IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 November 2022, accepted 3 January 2023, date of publication 9 January 2023, date of current version 12 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3235372

RESEARCH ARTICLE

MA-TEECM: Mutual Anonymous Authentication-Based Credential Migration Technology for Mobile Trusted Execution Environments

ZIWANG WANG, LIANG WANG, AND HUILI YAN
School of Artificial Intelligence and Big Data, Hefei University, Hefei 230022, China

Corresponding author: Ziwang Wang (wangzw@hfuu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 72271085, in part by the Talent Research Fund Project of Hefei University under Grant 21-22RC17, and in part by the University Natural Sciences Research Project of Anhui Province under Project 2022AH051803.

ABSTRACT ARM TrustZone is the most widely used mobile trusted execution environment (TEE) technology today. Its hardware-enabled isolated execution environment provides reliable assurance of secure storage of credentials in mobile devices. However, the research on managing credentials stored in the TEE throughout the lifecycle of mobile devices has received little attention in recent years, and the credentials in TEE generally face usability problems caused by the mobile device lifecycle events. Aiming at the risk of information disclosure caused by the third-party service providers in the traditional credential migration scheme, this paper presents a mutual anonymous authentication-based credential migration framework for mobile trusted execution environments. First, we propose a peer-to-peer credential migration model between mobile terminals based on TrustZone and SGX, which solves the single point of failure caused by attacks on trusted third parties that act as credential transfer stations and managers in traditional solutions; Second, we propose an identity authentication protocol between TEEs based on mutual anonymous authentication, and a detailed authentication process is designed based on the universal mobile TEE model; Third, we build a formal verification model using High-Level Protocol Specification Language (HLPSL). Finally, the formal and informal security analysis indicate that the improved scheme meets the expected security requirements and is secure against several known attacks.

INDEX TERMS Credential migration, trusted execution environments, mutaul authentication.

I. INTRODUCTION

Arm partners have shipped more than 232.4 billion Arm-based processor chips by mid-2022 [1], [2], which are widely used in mobile Internet devices such as mobile phones, tablet computers, and smartwatches. As mobile devices are more and more commonly used in business, finance, and information technology, the coexistence of sensitive data and normal data on mobile terminals is becoming very common. For example, Bring Your Own Device (BYOD) is a policy that allows employees to use their personal mobile devices to access office areas to process corporate data and login Intranet applications [3]. Many enterprises accept it by creating secure containers on employees' personal mobile devices to ensure data security. However, because sensitive data, such as user credentials, are tightly coupled with mobile devices, when an user tries to migrate data to a new device due to a device's lifecycle events (such as terminal replacement or employee separation), the user usually needs to manually re-register

The associate editor coordinating the review of this manuscript and approving it for publication was Diana Gratiela Berbecaru.

credentials acquired in various scenarios to the new devices one by one, instead of migrating directly from the old terminal to the new.

Credentials are the evidence that lets entities access privileged data and services, such as user keys, certificates, and other authentication information. As the device's usage time accumulates, a considerable amount of credentials will be stored in the trusted execution environment (TEE) [4] of the mobile device, which poses several challenges to the credential management of the mobile terminal. First, traditional user passwords are vulnerable to phishing and dictionary attacks, and key management software based on TEE is gradually gaining popularity to obtain more secure and convenient password management functions. For example, the Keystore system component has been introduced since Android 4.0, which makes the keys independent of the application or even the operating system. That is, the user can encrypt, decrypt and manage the key through the Keystore API without obtaining the key, which significantly improves the security of the keys. However, it also increases the cost for users to reconfigure keys. With the growth of the number of keys, it is no longer feasible to manually reconfigure keys on new terminals; Second, with the rapid development and broad application of artificial intelligence technology, the machine learning process has been introduced in increasingly digital credentialing systems. For example, in all series of iPhone devices, the fingerprint and face print data stored in the TEE will be gradually strengthened over time, and if users cannot migrate this credential directly, it will take some time to relearn in the new terminal; Finally, digital assets stored as credentials are gaining popularity, such as cryptocurrencies, NFT, and digital copyright certificates. Users urgently need a solution to automatically migrate their credential files to the new terminal when replacing devices. Therefore, it is necessary to migrate the credentials between devices considering device lifecycle events.

The security requirements for credential migration in mobile TEE scenarios can be summarized as follows: a) Ensure the integrity of the device trust root, b) Protect the confidentiality of credentials from unauthorized access, and c) Maintain confidentiality and integrity of sensitive processes. In addition, the process of credential migration is always accompanied by the deletion of the original credentials, so one-way security verification cannot meet the required security. We proposes a scheme that allows peer-to-peer credential migration between personal mobile devices to address the above challenges. The motivation is to provide users with a credential migration solution with enhanced usability and reduce the security risks that the credential migration scheme may pose. Our scheme is also based on a ''server-client'' interaction model, where a secure session is established between communicating entities through strict attestation of identity and integrity. The difference is that the server is completely isolated from the credentials during the migration process and is only used to assist in establishing a TEE-to-TEE secure transmission channel with mutual authentication.

The major contributions of this paper are as follows:

- We propose a LAN-oriented credential migration model for personal mobile devices that securely migrates credentials stored in a TEE from one device to another;
- We propose a mutual authentication protocol based on an improved direct anonymous authentication scheme, which replaces the traditional strategy of using a trusted third party to manage or relay credential transmission; that is, the credential will not be saved to any third party;
- We formally model the proposed protocol using the HLPSL formal language and verify the protocol model using the AVISPA automated verification tool.

II. RELATED WORK

The TEE credential migration refers to transferring and reloading credential data between different TEEs. Credential migration services can save significant device re-initialization overhead and are critical for lifecycle events such as mobile device replacement. However, the standard TEE implementation today still cannot solve the problem of credential migration very well.

The key migration issue first appeared in the research on the Trusted Platform Module (TPM), which is an essential part of TPM 1.2 and 2.0 specifications [5], and many researchers have proposed various methods to improve it [6]. However, research on key or credential migration for mobile TEE has not received sufficient attention.

