和非交互性；相反地，ReKeyGen算法输出一个双向代理重加密密钥，且该PRE方案具有交互性。此外，Encrypted算法和ReEncrypt算法输出的密文空间分别为$\mathcal{C}_1$和$\mathcal{C}_2$，当$\mathcal{C}_1 = \mathcal{C}_2$时，Encrypted算法和ReEncrypt算法的输出具有相同的密文空间，只需一个解密算法Decrypt就可以同时解密上述两种算法输出的密文；而当$\mathcal{C}_1 \neq \mathcal{C}_2$时，针对Encrypted算法和ReEncrypt算法，则需要两个不同的Decrypt算法。

代理重加密-特性及Umbral PRE系统

PRE系统具有九大特性，

(1) 单向性（unidirectional）：在一个单向PRE方案中，代理重加密密钥是单向的，即代理者可以利用一个单向代理重加密密钥$rk_{A\to B}$将Alice的密文转换为Bob的密文，而不能将Bob的密文转换为Alice的密文；相反地，双向（bidirectional）PRE不仅允许代理者将Alice的密文转换为Bob的密文，而且反之亦然。

(2) 复用性（multi- use）：在一个复用PRE方案中，Encryt算法和ReEncrypt算法的输出结果都可以再次作为ReEncrypt算法的输入；反之，单用（single-use）PRE只允许加密算法Encrypt的输出作为ReEncrypt算法的输入。

(3) 秘密代理（private-proxy）：在一个秘密代理重加密中，代理者是诚实的且能够确保代理重加密密钥的隐私性，即攻击者无法从密文转换过程中获取代理重加密密钥；反之，在公开代理（public-proxy）PRE中，攻击者可以通过观察代理者的输入与输出计算出代理重加密密钥。

(4) 透明性（transparent）：在一个具有透明性的PRE方案中，代理者是透明的，即由Encrypt算法输出的密文与由ReEncrypt算法输出的密文在计算上是不可区分的。

(5) 密钥优化（key-optimal）：在密钥优化的PRE方案中，不论存在多少委托者或被委托者，用户只需保护和存储的少量的秘密数据；

(6) 非交互性（non- interactive）：在非交互性的PRE方案中，代理重加密密钥由委托者的公私钥对和被委托者的公钥产生，即被委托者不参与代理重加密密钥的授权过程。

(7) 非传递性（non-transitive）：在非传递性PRE方案中，代理重加密密钥具有非传递性，即给定A→B的代理重加密密钥$rk_{A\to B}$和B→C的代理重加密密钥$rk_{B\to C}$，代理者无法通过计算得到A→C的代理重加密密钥$rk_{A\to C}$。

(8) 暂时性（temporary）：在暂时性的PRE方案中，代理重加密密钥是可撤销的，即委托者有权收回委托出去的解密授权；

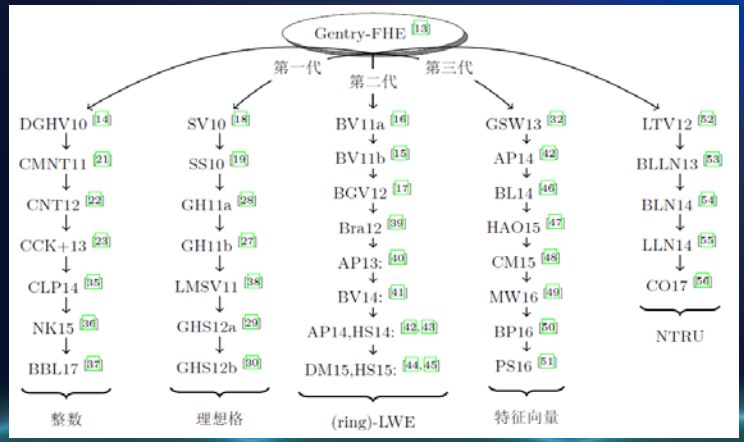
(9) 抗合谋攻击（collusion-resistant）：在抗合谋攻击的 PRE 方案中，代理重加密密钥能够抵抗合谋攻击，即当被委托者与代理者合谋时，二者无法揭露委托者的私钥和明文信息。

