计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(3):38−47 [doi: 10.15888/j.cnki.csa.008343]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于区块链的生物特征和口令双因子跨域认证与密钥协商方案①

李　广, 范冰冰

(华南师范大学 计算机学院, 广州 510631)
通信作者: 范冰冰, E-mail: 2019022659@m.scnu.edu.cn

摘　要: 针对用户跨域访问数据资源的数据共享场景, 为了保证用户的身份合法性以及安全通信, 提出了一种基于区块链的生物特征和口令双因子跨域认证与密钥协商方案. 用户生物特征通过模糊提取技术转换为生物密钥和生物公开信息参与认证, 避免生物特征被泄露. 包含生物密钥和生物公开信息的用户身份信息存储在区块链账本中, 保证身份信息一致以及不被篡改. 认证域的认证服务器在跨域认证时无须与用户注册域的认证服务器交互, 而是直接通过查询区块链账本获取用户身份信息, 完成对跨域访问用户的认证. 安全性和性能分析表明, 方案所提协议能够以更少的计算开销, 提供更强的安全性.

关键词: 跨域认证; 生物认证; 区块链; 模糊提取; 密钥协商

引用格式: 李广,范冰冰.基于区块链的生物特征和口令双因子跨域认证与密钥协商方案.计算机系统应用,2022,31(3):38–47. http://www.c-s-a.org.cn/1003-3254/8343.html

Biometric and Password Two-factor Cross-domain Authentication and Key Agreement Scheme Based on Blockchain

LI Guang, FAN Bing-Bing

(School of Computer Science, South China Normal University, Guangzhou 510631, China)

Abstract: In data sharing scenarios where users access data resources across domains, their identity legitimacy and secure communication need to be ensured. To this end, this study proposes a two-factor, i.e., biometrics and passwords, cross-domain authentication and key agreement scheme based on blockchain. Fuzzy extraction technology is used to extract the key and public information of users' biometrics for authentication participation, avoiding biometric information leakage. The blockchain ledger is used to store users' identity information including biometric keys and biometric public information, ensuring the consistency of users' identity information without any tampering. In cross-domain authentication, the authentication server in the authentication domain does not need to communicate with the authentication server in the user registration domain. Instead, it is completed by directly querying the blockchain ledger to obtain users' identity information. Security and performance analysis show that the proposed scheme can provide stronger security with less computational overhead.

Key words: cross-domain authentication; biometric authentication; Blockchain; fuzzy extraction; key agreement

大数据时代, 为了提升数据的使用价值, 越来越多机构、公司及组织进行数据共享, 例如电子病历[1]、政务数据[2]的共享. 由于数据源位于不同的机构、公司及组织的内部[3], 基于保护和管理自身数据源, 各个机

① 基金项目: 广东省重大科技专项 (2016B030305003)
收稿时间: 2021-05-05; 修改时间: 2021-05-28; 采用时间: 2021-06-08; csa 在线出版时间: 2022-01-24

构、公司以及组织都建立起各自的数据平台, 形成独立的信任域, 对域内用户进行身份认证. 然而数据共享离不开数据源中数据资源的跨域流通, 因此为了数据资源的安全共享, 需要在分布式、多域数据平台之间对访问数据资源的其他域用户进行跨域认证.

生物特征和用户有天然的不可分离性, 无须特别记忆和特殊保存[4], 同时生物特征具有唯一性和一定时期内无显著变化的稳定性, 并且其难以伪造. 因此生物特征可保证用户数字身份与物理身份的统一[5], 为用户提供更加安全便捷的身份认证. Dodis 等人提出的模糊提取技术[6] 实现了将一定误差范围内的两个生物特征模板通过模糊提取产生同一个字符串作为用户密钥参与认证, 在间接实现生物特征比对的同时保护生物特征不被泄露; 口令便于记忆、管理, 是一种简单且方便的辅助认证手段. 近些年利用模糊提取技术的生物特征和口令双因子认证受到研究者们的关注. 文献 [7] 将口令放大技术与模糊提取技术结合, 使低熵口令转换为高熵新口令, 克服了传统口令认证中低熵口令的低安全性缺点, 提高了认证安全性. 文献 [8] 实现了用户与服务器的生物特征和口令双因子认证与密钥协商协议, 该协议利用模糊提取技术使服务器不再保存用户生物特征, 避免了服务器被攻陷用户敏感信息丢失的风险, 同时使用服务器的公钥保护用户认证信息, 避免了认证信息中的口令遭受离线字典攻击. 文献 [9] 提出了一种基于区块链技术的无线传感网络用户认证和密钥协商方案, 该方案采用 ECC 与生物特征和口令相结合的方式兼顾协议的安全性和效率.

区块链通过深度整合密码学技术、P2P 网络、共识机制以及智能合约等技术, 实现了去中心化、分布式以及信息不可篡改的信任建立机制[3]. 基于此, 将区块链应用于跨域认证受到研究者们的重视. 目前将区块链与传统 PKI 认证相结合的跨域认证方案已有较多研究成果[10–12]. 而利用区块链与生物特征认证相结合的跨域认证方案目前研究较少. 文献 [13] 提出了一种基于区块链的生物特征结合动态口令的跨域认证方案, 该方案基于离散对数问题把用户的静态口令转化为动态口令, 基于区块链的分布式存储功能实现了用户在本地和异地环境下的双因子跨域认证, 但该方案未实现用户与服务器的密钥协商. 文献 [14] 利用模糊提取技术和区块链技术, 使用智能合约来实现认证逻辑, 也提出了一种用于本地和异地环境的生物特征和口令双因子跨域认证方案, 但该方案采用多次非对称加解密操作, 认证效率较低. 同时文献 [13] 和文献 [14] 的协议在跨域认证时都需要用户注册域的认证服务器参与认证, 而注册域的认证服务器可能发生繁忙、宕机等状况导致认证失败, 因此认证可靠性较弱.

针对用户跨域访问数据资源的数据共享场景, 为了保证用户的身份合法性以及安全通信, 本文提出了一种基于区块链的生物特征和口令双因子跨域认证与密钥协商方案. 该方案利用模糊提取技术提取用户生物特征的密钥和公开信息参与认证, 避免生物特征泄露; 利用不易篡改的区块链实现用户身份信息的分布式存储, 保证用户身份信息一致, 同时提高认证的可靠性; 基于椭圆曲线离散对数问题的困难性, 实现用户匿名性以及用户与服务器之间的密钥协商.