Based on a public resource known as the Open Certificate Platforms (OCP), Kari et al. [7] proposed a trusted domain certificate migration protocol. They recommended encrypting and backing up the credentials on a trusted server with a password known only to the user and then completing the credential migration by entering the password again. The protocol framework does not require complex user interaction and authentication processes, however, all user credentials must be stored in the server in clear text, and the migration process becomes the process of reconfiguring the backup files in the server. Although a key known only by users protects the process, the architecture lacks a discussion on the identity authentication between the OCP and the two devices' TEE. There is a privacy breach due to the service provider's full access to user credentials and personal data. Arfaoui et al. [8] propose a privacy-preserving scheme for migrating credentials between Global Platform TEEs, which requires dynamic interaction between service providers and TEE managers. Although the authors mention that the service provider must authenticate the TEE, the migration protocol does not provide a specific identity certification procedure, and the necessity of mutual authentication between the service provider and the TEE is not covered. Similarly, Literature [9] and [10] implement identity authentication management between credential migration devices through a trusted service provider. Carlton et al. [11] demonstrated the necessity of mutual authentication in the credential migration

service for the first time, and used formal tools to model their proposed mutual authentication protocol, proving the security of the protocol process. Tan and Song [12], [13] proposed a key migration protocol that supports mutual authentication between trusted roots, which achieves identity binding of both migration parties by adding device attributes in the authentication process between the source and target devices to the service provider. Nishimura et al. [14] propose using a trusted third party to identify the owner of a personal device to prevent the sharing of authentication keys to malicious nodes. The literature mentioned above, however, all needs to assume that the third-party service provider is trusted.

Instead of migrating specific credential files, recent researchers prefer to migrate the TEE itself as an entire [15], [16], [17], [18], [19]. It not only realizes the migration of security execution context, but also avoids the problem of reconfiguration after credential migration. For example, Gu et al. [18] proposed a microkernel architecture-based enclave coding model that supports secure migration of enclaves between heterogeneous hardware platforms such as ARM TrustZone and Intel SGX. However, such research focuses on designing reasonable migration models and addressing compatibility issues, ignoring identity and integrity attestation to ensure the platform and data security. Fortunately, the works in this study can serve as a complement to these research.

To create a peer-to-peer credential transmission channel between TEEs, Intel SGX technology is introduced. This approach addresses the mutual trust between nodes sending and receiving credentials and the information leakage that third-party service providers may cause.

III. PRELIMINARIES

A. ARM TrustZone

ARM TrustZone [20] is a hardware solution for ARM processors to implement trusted execution environment technology. Two completely isolated execution environments, the secure world and the normal world, are virtualized through hardware assistance. The TrustZone framework utilizes a trusted bootloader stored in independent read-only memory as the trusted root and implements the authentication and initialization of trusted components based on the trusted root to create a complete chain of trust and ensure the security of the entire system.

The ARM TrustZone architecture is shown in Figure 1, where the TrustZone technology defines two distinct and independent execution contexts. ARM CPU features unique register sets for the two worlds, guaranteeing that the chain of trust can successfully be passed to the kernel and the Trusted Application (TA) in the secure world. Generally, the secure world has higher privileges than the normal world. TrustZone aims to provide security assurance for mobile terminal devices at a lower cost, realize a transparent security environment isolated from the general execution

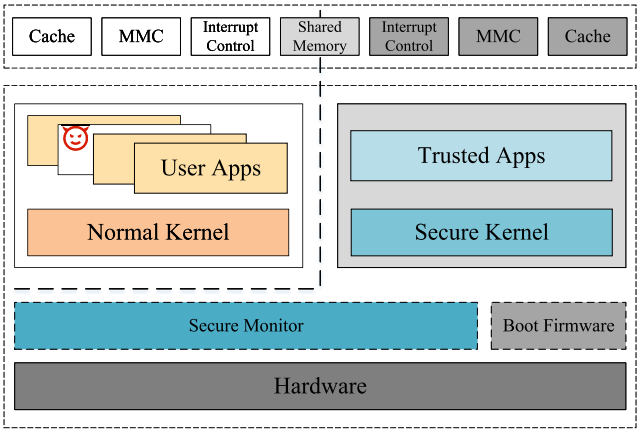

FIGURE 1. ARM TrustZone architecture.

environment, and can resist software attacks and some low-cost hardware attacks. Therefore, the isolation feature of TrustZone makes it an ideal choice for storing users' private data.

B. DIRECT ANONYMOUS ATTESTATION

Remote attestation is the process by which a trusted computing platform proves to external entities that it has a legitimate root of trust and is in a trusted operating state. The standard signature mechanism makes it easy for the verifier to distinguish the identity of the prover, which has the problem of privacy disclosure. Therefore, Trusted Computing Group (TCG) adopted the Direct Anonymous Attestation (DAA) technology proposed by Brickell et al. [21] as the identity authentication method of the TPM in the TPM1.2 technical standard. DAA is an attractive encryption scheme that provides a balance between platform authentication and anonymity, which make the TPM can directly prove the authenticity of the trusted computing platform to the verifier without the participation of a trusted third party. Furthermore, a DAA certificate can be used for multiple identifications and to guarantee anonymity. When the TPM proves platform's authenticity to the verifier, the secret values and messages are signed using the DAA certificate, and the verifier confirms the legitimacy of the TPM's identity based on the signature. However, DAA is a process in which the provider unilaterally proves to the challenger, the authenticity of the challenger's identity cannot be verified, and the security of data cannot be guaranteed in the credential migration scenario.

IV. THREAT MODEL AND ASSUMPTIONS

MA-TEECM uses a traditional three-party communication transmission model to build a flexible LAN credential migration framework: two trusted key management applications (KMA) in the source terminal and the target terminal and a migration management application (MGA) running on the PC, where KMA and MGA run in the trusted execution environment of ARM TrustZone and Intel SGX, respectively.

In this network model, security risks mainly come from the following four aspects:

i) Security Risks of MGA. Currently, MGA is usually provided by the manufacturer of the mobile device. Users can only confirm that the application comes from the issuer by verifying its hash. However, there is no way to guarantee whether the program is vulnerable or whether the producer has backed up the user's identity information, such as biometric information, for some reason.

ii) Security Risks of source device's KMA. On the one hand, the ARM TrustZone TEE is not unbreakable at this stage, and TAs running in it may still be affected by vulnerabilities in other TAs, trusted OSes, or even the TrustZone security mechanism itself. On the other hand, TEE cannot communicate independently from REE, and TEE generally does not support trusted UI and trusted input. As a result, TEEs cannot directly transfer credentials to each other, and their security is vulnerable to REEs. Therefore, when the source device's KMA establishes a connection, the malicious process or even the KMA program itself may still use protocol vulnerabilities to transmit malicious credentials to the receiver, causing the receiver to lose the ability to identify the connection to the sender.

iii) Security Risks of target device's KMA. If the TEE of the target device is vulnerable, user credentials will be leaked directly to the Internet after being transferred to the target device.

iv) Security Risks of the communication channel. The data transmission of mobile terminals is exposed to insecure channels and faces typical LAN attack vectors such as sniffing, masquerading, and replay.