(10) 符合以上PRE特点的Umbral PRE架构已经进入商用阶段，通付盾隐私数据加密云托管就是基于Umbral实现的。

同态加密-理论

同态加密（Homomorphic Encryption）是很久以前密码学界就提出来的一个Open Problem。早在1978年，Ron Rivest, Leonard Adleman, 以及ichael L. Dertouzos就以银行为应用背景提出了这个概念。提出第一个构造出全同态加密（Fully Homomorphic Encryption）[Gen09]的Craig Gentry给出的直观定义最好：**A way to delegate processing of your data, without giving away access to it.**

一般的加密方案关注的都是**数据存储安全**。即，我要给其他人发个加密的东西，或者要在计算机或者其他服务器上存一个东西，我要对数据进行加密后在发送或者存储。没有密钥的用户，不可能从加密结果中得到有关原始数据的任何信息。只有拥有密钥的用户才能够正确解密，得到原始的内容。我们注意到，这个过程中**用户是不能对加密结果做任何操作的**，只能进行存储、传输。对加密结果做任何操作，都将会导致错误的解密，甚至解密失败。同态加密方案最有趣的地方在于，其关注的是**数据处理安全**。同态加密提供了一种对加密数据进行处理的功能。也就是说，其他人可以对加密数据进行处理，但是处理过程不会泄露任何原始内容。同时，拥有密钥的用户对处理过的数据进行解密后，得到的正好是处理后的结果。



全同态加密发展概况：构造全同态加密方案的困难问题主要分为两大类：

(1) 基于整数上的近似最大公约数(approximate greatest common divisor, AGCD)问题

(2) 基于格上的问题：

1) 基于理想格以Gentry方案为代表的第一代FHE方案
2) 基于LWE假设利用模数转换、密钥交换等技术完成的第二代FHE方案
3) 基于LWE假设利用近似特征向量实现的第三代FHE方案
4) 基于NTRU假设利用密钥交换技术实现的FHE方案

同态加密-RSA

（1）密钥生成算法

随意选择两个大的质数p和q，p不等于q，计算$N = pq$。根据欧拉函数，求得$r = (p-1)(q-1)$。选择一个小于r的整数 e，求得e关于模r的模反元素，命名为d。（模反元素存在，当且仅当e与r互质）将 p 和 q 的记录销毁。(N,e)是公钥，(N,d)是私钥。

(2) 加密算法

假设Bob想给Alice送一个消息m，他知道Alice产生的N和e。他使用起先与Alice约好的格式将m转换为一个小于N的整数n，比如他可以将每一个字转换为这个字的Unicode码，然后将这些数字连在一起组成一个数字。假如他的信息非常长的话，他可以将这个信息分为几段，然后将每一段转换为n。用下面这个公式他可以将n加密为c：$n^e \equiv c \pmod N$，Bob算出c后就可以将它传递给Alice。

(3) 解密算法

Alice得到Bob的消息c后就可以利用她的密钥d来解码。她可以用以下这个公式来将c转换为n：$c^d \equiv n \pmod N$，得到n后，她可以将原来的信息m重新复原。

(4) 乘法同态性

对于明文m_1和m_2，加密后密文为$c_1 = {m_1}^e \pmod N$和$c_2 = {m_2}^e \pmod N$。则$c_1 \cdot c_2 \pmod N = {m_1}^e \cdot {m_2}^e \pmod N = (m_1 \cdot m_2)^e \pmod N$解密后即为$m_1 \cdot m_2$。