1 椭圆曲线离散对数问题

在椭圆曲线E上, 选取一个基点P, 其阶为大素数n. 对于任一数$k \in Z_n^*$, 有$Q = kP$, 此时称k为Q的椭圆曲线离散对数. 给定E和n, 根据P和Q求解k为椭圆曲线离散对数问题 (ECDLP). 其困难性主要体现在: 已知k和P计算出Q比较容易, 但是已知P和Q计算出k则很困难.

与有限域离散对数问题和大整数因子分解问题相比, 椭圆曲线离散对数问题的求解难度更大, 其在多项式时间内无法被所有的已知算法求解[15]. 这说明在密码学中利用椭圆曲线离散对数问题的困难性可获得更高的安全性; 在相同安全强度下, 椭圆曲线密钥长度更小, 对带宽和存储要求更低, 更加节省计算资源.

2 系统架构与区块链设计

2.1 系统架构

本文方案基于区块链的认证系统架构如图 1 所示, 由多个信任域和区块链网络共同组成. 每个信任域主要包含 6 种实体: 用户、生物特征采集器、客户机、认证服务器、资源服务器以及共识服务器. 其中生物特征采集器用于获取用户的生物特征模板; 客户机连接生物特征采集器, 用于处理用户的输入; 认证服务器用于处理用户的认证请求, 帮助用户和资源服务器建立信任; 资源服务器用于为用户提供数据资源, 一个信任域的数据平台可能整合多个资源服务器的数据资源; 认证服务器和共识服务器共同用于区块链网络的组建.

当用户想访问数据平台的数据资源时, 首先要通过认证服务器的认证登录该数据平台, 然后通过资源服务器的认证, 同时为了保证后续与资源服务器的安全通信, 用户需与资源服务器协商一个会话密钥.

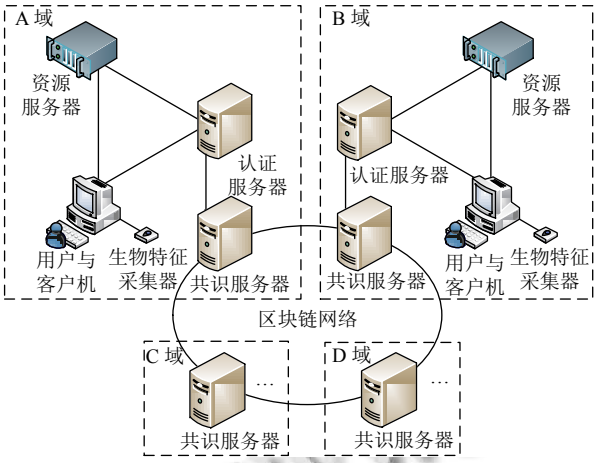

图 1 基于区块链的认证系统架构

由于联盟链具有多中心、集体维护的特点, 符合跨域认证中多个信任域的分布式网络环境. 因此本文是由在各个信任域的服务器上部署的网络节点组成的联盟链网络. 其中共识服务器上部署共识节点, 认证服务器上部署服务节点, 资源服务器上不部署节点. 在实际部署中, 服务节点也可不部署在认证服务器上; 将不同类型节点部署在不同服务器上, 可提高区块链整体运行效率. 共识节点的功能是参与共识机制将交易打包成区块追加到区块链中, 更新区块链账本; 服务节点通过与共识节点同步, 存储最新区块链账本, 其功能是接收并处理区块链交易请求, 对于获取账本数据的“读操作”交易请求, 服务节点直接通过查询本地存储的区块链账本返回相应结果, 不需要经过共识流程, 而对于更新账本数据的“写操作”交易请求, 服务节点将交易广播给共识节点进行处理.

2.2 区块链交易

本文使用区块链账本存储各个信任域中的用户身份信息, 其数据结构如图 2 所示, 其中, *UID*表示用户唯一身份标识; *P*表示用户生物特征经模糊提取技术产生的生物公开信息; $h(UID||PW||R)$表示用户登录凭证, 其中*PW*表示用户静态口令, *R*表示用户生物特征经模糊提取技术产生的生物密钥; *Status*表示用户状态, 分为注册 (register) 和注销 (revoke). 因此本文区块链交易主要是对用户身份信息的操作.

认证系统中发送用户身份信息相关交易请求有认证服务器和客户机两个实体, 共涉及 4 种交易: 注册交易、注销交易、更新交易以及查询交易. 注册和更新交易的数据结构如图 3 所示, *Signature*为认证服务器的签名; 注销交易的数据结构如图 4 所示, *Signature*为认证服务器的签名; 查询交易的数据结构如图 5 所示, *Signature*为认证服务器或客户机的签名.

UID	*P*	$h(UID\|\|PW\|\|R)$	*Status*

图 2 用户身份信息的数据结构

UID	*P*	$h(UID\|\|PW\|\|R)$	*Signature*

图 3 注册和更新交易的数据结构

UID	$h(UID\|\|PW\|\|R)$	*Signature*

图 4 注销交易的数据结构

UID	*Signature*

图 5 查询交易的数据结构

注册、更新和注销交易都属于“写操作”交易, 当认证服务器向区块链服务节点发送注册、注销和更新交易请求, 服务节点将交易广播给共识节点, 共识节点参与共识机制将交易打包成区块追加到区块链中, 完成认证服务器所在信任域的用户身份信息在区块链账本中的“增删改”. 查询交易属于“读操作”交易, 当认证服务器向区块链服务节点发送查询交易请求, 服务节点直接查询本地存储的区块链账本, 返回账本中不同信任域的用户身份信息给认证服务器; 当客户机向区块链服务节点发送查询交易请求, 服务节点直接查询本地存储的区块链账本, 返回账本中不同信任域的用户生物公开信息*P*给客户机.

2.3 区块链共识机制

共识机制用于实现区块链全网数据的一致性, 是设计区块链应用方案的关键. 为了高效地达到全网节点共识, 同时防止可能存在的节点恶意操作行为, 本文选择适用于联盟链的实用拜占庭容错共识机制——PBFT (practical Byzantine fault tolerance). 根据 PBFT 共识机制, 将本文的共识节点分为以下两种角色:

(1) 主节点: 根据选举策略从共识节点中选举一名

担任, 选举策略为公式$p = v \bmod |R|$, 其中p为当前主节点的编号, v为当前视图的编号 (视图编号从 0 开始连续递增), $|R|$为共识节点总个数 ($|R| = 3f + 1$, f为可容忍的无效或恶意共识节点个数, 共识节点用$\{0, 1, \cdots, |R| - 1\}$编号).