Therefore, this paper mainly considers the following attacker models:

- Assuming a Dolev-Yao attacker model exists in the communication channel, the attacker can not only eavesdrop, block, and intercept all the information flowing through the network but also perform attacks on keys and protocols. For example, attackers can manipulate data transfers between entities and tamper with data;
- Assuming that an attacker can compromise the TEE of the user's device, including the trusted OS with the TA, it is necessary to verify the integrity of the TEE;
- Assuming that an integrity-authenticated TEE can provide sufficient protection to the credentials stored in it. Even though the attacker can physically access the mobile device, the protected data in the TEE cannot be read;
- Assuming that the TEE can verify the integrity of the user program of the REE, even if the attacker can destroy the system environment of the sender and receiver at runtime, the running result of the agent program can be guaranteed to be correct;
- We do not consider DoS attacks and resource exhaustion attacks.

V. MA-TEECM ARCHITECTURE

Considering the target model, the attacker model, and the Global Platform TEE specification, this paper proposes a novel model MA-TEECM, for TEE credential migration based on mutual anonymous authentication. Specifically, a new group manager (GM) participant is introduced between the source TEE and the target TEE. GM is an enclave program running in Intel SGX, responsible for verifying the integrity of the access device's TEE, creating group signatures, and issuing group membership certificates for the source TEE and target TEE. With the assistance of the GM, a shared interaction channel is created for any legitimate TEE.

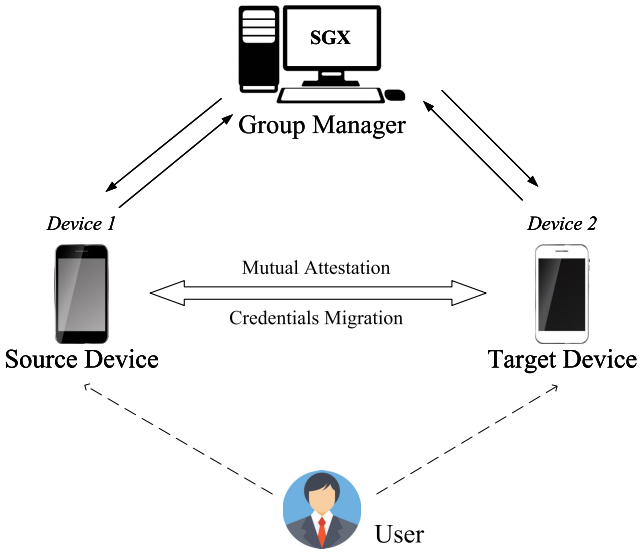

FIGURE 2. Credential migration network model.

We recommend that users implement credential migration between mobile devices in a peer-to-peer manner to prevent remote attackers from compromising key infrastructure. The peer-to-peer communication network model is shown in Figure 2. GM verifies the integrity of the TEE fingerprint of mobile devices and issues group member certificates to all nodes that pass the verification. Then group members sign the message through group signature to realize identity authentication. In brief, MA-TEECM divides the authentication in the credential migration process into two parts: the integrity authentication of the mobile node by the GM and the identity authentication between the nodes. It means that a common GM node can build a credential migration environment for all manufacturers' mobile terminals. Finally, this paper constructs a specific credential migration protocol, which implements the complete identity authentication process and the secure transfer of credential files from one device to another.

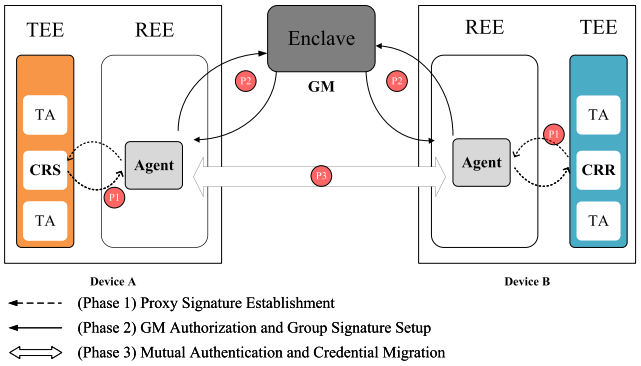

FIGURE 3. Proposed model and procedures for credential migration.

A. SYSTEM MODEL

The architecture of the proposed credential migration model is presented in Figure 3. There are two types of participants in our MA-TEECM architecture, one is the domain manager GM as the migration group administrator, and the other is the mobile device entity containing a TEE. A temporary group established by the GM for the target TEE and the source TEE is called the migration group. More specifically, first we use an open-source group signature key management system running in SGX as the GM model. Only devices registered with the GM can initiate or accept group-signed credential migration requests; Next, based on an improved direct anonymous authentication protocol, we propose a platform identity-based mutual authentication scheme; Furthermore, we symbolically declare the complete credential migration process. In summary, MA-TEECM presents how to perform credential migration between two different TEEs, thus addressing the possible security issues in related work.

B. CERTIFICATE MIGRATION PROTOCOL DESIGN

To migrate the credential file stored in the TEE from the source device to the target device, the user needs to connect the two devices to the LAN with a GM node simultaneously to establish point-to-point communication between the GM and the migration device. The mobile terminals participating in the migration all run a TEE that has undergone integrity verification, and a trusted TA runs inside the TEE as the executor of the migration process. That is, the Credential migration Request Sender (CRS) in the source TEE and the Credential migration Request Receiver (CRR) in the target TEE. And since the TEEs cannot communicate with each other directly, an agent program needs to be set up in the non-secure domain as the interactive entrance of the TA. Finally, we run a group key management program in the Intel SGX enclave as the GM, who generates and manages the group signature system and maintains the identity certificates of group members. In summary, two TEE-deployed mobile devices obtain group membership through the GM and complete initial verification. The user then initiates a credential migration requester within the source TEE.

MA-TEECM is mainly divided into the following three phases: the proxy signature authorization phase, the group signature establishment phase, and the mutual authentication phase.

1) PROXY SIGNATURE AUTHORIZATION PHASE

When the user initiates a credential migration in some way, the TA (CRS/CRR) in the migration device's TEE and the authentication agent running in the normal world first complete the verification of the proxy signature, that is, the TA_i grants the signature authority to its designated $Agent_i$. This approach ensures the signatures between TA and Agent are unified in the subsequent protocol interaction process.

2) GROUP SIGNATURE ESTABLISHMENT PHASE

The source TEE and the target TEE obtain the group member certificate from the GM respectively through the integrity attestation, and the group signature is used to verify the legitimacy of the group membership. Assuming that the GM can obtain the Hash value of the TEE environment and verify its integrity when issuing a certificate to the TEE, that is, the TEE is considered to have obtained a legitimate group membership.