安全多方计算-理论

当一个 MPC 计算任务发起时，枢纽节点传输网络及信令控制。每个数据持有方可发起协同计算任务。通过枢纽节点进行路由寻址，选择相似数据类型的其余数据持有方进行安全的协同计算。参与协同计算的多个数据持有方的 MPC 节点根据计算逻辑，从本地数据库中查询所需数据，共同就 MPC 计算任务在数据流间进行协同计算。在保证输入隐私性的前提下，各方得到正确的数据反馈，整个过程中本地数据没有泄露给其它任何参与方。

安全多方计算理论主要研究参与者间协同计算及隐私信息保护问题，其特点包括输入隐私性、计算正确性及去中心化特性。

（1）输入隐私

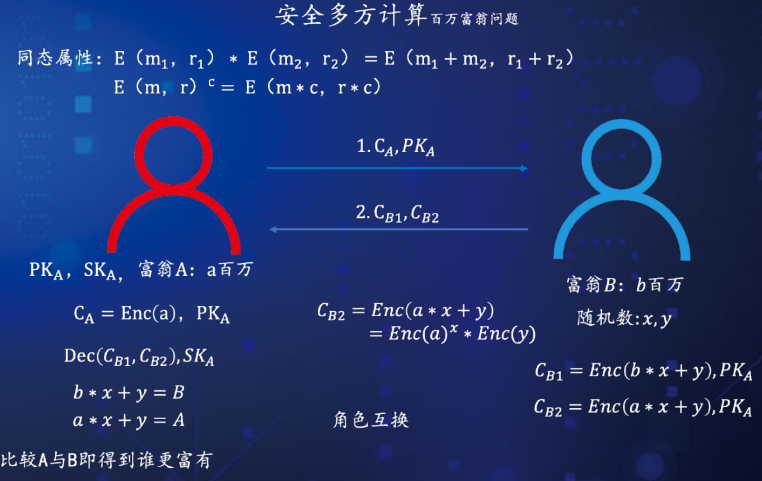
安全多方计算研究的是各参与方在协作计算时如何对各方隐私数据进行保护，重点关注各参与方之间的隐私安全性问题，即在安全多方计算过程中必须保证各方私密输入独立，计算时不泄露本地任何数据。

（2）计算正确

多方计算参与各方就某一约定计算任务，通过约定 MPC 协议进行协同计算。计算结束后，各方得到正确的数据反馈。

（3）去中心化

传统的分布式计算由中心节点协调各用户的计算进程，收集各用户的输入信息。而在安全多方计算中，各参与方地位平等，不存在任何有特权的参与方或第三方，提供一种去中心化的计算模式。



安全多方计算-应用

1. 多头借贷问题

两个网贷公司A和B，想知道借款人是否在对方公司借款了，但他们又不想让对方知道借款人的信息。

2. 安全电子选举问题

电子选举方案需要满足，选票保密性、无收据性、健壮性、公平性和普遍验证性等性质。整个选举方案没有可信第三方，任何投票人都可以计票，比一般的方案具有更强的安全性。有具体解决方案。

3. 遗传病诊断

Alice认为她得了某种遗传病，想验证自己的想法。正好她知道Bob有一个关于疾病的DNA模型的数据库。如果她把自己的DNA样品寄给Bob，那么Bob可以给出她的DNA的诊断结果。但是Alice又不想别人知道，这是她的隐私。所以，她请求Bob帮忙诊断自己DNA的方式是不可行的。因为这样Bob就知道了她的DNA及相关私人信息。

4. 商业竞争与业务合作

经过一次花费查后，A公司决定扩展在某些地区的市场份额来获取丰厚的回报。同时，A公司也注意到B公司也在扩展一些地区的市场份额。在策略上，两个公司都不想在相同地区互相竞争，所以他们都想在不泄露市场地区位置信息的情况下知道他们的市场地区是否有重叠。(信息的泄露可能会导致公司很大的损失。比如另一家对手公司知道A和B公司的扩展地区，提前行动占领市场；又比如房地产公司知道A和B公司的扩展计划，提前提高当地的房租等等)所以他们需要一种方法在保证私密的前提下解决这个问题。