(2) 备份节点: 未选举上主节点的其他共识节点.

在本文区块链网络中, 各类节点职责的具体描述如下: 作为主节点的共识节点负责将交易打包成新区块并广播给备份节点, 参与新区块共识更新自身的区块链账本; 作为备份节点的共识节点仅负责参与新区块共识更新自身的区块链账本; 而作为非共识节点的服务节点负责从主节点或备份节点获取达成共识的新区块, 同步更新自身的区块链账本.

3 双因子认证与密钥协商

本节基于认证系统架构, 对方案的双因子认证与密钥协商协议进行设计. 由于区块链实现了用户身份信息的分布式存储, 客户机查询区块链账本即可得到用户的生物公开信息, 进而可通过模糊提取器技术恢复出用户的生物密钥; 认证服务器查询区块链账本即可得到用户的身份信息, 进而在跨域认证时无须与用户注册域的认证服务器交互, 即可对跨域访问的用户进行认证, 使得同域认证与跨域认证的步骤大致相同. 因此本协议分为用户身份注册、认证与密钥协商两个阶段. 协议中使用的符号说明如表 1 所示.

表 1 协议符号说明

符号	含义
U_{k-i}	k域的第i个用户
UID_{k-i}	用户U_{k-i}的身份标识
PW_{k-i}	用户U_{k-i}的静态口令
ω_{k-i}	用户U_{k-i}的生物特征
AS_k	k域认证服务器
(a_kP, a_k)	认证服务器AS_k的公私钥对
RS_{k-j}	k域的第j个资源服务器
RID_{k-j}	资源服务器RS_{k-j}的身份标识
X_{k-j}	AS_k与资源服务器RS_{k-j}的共享密钥
P	选取的椭圆曲线基点(系统公开信息)
$Gen(\cdot)/Rep(\cdot)$	模糊提取技术的生成/再生函数
$Enc_x(\cdot)/Dec_x(\cdot)$	对称加/解密函数(x为当前密钥)
T	时间戳
$h(\cdot)$	哈希函数
$\oplus$	异或操作
$\|\|$	级联操作

在协议设计前, 需进行以下初始化工作: AS_k首先利用ECC算法生成自身的公私钥对(a_kP, a_k), 然后将其私钥a_k与所在域内的资源服务器身份标识RID_{k-j}计算出密钥$X_{k-j} = h(RID_{k-j}||a_k)$, 最后通过安全信道将密钥$X_{k-j}$发送给对应的$RS_{k-j}$.

3.1 用户身份注册

用户身份注册指的是用户通过客户机向认证服务器进行注册. 本协议以k域用户U_{k-i}向k域认证服务器AS_k注册为例. 具体步骤如下.

(1) U_{k-i}在客户机上设置身份标识UID_{k-i}和静态口令PW_{k-i}, 通过生物特征采集器获取生物特征模板ω_{k-i}, 然后利用模糊提取技术的生成函数$Gen(\omega_{k-i})$产生对应的生物密钥R_{k-i}和生物公开信息P_{k-i}.

(2) U_{k-i}首先计算出$W_{k-i} = h(UID_{k-i}||PW_{k-i}||R_{k-i})$, 然后通过安全的信道向$AS_k$发送注册请求消息$(UID_{k-i}, P_{k-i}, W_{k-i})$.

(3) AS_k收到来自U_{k-i}的注册请求消息后, 向区块链服务节点发送查询交易请求, 服务节点查询本地存储的区块链账本. 如果账本中不存在身份标识UID_{k-i}或者UID_{k-i}对应的用户状态为“Revoke”, 则AS_k进行下一步, 否则终止会话.

(4) AS_k向区块链服务节点发送注册交易请求, 服务节点将交易广播给区块链共识节点, 共识节点通过共识机制将$(UID_{k-i}, P_{k-i}, W_{k-i})$记录到区块链账本.

(5) AS_k将$(UID_{k-i}, P_{k-i}, W_{k-i})$备份到本地数据库.

3.2 认证与密钥协商

本协议以A域用户U_{A-i}访问A/B域资源服务器$RS_{A/B-j}$为例设计同域/跨域认证与密钥协商. 当U_{A-i}想访问$RS_{A/B-j}$的数据资源时, 首先必须通过$AS_{A/B}$和$RS_{A/B-j}$的认证, 同时为了保证后续安全通信, U_{A-i}要与$RS_{A/B-j}$协商一个会话密钥. 认证与密钥协商流程图如图 6 所示, 具体步骤如下.

(1) U_{A-i}在客户机输入身份标识UID_{A-i}和静态口令PW_{A-i}', 通过生物特征采集器获取生物特征ω_{A-i}'.

(2) U_{A-i}通过客户机向区块链服务节点发送查询交易请求, 服务节点查询本地存储的区块链账本, 返回UID_{A-i}对应的生物公开信息P_{A-i}, 然后U_{A-i}对生物特征模板ω_{A-i}'进行处理, 使用模糊提取技术的再生函数$Rep(\omega_{A-i}', P_{A-i})$计算产生对应的生物密钥$R_{A-i}'$.

(3) U_{A-i}生成随机数m, 计算:

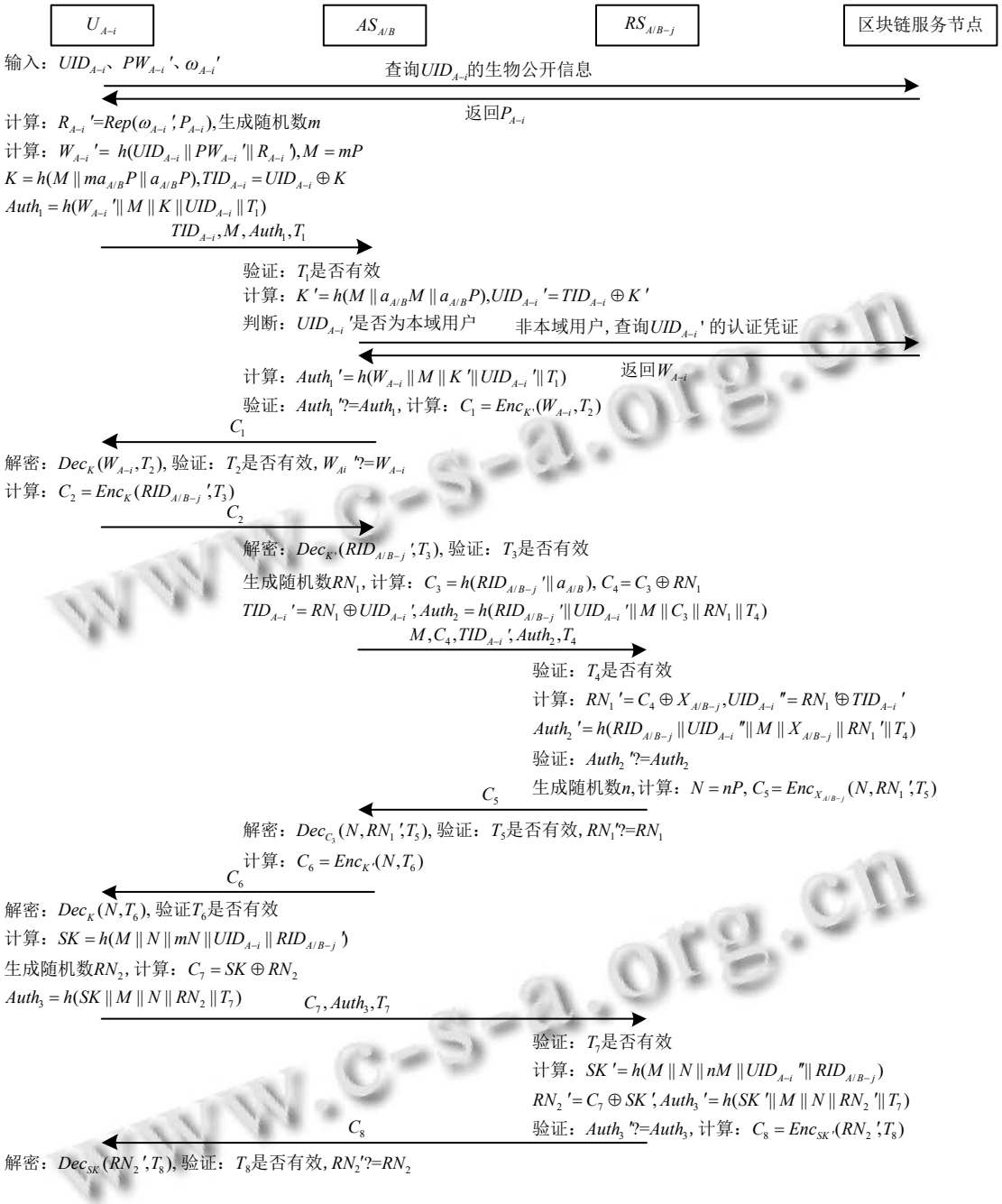

图 6 认证与密钥协商流程图

$$W_{A-i}'=h(UID_{A-i}\|PW_{A-i}'\|R_{A-i}')$$

$$M=mP$$

$$K=h(M\|ma_{A/B}P\|a_{A/B}P)$$

$$TID_{A-i}=UID_{A-i}\oplus K$$

$$Auth_1=h(W_{A-i}'\|M\|K\|UID_{A-i}\|T_1)$$

其中，T_1为当前时间戳，然后U_{A-i}将消息$(TID_{A-i}, M, Auth_1, T_1)$发送给$AS_{A/B}$.

(4) $AS_{A/B}$收到来自U_{A-i}的消息后，首先验证时间戳T_1的有效性，若不能通过验证，则终止会话，否则$AS_{A/B}$利用自己的私钥$a_{A/B}$，计算:

$$K'=h(M\|a_{A/B}M\|a_{A/B}P)$$

$$UID_{A-i}'=TID_{A-i}\oplus K'$$

(5) $AS_{A/B}$通过查询本地数据库判断UID_{A-i}'是否为本域用户身份标识, 结果分为以下两种情况.

情况一. UID_{A-i}'为AS_A所在域的用户身份标识, 此时为同域认证, AS_A直接从本地数据库获取UID_{A-i}'对应的W_{A-i};

情况二. UID_{A-i}'不为AS_B所在域的用户身份标识, 此时为跨域认证, AS_B向区块链服务节点发送查询交易请求, 服务节点查询本地存储的区块链账本, 返回UID_{A-i}'对应的W_{A-i}.

(6) $AS_{A/B}$计算:

$$Auth_1' = h(W_{A-i}||M||K'||UID_{A-i}'||T_1)$$

验证$Auth_1' = Auth_1$是否成立, 若不成立, 则终止会话, 否则$AS_{A/B}$对U_{A-i}认证成功.

(7) $AS_{A/B}$利用K'作为对称密钥, 计算:

$$C_1 = Enc_{K'}(W_{A-i}, T_2)$$

其中, T_2为当前时间戳, 然后, $AS_{A/B}$将消息C_1发送给U_{A-i}.

(8) U_{A-i}收到来自$AS_{A/B}$的消息后, 首先利用K解密C_1得到参数W_{A-i}、T_2, 然后验证时间戳T_2的有效性, 若不能通过验证, 则终止会话, 否则验证$W_{A-i}' = W_{A-i}$是否成立, 若不成立, 则终止会话, 否则U_{A-i}对$AS_{A/B}$认证成功.

至此U_{A-i}与$AS_{A/B}$的双向认证完成.

(9) 当U_{A-i}想获取$RS_{A/B-j}$中的数据资源时, 首先利用K作为对称密钥, 计算:

$$C_2 = Enc_K(RID_{A/B-j}', T_3)$$

其中, T_3为当前时间戳, 然后U_{A-i}将消息C_2发送给$AS_{A/B}$, 请求访问$RS_{A/B-j}$.

(10) $AS_{A/B}$收到来自U_{A-i}的消息后, 首先利用K'解密C_2得到参数$RID_{A/B-j}'$、T_3, 然后验证时间戳T_3的有效性, 若不能通过验证, 则终止会话, 否则$AS_{A/B}$生成随机数RN_1, 同时利用自身私钥$a_{A/B}$, 计算:

$$C_3 = h(RID_{A/B-j}'||a_{A/B})$$

$$C_4 = C_3 \oplus RN_1$$

$$TID_{A-i}' = RN_1 \oplus UID_{A-i}'$$

$$Auth_2 = h(RID_{A/B-j}'||UID_{A-i}'||M||C_3||RN_1||T_4)$$

其中, T_4为当前时间戳, 然后$AS_{A/B}$将消息$(M, C_4, TID_{A-i}', Auth_2, T_4)$发送给$RS_{A/B-j}$.