3) MUTUAL AUTHENTICATION AND CREDENTIAL MIGRATION PHASE

This phase requires the source TEE and the target TEE to verify the authenticity of each other's identity information. Therefore, the migration device will construct a mutual direct anonymous authentication between CRS and CRR using the legitimate group membership certificate. Finally, the credential ciphertext is transferred from the source TEE to the trusted storage of the target TEE by the migration key established in the mutual authentication phase.

The detailed description of the protocol process involving the source TEE and the target TEE is as follows:

A typical MA-TEECM model consists of three participants: the migration group manager GM, the migration handler TA running in the TEE, and the Agent as its interaction portal. The migration process includes two polynomial algorithms (set up and verify) and four interaction protocols (bind, join, sign and migrate).

- **Bind**. The migration manager TA located in the secure world starts and verifies the migration Agent in the normal world. We assume that TA can perform continuous integrity checks on the client application process under the standard TrustZone architecture [22]. Then, TA constructs a proxy signature and sends the proxy certificate (σ, K) to Agent, and Agent uses the signature to sign the communication message after successful verification.
- **Setup**. The user starts an SGX enclave program GM on the LAN host, GM builds the CL group signature system and constructs the DAA parameters based on it.
- **Join**. Agent establishes a connection with GM through (σ, K), and then TA sends the system environment hashs

(similar to platform configuration register (PCR) [23]) signed by K to GM. GM verifies the integrity and issues group membership certificates for Agent and TA. Finally, TA generates a private secret value, and GM issues the CL signature on it, that is, TA obtains the DAA certificate based on a secure two-party protocol.

- **Sign**. TA uses the DAA certificate and the secret value to sign the message m.
- **Verify**. The migration sender CRS and receiver CRR verify the legitimacy of each other's identity information through a deterministic algorithm. That is, after entering the signature c and the verifier's public key, the deterministic algorithm will return a decision of acceptance or rejection.
- **Migrate**. After establishing the communication channel between CRS and CRR through mutual authentication, CRS encrypts the credential file with its private key and the public key of the receiver CRR and transmits it, which means the migration of credentials is completed.

VI. IDENTITY AUTHENTICATION PROTOCOL DESIGN

The TCG TPM 2.0 Library Specification clearly states that the root of trust must satisfy anonymity in the authentication and remote attestation process, and the TPM must generate a different session key with the verifier for each authentication interaction [24]. Therefore, based on the direct anonymous authentication protocol, this paper designs a mutual authentication algorithm for credential migration.

The identity authentication protocol of the MA-TEECM is shown in Figure 4. The participants of the protocol include the migration group manager GM, the migration requester CRS, and the credential receiver CRR. Both CRS and CRR contain a trusted application TA and an authentication Agent. TA runs in the TEE and is responsible for processing credential migration's interaction process and verifying other participants' identity information. GM is an open-source cryptography software entity running in the SGX environment, offering transparent system security parameter creation services and granting other entities group membership certificates. Agent is a client application running outside the TEE, providing interactive entry and agent signature services for TA.

To facilitate the symbolic description of the authentication protocol, this paper abstracts the MA-TEECM protocol into three phases: proxy signature authorization, migration group build, and mutual authentication, where the implementation of the credential transfer is included in the mutual authentication phase, and the encryption and decryption process of the credential by the TA is ignored. In fact, the credential must be stored encrypted by the trusted root's endorsement key (EK) or the storage key protected by EK. The mathematical notations used in MA-TEECM and their descriptions are shown in Table 1. The specific process is as follows:

TABLE 1. Notations and descriptions.

Notation	Description
g_1, g_2	Random Generator
p_1, p_2, q_1, q_2	Big Primes
(x, V)	Origin Signature
(σ, K)	Proxy Signature
X, Y	Integer Constants
l_c, l_s, l_b	Security Parameters
(E, s)	Group Membership Certificate
(PK, SK)	Public Key and Private Key of the Trust Root

A. PROXY SIGNATURE AUTHORIZATION

First, TA_i verifies the availability of Agent_i by checking the integrity of its process and issues a proxy signature certificate $\{\sigma, K\}$ for the verified Agent_i, and then Agent_i verifies the legitimacy of the certificate and uses it to secure subsequent communication. Silimar to literature [25], the specific process is as follows:

Step 1: (x, V). TA_i randomly select the big primes p_1, q_1, such that $q_1 | p_1 - 1$ and $g_1 \in Z^*_{q_1}$ is a generator which order is q_1. Generate the original signature private key $x \in_R Z_{p_1-1}$, and the corresponding public key is $V = g_1^x \bmod p_1$. Where V, g_1 is disclosed to integrity-verified Agent_i and potential signature verifiers.

Step 2: (σ, K). TA_i generates random numbers $k \in_R Z_{p_1-1}$, and calculates $K = g_1^k \bmod p_1$ and $\sigma = (x + kK) \bmod (p_1 - 1)$. Finally, send $\{\sigma, K\}$ to Agent_i over a secure channel.

Step 3: Proxy certificate verification. Agent_i verify $g_1^\sigma \stackrel{?}{=} VK^K \bmod p_1$. If holds, Agent_i will become a legal proxy, otherwise, Agent_i rejects the signature and terminates the protocol.

B. MIGRATION GROUP BUILD

This phase is completed by CRR and CRS interacting with group administrator GM, respectively. First, GM verifies the platform's integrity and builds group membership for legitimate platforms, including the process of signing the message by the platform (TA/Agent) using the proxy signature $\{\sigma, K\}$ and verifying the signature by the GM. Next, the integrity of TA is verified by the enclave program in GM. Then, TA obtains the group membership certificate $\{E, s\}$ through the *Join* protocol of the group signature scheme. The specific process is as follows:

Step 1: n, p_2, q_2. Randomly select the big primes p_2, q_2, calculate $n = p_2 q_2$, where p_2, q_2 are greater than β bits. That is $p_2, q_2 > 2^\beta$. And $p_2 = 2p' + 1$, $q_2 = 2q' + 1$, where p', q', p_2, q_2 are both prime numbers. Among then, β is the security level parameter set according to security requirements and g_2 is a random generator on the quadratic residual group QR_n.

Step 2: $X, Y, \alpha, l_c, l_s, l_b$. Randomly select the integer constants $\alpha, l_c, l_s, l_b \in [1, p'q']$, and $Y > 2^{\alpha(l_c+l_b)+1}$, $X > 2Y + 2^{\alpha(l_s+l_c)+2}$.