5. 电子拍卖

在进行电子拍卖时，各方依赖于拍卖服务器的安全性来进行拍卖活动，但是在实际中，会出现投标者不愿意泄露标的投标价、各方对于拍卖行不信任的情况。而安全多方安全计算技术的加入则使得电子拍卖成为现实，大部分方案都采取了可验证秘密共享协议或者使用其思想，具备灵活性、保密性、鲁棒性和可验证性。

6. 与区块链的结合

区块链与安全多方计算有一个共同点，都是多个节点协同完成一项任务，不同之处在于区块链所搭建的是一套可信的去信任网络，即保证不会有篡改计算结果的问题发生，但不能做到保护原始数据，而安全多方计算可以在多个节点之间进行协同计算任务时，让数据还保留在原节点上，不会让隐私数据泄露到外界。如果将两者结合在一起就可以做到既限制人为作恶篡改计算结果，又能够保护数据隐私，形成一套非常完整的解决方案，全面提高区块链系统隐私保护方面的能力。

实践：通付盾BA-PP子系统

为减小传统数字货币中因密钥丢失带来的危害，通付盾在基于区块链的身份认证系统（BA系统）中设计并实现了BA-PP隐私保护子系统。在该系统中，用户在丢失密钥时可通过向存管机构等认证节点提供有效身份信息（如安全问题答案等）更换密钥。 BA-PP子系统使用了零知识证明技术，使得能够在对用户隐私身份信息做认证的同时保障用户隐私身份信息不暴露在网络中，实现了用户信息的隐匿性。

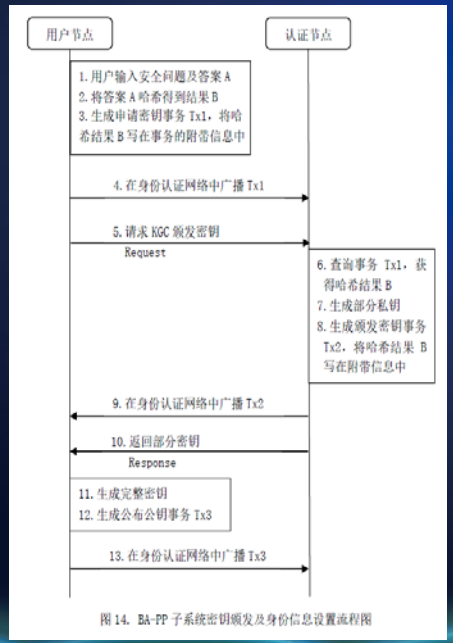


图 14. BA-PP 子系统密钥颁发及身份信息设置流程图

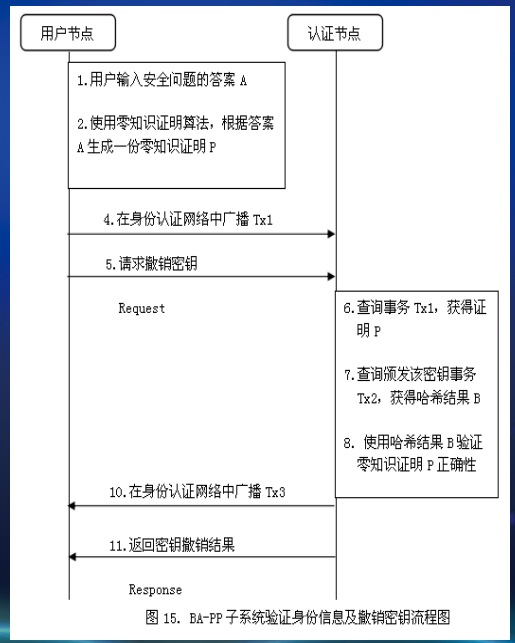


图 15. BA-PP 子系统验证身份信息及撤销密钥流程图

感谢您的聆听

THANK YOU FOR LISTENING

聚力·引领

软件安全学术论坛