(11) $RS_{A/B-j}$收到来自$AS_{A/B}$的消息后, 首先验证时间戳T_4的有效性, 若不能通过验证, 则终止会话, 否则$RS_{A/B-j}$利用协议初始化时$AS_{A/B}$共享的密钥$X_{A/B-j}$, 计算:

$$RN_1' = C_4 \oplus X_{A/B-j}$$

$$UID_{A-i}'' = RN_1' \oplus TID_{A-i}'$$

$$Auth_2' = h(RID_{A/B-j}||UID_{A-i}''||M||X_{A/B-j}||RN_1'||T_4)$$

验证$Auth_2' = Auth_2$是否成立, 若不成立, 则终止会话, 否则$RS_{A/B-j}$对$AS_{A/B}$认证成功.

(12) $RS_{A/B-j}$生成随机数n, 同时利用$X_{A/B-j}$作为对称密钥, 计算:

$$N = nP$$

$$C_5 = Enc_{X_{A/B-j}}(N, RN_1', T_5)$$

其中, T_5为当前时间戳, 然后$RS_{A/B-j}$将消息C_5发送给$AS_{A/B}$.

(13) $AS_{A/B}$收到来自$RS_{A/B-j}$的消息后, 首先利用C_3解密C_5得到参数N、RN_1'、T_5, 然后验证时间戳T_5的有效性, 若不能通过验证, 则终止会话, 否则验证$RN_1' = RN_1$是否成立, 若不成立, 则终止会话, 否则$AS_{A/B}$对$RS_{A/B-j}$认证成功.

至此$AS_{A/B}$与$RS_{A/B-j}$的双向认证完成.

(14) $AS_{A/B}$利用K'作为对称密钥, 计算:

$$C_6 = Enc_{K'}(N, T_6)$$

其中, T_6为当前时间戳, 然后$AS_{A/B}$将消息C_6发送给U_{A-i}.

(15) U_{A-i}收到来自$AS_{A/B}$的消息后, 首先利用K解密C_6得到参数N、T_6, 然后验证时间戳T_6的有效性, 若不能通过验证, 则终止会话, 否则U_{A-i}计算其与$RS_{A/B-j}$协商的密钥:

$$SK = h(M||N||mN||UID_{A-i}||RID_{A/B-j}')$$

同时生成随机数RN_2, 计算:

$$C_7 = SK \oplus RN_2$$

$$Auth_3 = h(SK||M||N||RN_2||T_7)$$

其中, T_7为当前时间戳, 然后U_{A-i}将消息$(C_7, Auth_3, T_7)$发送给$RS_{A/B-j}$.

(16) $RS_{A/B-j}$收到来自U_{A-i}的消息后, 首先验证时间戳T_7的有效性, 若不能通过验证, 则终止会话, 否则计算其与U_{A-i}协商的密钥:

$$SK' = h(M||N||nM||UID_{A-i}''||RID_{A/B-j})$$

同时计算:

$$RN_2' = C_7 \oplus SK'$$

$$Auth_3' = h(SK'||M||N||RN_2'||T_7)$$

验证$Auth_3' = Auth_3$是否成立, 若不成立, 则终止会话, 否则$RS_{A/B-j}$对U_{A-i}认证成功.

(17) $RS_{A/B-j}$利用SK'作为对称密钥, 计算:

$$C_8 = Enc_{SK'}(RN_2', T_8)$$

其中, T_8为当前时间戳, 然后$RS_{A/B-j}$将消息C_8发送给U_{A-i}.

(18) U_{A-i}收到来自$RS_{A/B-j}$的消息后, 首先利用SK解密C_8得到参数RN_2'、T_8, 然后验证时间戳T_8的有效性, 若不能通过验证, 则终止会话, 否则验证$RN_2' = RN_2$是否成立, 若不成立, 则终止会话, 否则U_{A-i}对$RS_{A/B-j}$认证成功.

至此U_{A-i}与$RS_{A/B-j}$的双向认证完成, 同时$SK = SK'$, 密钥协商完成.

4 协议安全性与性能分析

4.1 安全性分析

本小节采用非形式化的方式分析方案所提协议的安全性, 具体内容如下.

(1) 双向认证

认证服务器$AS_{A/B}$对用户U_{A-i}的认证: $AS_{A/B}$通过验证$Auth_1' = Auth_1$是否成立, 由此间接验证$W_{A-i}' = W_{A-i}$是否成立来判定U_{A-i}的合法性, 因为只有合法的U_{A-i}才能得到与区块链账本中存储的W_{A-i}相同的登录凭证W_{A-i}', 进而计算出与$Auth_1'$相同的$Auth_1$.

用户U_{A-i}对认证服务器$AS_{A/B}$的认证: U_{A-i}通过验证$W_{A-i}' = W_{A-i}$是否成立判定$AS_{A/B}$的合法性, 因为首先只有拥有私钥$a_{A/B}$的$AS_{A/B}$才能生成正确的临时密钥K', 然后由于U_{A-i}没有将W_{A-i}'加密传输, 而是将其包含在哈希运算后的$Auth_1$中, 根据哈希函数的单向性, 无法恢复出W_{A-i}'. 只有有权访问区块链获取用户身份信息的$AS_{A/B}$通过查询区块链账本才可获取到与W_{A-i}'相同的W_{A-i}, 进而使用K'对W_{A-i}进行对称加密.

资源服务器$RS_{A/B-j}$对认证服务器$AS_{A/B}$的认证: $RS_{A/B-j}$通过验证$Auth_2' = Auth_2$是否成立判定$AS_{A/B}$的合法性, 因为除了$RS_{A/B-j}$, 只有拥有私钥$a_{A/B}$的$AS_{A/B}$通过哈希计算$h(RID_{A/B-j}||a_{A/B})$可以得到与共享密钥$X_{A/B-j}$一样的C_3, 进而计算出与$Auth_2'$相同的$Auth_2$.

认证服务器$AS_{A/B}$对资源服务器$RS_{A/B-j}$的认证: $AS_{A/B}$通过验证$RN_1' = RN_1$是否成立判定$RS_{A/B-j}$的合法性, 因为除了$AS_{A/B}$, 只有拥有共享密钥$X_{A/B-j}$的$RS_{A/B-j}$才可以根据C_4正确解密出随机数RN_1', 进而使用$X_{A/B-j}$对RN_1'进行对称加密.