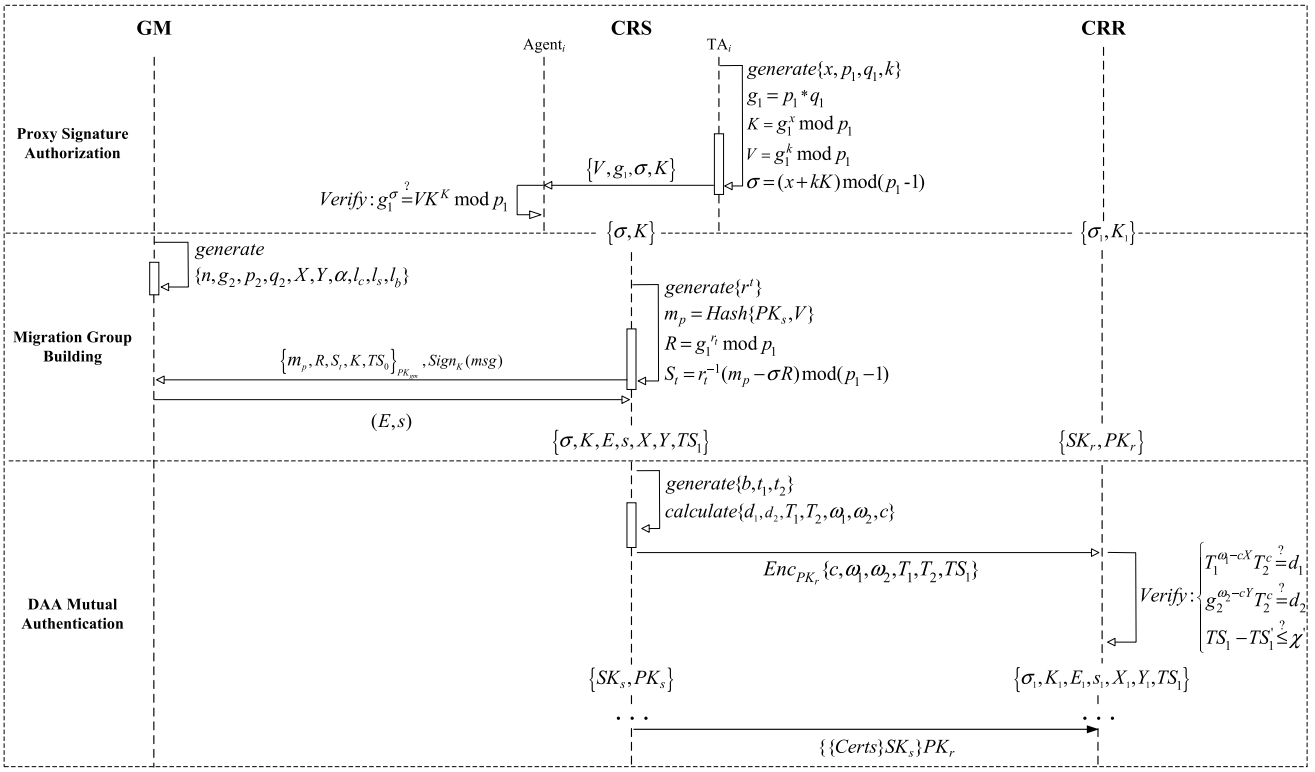

FIGURE 4. The authentication phases of this scheme.

Step 3: m_p, R, S_t. First, generating the proxy signature for the platform identity information and public key information. Select random number $r_t \in_R Z_{P_I-1}$, calculate $m_p = Hash(PK_s, V)$, $R = g_1^{r_t} \bmod p_1$ and $S_t = r_t^{-1}(m_p - \sigma R) \bmod (p_1 - 1)$, where one-way hash function $Hash$ satisfies $Hash : \{0,1\}^* \rightarrow \{0,1\}^l c$, and PK_s is the public key of the prover. Then, encrypt m_p, R, S_t, K and transfer to GM.

Step 4: Proxy signature verification. GM decrypt with private key to obtain m_p, R, S_t, K, calculate $m_p = Hash'(PK_s, V)$ and $V' \equiv VK^K \bmod p_1$ to verify $g_1^{m_p} \stackrel{?}{=} R^{S_t} V' \bmod p_1$. If holds, GM will next check if PK_s, V matches, otherwise, terminates the protocol.

Step 5: (E, s). The verified platform joins the migration group and obtains the group membership certificate (E, s). Where $s \in (X, X + 2^{l_s})$, s is prime number, and $E^s \equiv g_2 \bmod n$.

C. DAA MUTUAL AUTHENTICATION

The mutual authentication phase can be divided into two separate parts, namely the two independent phases of the requester CRS proving to the receiver CRR and CRR proving to CRS. The details are as follows:

Step 1: b, t_1, t_2.TA$_s$ generate random number $b \in [Y - 2^{l_b}, Y + 2^{l_b}]$, $t_1 \in \{0,1\}^{\alpha(l_s+l_c)}$ and $t_2 \in \{0,1\}^{\alpha(l_b+l_c)}$.

Step 2: $c, \omega_1, \omega_2, T_1, T_2$. Calculate $T_1 = E^b \bmod n = g_2^{s^{-1}b}$, $T_2 = {g_2}^b \bmod n$, $d_1 = T_1^{t_1} \bmod n$, and $d_2 = {g_2}^{t_2} \bmod n$ [26]. For any message m, compute $c = Hash(g_2||T_1||T_2||d_1||d_2||m||TS_1)$, $\omega_1 = t_1 - c(s - X)$, and $\omega_2 = t_2 - c(b - Y)$. Finally, send $c, \omega_1, \omega_2, T_1, T_2$ to the verifier.

Step 3: Signature Verification. Verifier received $c, \omega_1, \omega_2, T_1, T_2$, then recalculate $c' = Hash(g_2||T_1||T_2 ||T_1^{\omega_1 - cX} T_2^c ||g_2^{\omega_2 - cY} T_2^c||m||TS_1)$. Then accepts the signature if and only if both of $c = c', \omega_1 \in \pm\{0,1\}^{\alpha(l_s+l_c)+1}$ and $\omega_2 \in \pm\{0,1\}^{\alpha(l_b+l_c)+1}$ are all satisfied.

Step 5: If Agent$_R$ successfully verifies Agent$_S$, then Agent$_S$ becomes the new verifier, Agent$_R$ becomes the new provider. Repeat steps 1-4 for mutual authentication.

VII. SECURITY ANALYSIS

A. USER IDENTIFICATION

Identifying whether the target device belongs to the source device owner is critical in the credential migration. MA-TEECM is a wireless credential migration protocol designed for LANs where identification is replaced by the restrictive conditions in the process of LAN construction. Specifically, MA-TEECM splits the user identification task of the migrating terminal into the following two parts: 1) Ensure that the root of trust of the current device is secure, that is, satisfy the integrity, and grant it a ticket for end-to-end communication; 2) Verify that the terminal contains a root of trust before credential migration. Among them, the construction of the migration group is used to realize the first task, and mutual anonymous authentication is used to achieve

the second task. Furthermore, in practical applications, the user usually needs to shut down the device and reboot into engineering mode to initiate credential migration. Therefore, it is also possible to ensure the consistency of user identities through a specified operation procedure or further using a migration password known only by the current user.