资源服务器$RS_{A/B-j}$对用户U_{A-i}的认证: 依赖于认证服务器$AS_{A/B}$对U_{A-i}的认证, 因为只有合法的U_{A-i}, $AS_{A/B}$才会作为中介将$RS_{A/B-j}$的N传输过去. $RS_{A/B-j}$通过验证$Auth_3' = Auth_3$是否成立判定U_{A-i}的合法性, 因为只有接收到来自$AS_{A/B}$传输的N, 并且知道随机数m的U_{A-i}才能生成正确的协商密钥SK, 进而计算出与$Auth_3'$相同的$Auth_3$.

用户U_{A-i}对资源服务器$RS_{A/B-j}$的认证: 依赖于认证服务器$AS_{A/B}$对$RS_{A/B-j}$的认证, 因为只有合法的$RS_{A/B-j}$, $AS_{A/B}$才会作为中介将U_{A-i}的M和身份标识UID_{A-i}传输过去. U_{A-i}通过验证$RN_2' = RN_2$是否成立判定$RS_{A/B-j}$的合法性, 因为只有接收到来自$AS_{A/B}$传输的M和UID_{A-i}, 并且知道随机数n的$RS_{A/B-j}$才能生成正确的协商密钥SK', 进而根据C_7正确解密出随机数RN_2', 并最终使用SK'对RN_2'进行对称加密.

(2) 用户匿名性

本协议认证过程中, 当用户U_{A-i}向认证服务器$AS_{A/B}$传递消息$(TID_{A-i}, M, Auth_1, T_1)$时, U_{A-i}的身份标识UID_{A-i}被隐藏在临时身份标识TID_{A-i}中, 除了知道随机数m的U_{A-i}, 只有拥有私钥$a_{A/B}$的$AS_{A/B}$才能通过计算得到临时密钥K', 进而恢复出UID_{A-i}', 攻击者即使窃取到$AS_{A/B}$的公钥$a_{A/B}P$和M, 基于椭圆曲线离散对数问题的困难性, 攻击者难以计算出$a_{A/B}M$, 也就难以计算出K; 当$AS_{A/B}$向资源服务器$RS_{A/B-j}$传递消息$(M, C_4, TID_{A-i}', Auth_2, T_4)$时, U_{A-i}的身份标识UID_{A-i}'被隐藏在临时身份标识TID_{A-i}'中, 只有知道随机数RN_1的$AS_{A/B}$, 以及拥有共享密钥$X_{A/B-j}$可以计算出随机数RN_1'的$RS_{A/B-j}$才能获取U_{A-i}的真实身份标识.

在每一次新的认证过程中, U_{A-i}和$AS_{A/B}$都会分别生成一个新鲜的随机数m和RN_1, 使每次的认证消息$(TID_{A-i}, M, Auth_1, T_1)$和$(M, C_4, TID_{A-i}', Auth_2, T_4)$都不同, 从而保证用户的不可追踪性, 实现真正意义上的用户匿名性.

(3) 抗身份标识和口令猜测攻击

通过用户匿名性分析可知, 只要攻击者窃取不到以下关键数据之一: 用户U_{A-i}的随机数m, 认证服务器

$AS_{A/B}$的私钥$a_{A/B}$, 资源服务器$RS_{A/B-j}$的共享密钥$X_{A/B-j}$. 那么攻击者就无法通过TID_{A-i}或者TID_{A-i}'去验证所猜测的U_{A-i}身份标识的正确性. 基于哈希函数的单向性, 攻击者也不可能根据$Auth_1$和$Auth_2$猜测出U_{A-i}的身份标识. 而由于用户U_{A-i}的登录凭证$W_{A-i}' = h(UID_{A-i}||PW_{A-i}'||R_{A-i}')$被包含在哈希运算后的$Auth_1$中, 攻击者无法得到$W_{A-i}'$, 因此有效避免了遭受口令猜测攻击. 即使攻击者通过某种手段得到W_{A-i}', 由于U_{A-i}的身份标识UID_{A-i}被匿名保护, 并且攻击者没有U_{A-i}的生物特征ω_{A-i}', 不能恢复出R_{A-i}', 同样无法发起口令猜测攻击.

(4) 前向安全性

本协议用户U_{A-i}和资源服务器$RS_{A/B-j}$的会话密钥为$SK = SK' = h(M||N||mnP||UID_{A-i}||RID_{A/B-j})$, 由于$m$和$n$是每次密钥协商时$U_{A-i}$和$RS_{A/B-j}$随机生成的, 而在会话密钥协商结束后$m$和$n$都会被及时销毁掉, 确保攻击者不能得到任何有关这两个随机数的信息. 因此, 即使资源服务器被攻陷, 在不知道m和n的情况下, 虽然攻击者可以得到UID_{A-i}以及$RID_{A/B-j}$, 并且攻击者窃取了之前的M和N, 但是基于椭圆曲线离散对数问题的困难性, 攻击者依然难以计算出mnP, 不能得到之前的会话密钥, 基于以上分析, 本协议提供了会话密钥的前向安全性.

(5) 抗已知密钥攻击

本协议用户U_{A-i}和资源服务器$RS_{A/B-j}$的会话密钥为$SK = SK' = h(M||N||mnP||UID_{A-i}||RID_{A/B-j})$. 假设攻击者通过某种手段获取到某次会话密钥, 由于U_{A-i}和$RS_{A/B-j}$每次进行密钥协商都会生成新的随机数m和n, 使得每次的会话密钥都不同. 同时会话密钥中的用户身份标识UID_{A-i}在传输时被匿名保护, 资源服务器身份标识$RID_{A/B-j}$在传输时被对称密钥加密保护, 攻击者难以得到UID_{A-i}和$RID_{A/B-j}$. 即使攻击者窃取到M和N, 基于哈希函数的单向性, 攻击者也不能通过已知密钥推导出UID_{A-i}和$RID_{A/B-j}$的任何一个. 因此, 攻击者通过某次会话密钥并不能得到对计算出其他会话密钥有帮助的任何信息, 本协议可抵抗已知密钥攻击.

(6) 抗伪装攻击

当攻击者想伪装成合法用户U_{A-i}去访问资源服务器$RS_{A/B-j}$的数据资源时, 首先需要通过认证服务器$AS_{A/B}$的认证. 假设攻击者截获了U_{A-i}对$AS_{A/B}$的登录认证请求消息$(TID_{A-i}, M, Auth_1, T_1)$, 由于$U_{A-i}$生成的随机数$m$和$AS_{A/B}$的私钥$a_{A/B}$对攻击者都是未知的, 攻击者无法计算出$K$和$UID_{A-i}$. 同时, 由于$W_{A-i}'$被包含在哈希运算后的$Auth_1$中, 即使攻击者通过某种手段获取到$K$和$UID_{A-i}$, 因为不知道$W_{A-i}'$, 攻击者依然无法伪造$Auth_1$. 因此, 本协议可抵抗用户假冒攻击.

当攻击者想伪装成合法认证服务器$AS_{A/B}$去欺骗用户U_{A-i}和资源服务器$RS_{A/B-j}$时, 需要通过U_{A-i}和$RS_{A/B-j}$的认证, 此时攻击者必须知道$AS_{A/B}$的私钥$a_{A/B}$才能生成正确的C_1和$Auth_2$. 由于私钥$a_{A/B}$对攻击者是未知的. 因此, 本协议能够抵抗认证服务器假冒攻击.

当攻击者想伪装成合法资源服务器$RS_{A/B-j}$为用户U_{A-i}提供虚假数据资源时, 首先需要通过认证服务器$AS_{A/B}$的认证. 由于共享密钥$X_{A/B-j}$对攻击者是未知的, 除了$RS_{A/B-j}$之外, 只有$AS_{A/B}$可通过私钥$a_{A/B}$计算得到. 因此, 本协议可抵抗资源服务器假冒攻击.

(7) 抗内部攻击

本文基于区块链实现了用户身份信息的分布式存储, 部署区块链网络节点的服务器存储了包含所有信任域用户身份信息的区块链账本, 即内部攻击者可以得到任意用户的以下信息: 身份标识UID_{k-i}、生物公开信息P_{k-i}以及登录凭证$h(UID_{k-i}||PW_{k-i}||R_{k-i})$. 由于哈希函数具有单向性, 即使攻击者知道$UID_{k-i}$, 也无法还原出$PW_{k-i}$和$R_{k-i}$中的任何一个. 同时内部攻击者只知道$P_{k-i}$, 没有用户的生物特征$\omega_{k-i}$, 导致其不能恢复出生物密钥$R_{k-i}$, 进而也就不能通过离线字典攻击来猜测用户口令$PW_{k-i}$. 因此, 内部攻击者通过用户身份信息无法得到用户的生物密钥以及口令, 本协议可抵抗内部攻击.

(8) 抗重放攻击

本协议中消息的新鲜性由时间戳和验证Hash值保证. 在认证过程中, 发送方的每条消息都会附加当前时间戳, 对于对称加密的发送方消息, 由于对称密钥对攻击者是未知的, 因此攻击者不能成功解密消息, 也就无法修改消息中包含的时间戳, 只能重放截获的消息, 此时接收方只需通过验证解密后消息所含时间戳的有效性, 即可快速判断接收的消息是否为经过重放的消息; 对于没有使用对称密钥加密的发送方消息, 即使攻击者可以修改消息中的时间戳进行重放, 但在不知道消息中所用相关密钥及随机数的情况下, 攻击者不能计算出相应的$Auth$, 即验证Hash值, 同样无法通过后续的验证. 因此, 本协议可抵抗重放攻击.

4.2 性能分析

本小节从安全性和计算开销两个方面对方案所提协议的性能与已有的基于生物特征和口令双因子认证与密钥协商协议[8,9]进行对比分析.

表 2 给出了本文协议与已有协议具备的安全特性, 其中本文 I 表示同域认证与密钥协商协议, 本文 II 表示跨域认证与密钥协商协议; “√”表示协议能提供相应的安全特性, “×”表示协议不能提供相应的安全特性. 由于实现步骤几乎一样, 因此本文协议 I 与 II 具备相同的安全特性. 从表中可以看出, 文献 [8] 协议不具备用户匿名性, 该协议将用户身份标识直接明文传输, 攻击者可以轻松窃取到用户身份标识, 并对相应用户进行追踪. 由于在注册时直接将用户口令和用户生物特征发送给服务器, 内部攻击者可以获取到用户的口令和生物特征信息, 因此文献 [8] 协议不能抵抗内部攻击. 而文献 [9] 协议没有使通信双方进行相互认证, 只是单向认证. 综上, 本文协议的安全性最高.

表 2 安全性比较

安全特性	文献[8]	文献[9]	本文 I	本文 II
双向认证	√	×	√	√
用户匿名性	×	√	√	√
抗猜测攻击	√	√	√	√
前向安全性	√	√	√	√
抗已知密钥攻击	√	√	√	√
抗伪装攻击	√	√	√	√
抗内部攻击	×	√	√	√
抗重放攻击	√	√	√	√

表 3 比较了本文协议与已有协议的计算开销, 其中本文 I 表示同域认证与密钥协商协议, 本文 II 表示跨域认证与密钥协商协议. 从表中可以看出, 本文协议和已有协议的点乘运算次数相同, 都是 6 次, 这是因为所有的协议都是基于椭圆曲线离散对数问题的困难性来实现认证与密钥协商. 本文协议 I 、II 和文献 [8] 协议相比, 除了最后两项指标外, 前 3 项指标的计算开销基本相当. 额外计算开销的主要原因是本文协议涉及用户、认证服务器与资源服务器三方交互, 而文献 [8] 协议只是用户与服务器的两方交互. 同时由于本文利用区块链来存储用户身份信息, 而与区块链交互需要通过交易, 交易需要签名, 因此本文协议 I 与 II 分别多使用了 1 个和 2 个签名的计算开销用于查询区块链获取用户相关信息. 协议 I 比协议 II 少一个签名的原因是: 本域用户身份信息在区块链上的更改由本域认证服务器负责, 本域认证服务器可以备份到最新的本域用户身份信息. 使得同域认证时, 本域认证服务器不必访问区块链, 直接访问备份数据库获取本域用户身份信息即可. 而跨域认证时, 认证服务器必须访问区块链才能获取其他域的最新用户身份信息, 因此协议 II 需多 1 个签名开销. 本文协议 I 、II 和文献 [9] 协议相比, 少了 7 次哈希运算, 多了 10 次对称加/解密运算, 同时分别少了 3 次和 2 次非对称签名运算. 虽然对称加/解密运算开销比哈希运算开销大, 但是签名运算开销远大于对称加/解密运算, 因此整体上本文协议的计算开销比文献 [9] 协议小. 本文协议对称加/解密运算次数多的原因是为了保证强安全性, 在通信双方认证后, 协议使用双方的共有密钥对称加密通信. 虽然文献 [9] 协议同样涉及三方交互, 包括网关节点、传感器节点和用户, 但该协议只是单向认证, 而本文协议实现了用户、认证服务器以及资源服务器三方之间的两两相互认证. 此外, 文献 [9] 相比本文协议 I 与 II 还多了 3 次和 2 次交易操作用于从区块链获取信息.