B. ANONYMITY

Anonymity means that the identity information of the device will not be revealed during the authentication process. The TCG specification requires the root of trust to generate different session keys PK based on the public key of the device endorsement key EK, to ensure that the verifier cannot associate a specific root of trust with the session key. To determine whether the session key was generated from the same TA, the attacker needs to determine that T_1, T_2 and T_1', T_2' were generated from the same E. According to the DDH assumption [27], this is impractical. According to the property of direct anonymous authentication, the challenger can only confirm that the verifier is from a valid trusted root, but cannot identify its real identity. Therefore, the MA-TEECM scheme satisfies anonymity.

C. ROBUSTNESS

Robustness refers to the ability of MA-TEECM to defend against various malicious attackers.

- Replay attack. First, assume that the attacker intercepts the PCR message between TA and GM and replays it to obtain group membership, which means that the attacker needs to decrypt and replace $enc_{PK_{gm}}(PCR, TS_1)$, which contradicts the assumption that SGX programs satisfy confidentiality; Similarly, timestamp TS_1 in $enc_{PK_i}(c, \omega_1, \omega_2, T_1, T_2, TS_1)$ ensures that an attacker simply replaying the message can not establish any valid verification. Obviously, the attacker tampering with TS_1 makes c cannot pass the verification of the challenger, and the replay attack will not be effective. Therefore, the MA-TEECM scheme is resistant to replay attacks.
- Collusion attack. Collusion attack is the major challenge for privacy-preserving anonymous authentication. On the one hand, since authentication is bidirectional in MA-TEECM, it means that knowledge-based public key authentication will be performed twice by different initiators to defend against key substitution attacks. On the other hand, a collusion attack requires the TEE to actively share secrets such as EK with the attacker, which goes against our assumption that the integrity-verified TEE can secure keys. Therefore, the MA-TEECM scheme is resistant to collusion attacks.
- Masquerade attack. First, TA not only needs to prove to the challenger that it contains the trusted root in MA-TEECM but also to prove the integrity of its trust root to GM. Obviously, a trusted TEE will not actively masquerade as another TEE to obtain private information. Second, although the protocol process requires the participation of Agent, and then the data transmission between GM, TA_s, and TA_r is encrypted with the public key, an attacker disguised as Agent cannot obtain confidential information from the communication, nor can he help malicious TA establish a connection. Finally, when an attacker can masquerade as a legitimate GM node in the LAN, another necessary condition is that the attacker is able to control the source device to send credential migration requests. However, an attacker who can control the terminal to enter migration mode only needs to build a legitimate GM node instead of creating a fake node in the LAN. Therefore, the MA-TEECM scheme can resist masquerading attacks.

VIII. PROTOCOL VERIFICATION AND RESULTS

Due to the openness of the mobile terminal operating environment, attackers can easily eavesdrop, intercept, modify or even forge the communication process. To simulate the attack behavior in LAN, the AVISPA tool is selected to verify the security of our scheme.

A. AVISPA TOOL

HLPSL [28] is a role-based formal language based on action sequence logic and can express both logical rules and model procedures. Therefore, it is widely used to describe the security properties of the protocol. The basic elements of HLPSL specifications are role, including the basic role and the composed role. Automated Validation of Internet Security Protocols and Applications (AVISPA) is a tool for automated validation of Internet security protocols and applications, which is used to build and analyze the models of security protocols and their robustness [29], [30]. AVISPA uses a modular formal language to describe the security properties of the protocols and implements a tool model for verifying the protocols' efficacy. Users can set the variables such as participant roles, operating environments, implementation goals, and attacker capabilities for security protocols. Furthermore, AVISPA can generate attack trajectories for insecure protocol models. Users can find security vulnerabilities in protocols according to the results of automated analysis to design corresponding security strategies for defense.

B. BASIC ROLE

We have defined a basic role for each participant according to the needs of the MA-TEECM (Table 2). MA-TEECM contains three types of participants according to section V-B, namely trusted migration units (CRS, CRR), proxy units (CRSA, CRRA), and enclave units (GM). The specific role descriptions are as follows:

The trusted migration unit (CRS/CRR) is the final initiator and verifier of the identity authentication process running in the secure world, and is implemented by a trusted application TA entity in the source and target terminals.

The enclave unit GM is the server node running in the SGX Enclave. In MA-TEECM, GM is used to build a group signature system to provide group membership registration

TABLE 2. Basic roles.

Basic role	Definition
role_CRS	CRS,CRSA,CRR,CRRA,GM:agent,PKs,EKs,PKr:public_key,GK:symmetric_key,TS1:text, Nonce_set_CRSA_CRS,Nonce_set_CRS_CRSA:(text) set,Hash_0:hash_func,SND,RCV:channel(dy)
role_CRR	CRR,CRS,CRSA,CRRA,GM:agent,PKr,EKr,PKs:public_key,GK:symmetric_key,TS2:text, Nonce_set_CRRA_CRR,Nonce_set_CRR_CRRA:(text) set,Hash_0:hash_func,SND,RCV:channel(dy)
role_GM	GM,CRS,CRSA,CRR,CRRA:agent,PKs,PKr,PKgm,EKs,EKr:public_key, GK:symmetric_key,TS3:text,SND,RCV:channel(dy)
role_CRSA	CRSA,CRS,CRR,CRRA,GM:agent,PKs,EKs,PKr:public_key,GK:symmetric_key, Nonce_set_CRSA_CRS,Nonce_set_CRS_CRSA:(text) set,Hash_0:hash_func,SND,RCV:channel(dy)
role_CRRA	CRRA,CRS,CRSA,CRR,GM:agent,PKr,EKr,PKs:public_key,GK:symmetric_key, Nonce_set_CRRA_CRR,Nonce_set_CRR_CRRA:(text) set,Hash_0:hash_func,SND,RCV:channel(dy)

goal
authentication_on *GM_CRS_PCRs*
authentication_on *GM_CRR_PCRr*
authentication_on *CRS_CRR_Nsg*
authentication_on *CRR_CRS_Nrg*
secrecy_of *Secret* []
end goal

FIGURE 5. Security goals.

for nodes that pass its verification, and build DAA parameters based on the group membership.