表 3 计算开销比较

协议	点乘运算	哈希运算	MAC 运算	非对称签名运算	对称加/解密运算
文献[8]	6	10	2	0	0
文献[9]	6	20	0	4	0
本文 I	6	13	0	1	10
本文 II	6	13	0	2	10

综合以上分析, 本文方案所提协议能够以更少的计算开销, 提供更强的安全性.

5 方案优点

(1) 避免用户生物特征被泄露

本文方案使用模糊提取技术将用户生物特征转换为生物密钥R和生物公开信息P用于认证, 由于根据R和P无法恢复出用户生物特征ω, 并且每次用户输入生物特征ω完成模糊提取操作之后都将其及时删除, 使用户生物特征ω没有进行传输, 也没有存储在任何设备上. 因此, 本文方案可避免用户生物特征被泄露.

(2) 保证用户身份信息一致

本文方案通过认证服务器向区块链服务节点发送交易请求, 服务节点广播交易给区块链共识节点, 共识节点通过共识机制将交易涉及的用户身份信息记录到区块链账本中. 由于共识机制实现了全网节点对交易

执行结果的共识, 因此可保证所有节点存储的区块链账本中用户身份信息的一致. 并且本文使用的共识机制是 PBFT, 可有效防止可能存在的恶意节点阻碍全网节点达成共识的行为.

(3) 提高认证的可靠性

由于本文方案基于区块链实现了用户身份信息的分布式存储, 每个域的区块链服务节点都存储了包含所有信任域用户身份信息的区块链账本. 无论是同域认证还是跨域认证, 客户机和认证服务器都可以通过与服务节点进行交互, 去获取所需的用户身份信息. 而各个信任域内都部署有服务节点, 如果客户机和认证服务器当前请求的服务节点响应超时的话, 可以及时切换到另一个服务节点进行请求. 对比文献 [13] 和文献 [14] 的跨域认证方案, 认证域的认证服务器需要与用户注册域的认证服务器进行交互, 请求获取跨域用户的身份信息或认证结果, 而由于注册域的认证服务器可能发生繁忙、宕机等状况, 导致未能及时响应认证域的认证服务器, 进而使得跨域认证时间过长甚至跨域认证失败. 综上所述, 本文方案可以提高认证的可靠性.

6 结论与展望

本文为了保证跨域访问数据资源的数据共享场景下用户的身份合法性以及安全通信, 提出了一种基于区块链的生物特征和口令双因子跨域认证与密钥协商方案. 方案利用区块链实现用户身份信息的分布式存储, 使得认证服务器通过查询区块链账本即可得到用户的身份信息, 进而实现认证服务器在跨域认证时无须与用户注册域的认证服务器交互, 即可对跨域访问的用户进行认证, 提高跨域认证的可靠性. 下一步将优化区块链共识机制, 提高区块链网络的共识效率, 降低区块链网络的通信开销.

参考文献

1 张馨之. 互联网+电子病历档案大数据跨医院共享信息安全保护机制探究. 中国医药导报, 2017, 14(27): 189–192.

2 霍炜. 政务数据资源共享是区块链技术自主创新的主战场. 信息安全与通信保密, 2021, (1): 2–13.

3 刘敖迪, 杜学绘, 王娜, 等. 基于区块链的大数据访问控制机制. 软件学报, 2019, 30(9): 2636–2654. [doi: 10.13328/j.cnki.jos.005771]

4 毋立芳, 马玉琨, 周鹏, 等. 生物特征模板保护综述. 仪器仪表学报, 2016, 37(11): 2407–2420. [doi: 10.3969/j.issn.0254-3087.2016.11.001]

5 张宁, 臧亚丽, 田捷. 生物特征与密码技术的融合——一种新的安全身份认证方案. 密码学报, 2015, 2(2): 159–176.

6 Dodis Y, Reyzin L, Smith A. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. International Conference on the Theory and Applications of Cryptographic Techniques. Interlaken: Springer, 2004. 523–540.

7 杨得新, 杨波, 郭艾侠. 基于生物特征和口令放大的远程认证协议. 计算机工程与应用, 2010, 46(30): 108–111.

8 李晓伟, 杨邓奇, 陈本辉, 等. 基于生物特征和口令的双因子认证与密钥协商协议. 通信学报, 2017, 38(7): 89–95. [doi: 10.11959/j.issn.1000-436x.2017148]

9 张晓薇, 李洪赭, 孙广成, 等. 基于区块链的 WSN 用户认证和密钥协商方案. 计算机与数字工程, 2020, 48(11): 2717–2722. [doi: 10.3969/j.issn.1672-9722.2020.11.035]

10 Wang WT, Hu N, Liu X. BlockCAM: A blockchain-based cross-domain authentication model. 2018 IEEE 3rd International Conference on Data Science in Cyberspace (DSC). Guangzhou: IEEE, 2018. 896–901.

11 张金花, 李晓伟, 曾新, 等. 边缘计算环境下基于区块链的跨域认证与密钥协商协议. 信息安全学报, 2021, 6(1): 54–61.

12 张亚兵, 邢镔. 基于多层区块链的跨域认证方案. 计算机应用研究, 2021, 38(6): 1637–1641.

13 周致成, 李立新, 郭松, 等. 基于区块链技术的生物特征和口令双因子跨域认证方案. 计算机应用, 2018, 38(6): 1620–1627.

14 张昊迪, 刘国荣, 汪来富, 等. 基于区块链技术的跨域身份认证机制研究. 广东通信技术, 2018, 38(7): 23–31. [doi: 10.3969/j.issn.1006-6403.2018.07.005]

15 张平, 栗亚敏. 前向安全的椭圆曲线数字签名方案. 计算机工程与应用, 2020, 56(1): 115–120. [doi: 10.3778/j.issn.1002-8331.1810-0283]