Since the authentication unit TA running in the trusted execution environment cannot directly establish a connection with GM and other TA, a communication proxy role of CRSA and CRRA are set in the normal world of both sides of the migration terminal, and CRSA and CRRA are used as the signature proxy for TA to help it complete authentication and mediate credential transmission.

C. SECURITY GOALS

The security objectives of the MA-TEECM mainly include the realization of mutual authentication between the authentication units CRS and CRR and the secure distribution of keys in different phases of protocol communication. The process of session key distribution is mainly the communication of proxy signature authorization between (CRS, CRSA) and (CRR, CRRA), and CRS, CRR obtains the group member certificate from GM. Mutual authentication is an authentication process between the root of trust in the security domain realized by CRSA/CRRA as the intermediate node of CRS/CRR.

As described in Section V-B, MA-TEECM mainly involves three sub-authentication phases. In Phase 1, CRS verifies the security of CRSA and grants it proxy signing authority. In our formal model, we ignore the security verification process and directly let CRS and CRSA share the signature key pair ($PK_s\prime$, PK_s), which is similar to CRR and CRRA. Therefore, security goals are not defined at this phase; In Phase 2, CRS communicates with GM through CRSA for integrity verification and group membership establishment. Since CRSA is untrusted, and the channel between CRSA and GM is not secure. To prevent attackers from obtaining the proof ticket of CRS to masquerade as a legitimate group member, the PCRs passed between CRS and GM needs to satisfy confidentiality. Therefore, the security objectives defined in this phase are shown in Figure 5, ①, ②; In Phase 3, trusted migration units in different terminals verify that each other contains a root of trust through zero-knowledge proof. Specifically, first, the challenger needs to check the legitimacy of the verifier's group membership. Second, the challenger needs to check whether the verifier contains the private value b of the DAA certificate. This verification process is simulated in our simulation model by a secret value shared with GM by the trusted migration units. Therefore, the security goals defined at this stage are shown in Figure 5, ③ ∼ ⑤;

D. SIMULATION RESULTS

We simulated the process of keys' maintenance and conversion in the authentication process by defining roles, ignoring the specific implementation mechanism of keys' construction and agreement, and focusing on protocol verification on the confidentiality of keys and secrets in the mutual authentication process and the expected security properties. AVISPA's two back-end analysis technologies, On-the-Fly Model-Checker (OFMC) and Constraint-Logic-based A Ttack SEarcher(CL-AtSe), are used to discover errors in protocols and sessions. These two tools are complementary for checking encryption protocols. To determine whether the suggested protocol is resistant to replay attacks and leak attacks, we defined a secrecy target of secret value and four different authenticity targets in the model. In addition, we defined a Delev-Yao intruder with knowledge of the role Agent in the environment role to verify the man-in-the-middle attack.

The security Protocol Animator (SPAN) tool was used to simulate and analyze our protocol to check whether it is secure. The verification results are shown in Figure 6, where the SUMMARY field will display SAFE and UNSAFE according to the detection result. If UNSAFE is displayed, the automatically generated attack path will be displayed in the ATTACK TRACE field. According to the simulation results

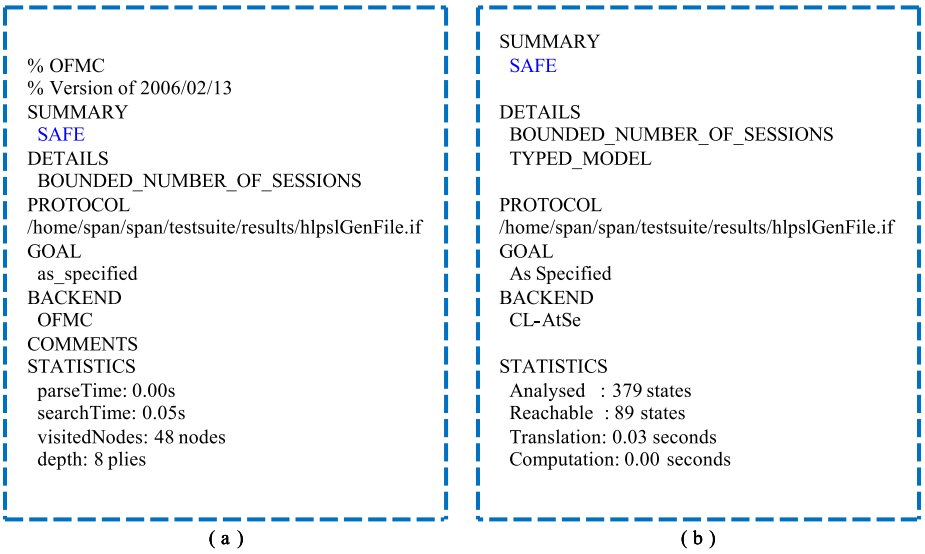

FIGURE 6. AVISPA Output. (a) The result of analysis using OFMC, (b) The result of analysis using CL-AtSe.

in Figure 6 (a), (b), the search time for OFMC to access 48 nodes is 0.05 seconds, and CL-Atse analyzes 379 states with a transition time of 0.03 seconds. The analysis results showed that MA-TEECM meets the security requirements of the migration protocol.

IX. CONCLUSION

Trusted Execution Environment is emerging as a flexible mobile security mechanism that can provide enhanced security guarantees for security-critical applications, credential files, and other types of sensitive data on any mobile device. This paper proposed a model framework that enables peer-to-peer credential migration between personal mobile devices to address credential availability issues caused by device lifecycle events. A third party, insulated from sensitive data, was introduced in the channel establishment process of credential migration, which is responsible for assisting two mobile devices in the local area network to establish group membership. Furthermore, a peer-to-peer credential migration protocol based on the mutual authentication scheme was designed, and the algorithm and model of credential migration in TEE were created. Security analysis showed that MA-TEECM could guarantee the confidentiality and integrity of credential data. Finally, AVISPA's back-end automated verification tools, OFMC and ATSE, were used to verify the security of the proposed protocol successfully.

REFERENCES

[1] A Ltd. *Arm Delivers Record Revenues and Record Profits in FY21*. Accessed: May 13, 2022. [Online]. Available: https://www.arm.com/ja/company/news/2022/05/arm-delivers-record-revenues-and-record-profits-in-fy21

[2] *Arm Achieves Record Revenue and Shipments in Q1 FY 2022*. Accessed: Aug. 9, 2022. [Online]. Available: https://www.arm.com/ja/company/news/2022/08/arm-achieves-record-revenue-and-shipments-in-q1-fy-2022

[3] R. Ballagas, M. Rohs, J. G. Sheridan, and J. Borchers, "BYOD: Bring your own device," in *Proc. Workshop Ubiquitous Display Environ. (Ubicomp)*, vol. 2004, Sep. 2004.

[4] M. Sabt, M. Achemlal, and A. Bouabdallah, "Trusted execution environment: What it is, and what it is not," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, vol. 1, Aug. 2015, pp. 57–64.

[5] T. Hardjono and G. Kazmierczak. (2008). *Overview of the TPM Key Management Standard*. TCG Presentations. [Online]. Available: https://www.trustedcomputinggroup.org/news

[6] L. Karlsson and M. Hell, "Enabling key migration between non-compatible TPM versions," in *Proc. Int. Conf. Trust Trustworthy Comput.* Cham, Switzerland: Springer, 2016, pp. 101–118.

[7] K. Kostiainen, N. Asokan, and A. Afanasyeva, "Towards user-friendly credential transfer on open credential platforms," in *Proc. Int. Conf. Appl. Cryptogr. Netw. Secur.* Berlin, Germany: Springer, 2011, pp. 395–412.

[8] G. Arfaoui, S. Gharout, J.-F. Lalande, and J. Traoré, "Practical and privacy-preserving tee migration," in *Proc. IFIP Int. Conf. Inf. Secur. Theory Pract.* Cham, Switzerland: Springer, 2015, pp. 153–168.

[9] H. Li, Z. Li, Z. Wang, and X. Chang, "Authorization credential migration method, terminal device, and service server," U.S. Patent 16 476 988, Nov. 21, 2019.

[10] N. Kumar, "Identity authentication migration between different authentication systems," U.S. Patent 10 412 077, Sep. 10, 2019.

[11] C. Shepherd, R. N. Akram, and K. Markantonakis, "Remote credential management with mutual attestation for trusted execution environments," in *Proc. IFIP Int. Conf. Inf. Secur. Theory Pract.* Cham, Switzerland: Springer, 2018, pp. 157–173.

[12] T. Liang and S. Min, "TPM2.0 key migration-protocol based on duplication authority," *J. Softw.*, vol. 30, no. 8, pp. 2287–2313, 2019.

[13] M. Song and L. Tan, "A TPM2. 0 key migration protocol and security analysis," *ACTA ELECTONICA SINICA*, vol. 47, no. 7, p. 1449, 2019.

[14] H. Nishimura, Y. Omori, and T. Yamashita, "Secure authentication key sharing between personal mobile devices based on owner identity," *J. Inf. Process.*, vol. 28, pp. 292–301, Apr. 2020.

[15] J. Guerreiro, R. Moura, and J. N. Silva, "TEEnder: SGX enclave migration using HSMs," *Comput. Secur.*, vol. 96, Sep. 2020, Art. no. 101874.

[16] V. A. B. Pop, S. Virtanen, P. Sainio, and A. Niemi, "Secure migration of WebAssembly-based mobile agents between secure enclaves," M.S. thesis, Univ. Turku, 2021.

[17] J. Wang, P. Mahmoody, F. Brasser, P. Jauernig, A.-R. Sadeghi, D. Yu, D. Pan, and Y. Zhang, "VirTEE: A full backward-compatible TEE with native live migration and secure I/O," in *Proc. 59th ACM/IEEE Design Autom. Conf.*, Jul. 2022, pp. 241–246.

[18] J.-Y. Gu, H. Li, Y.-B. Xia, H.-B. Chen, C.-G. Qin, and Z.-Y. He, "Unified enclave abstraction and secure enclave migration on heterogeneous security architectures," *J. Comput. Sci. Technol.*, vol. 37, no. 2, pp. 468–486, Apr. 2022.
[19] W. Geisler, "Reliable migration of WebAssembly trusted applications," Tech. Rep., 2022.
[20] *File: PRD29-GENC-009492C Trustzone Security Whitepaper. pdf MILEDROPEDIA*. Accessed: Aug. 26, 2021. [Online]. Available: http://www.droid-developers.org/wiki/File:Prd29-genc-009492c_trustzone_security_whitepaper.pdf
[21] E. Brickell, J. Camenisch, and L. Chen, "Direct anonymous attestation," in *Proc. 11th ACM Conf. Comput. Commun. Secur. (CCS)*, 2004, pp. 132–145.
[22] A. M. Azab, P. Ning, J. Shah, Q. Chen, R. Bhutkar, G. Ganesh, J. Ma, and W. Shen, "Hypervision across worlds: Real-time kernel protection from the ARM TrustZone secure world," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2014, pp. 90–102.
[23] W. Arthur, D. Challener, and K. Goldman, "Platform configuration registers," in *A Practical Guide to TPM 2.0*. Berkeley, CA, USA: Apress, 2015, pp. 151–161.
[24] A. Tomlinson, "Introduction to the TPM," in *Smart Cards, Tokens, Security and Applications*. Cham, Switzerland: Springer, 2017, pp. 173–191.
[25] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in *Proc. 3rd ACM Conf. Comput. Commun. Secur. (CCS)*, 1996, pp. 48–57.
[26] L. Yang, J. Ma, W. Lou, and Q. Jiang, "A delegation based cross trusted domain direct anonymous attestation scheme," *Comput. Netw.*, vol. 81, pp. 245–257, Apr. 2015.
[27] D. Boneh, "The decision Diffie–Hellman problem," in *Proc. Int. Algorithmic Number Theory Symp.* Berlin, Germany: Springer, 1998, pp. 48–63.
[28] D. Von Oheimb, "The high-level protocol specification language HLPSL developed in the EU project AVISPA," in *Proc. APPSEM Workshop*, Tallinn, Estonia, 2005, pp. 1–17.
[29] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. H. Drielsma, P. C. Heám, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, and L. Vigneron, "The AVISPA tool for the automated validation of internet security protocols and applications," in *Proc. Int. Conf. Comput. Aided Verification*. Springer, 2005, pp. 281–285.
[30] L. Viganò, "Automated security protocol analysis with the AVISPA tool," *Electron. Notes Theor. Comput. Sci.*, vol. 155, pp. 61–86, May 2006.

ZIWANG WANG received the Ph.D. degree in computer science and technology from the Nanjing University of Aeronautics and Astronautics, China, in 2020. He is currently a Lecturer with the Artificial Intelligence and Big Data Department, Hefei University, China. He has published more than ten peer-reviewed journals or conference papers. His research interests include information security and blockchain.

LIANG WANG was born in Zhejiang, China, in 2000. He received the B.S. degree in information management and information systems from Shandong Yingcai University, Jinan, in 2022. He is currently pursuing the M.S. degree in computer science and technology with Hefei University.

HUILI YAN was born in Anhui, China, in 2000. He received the B.S. degree in computer science and technology from Anhui Xinhua University, Heifei, in 2022. He is currently pursuing the M.S. degree in computer science and technology with Hefei University.