An Enhanced Direct Anonymous Attestation Scheme with Mutual Authentication for Network-Connected UAV Communication Systems

Liquan Chen[1*], Sijie Qian[1], Ming Lim[2], Shihui Wang[1]

[1] School of Information Science and Engineering, Southeast University, Nanjing China, 210096
[2] Research Center of Supply Chain and Operations Management, Coventry University, Coventry, CV1 5FB
* The corresponding author, email: Lqchen@seu.edu.cn

Abstract: In network-connected UAV (NC-UAV) communication systems, user authentication is replaced by platform identity authentication and integrity check because many NC-UAVs are operated without human intervention. Direct anonymous attestation (DAA) is an attractive cryptographic scheme that provides an elegant balance between platform authentication and anonymity. However, because of the low-level computing capability and limited transmission bandwidth in UAV, the existing DAA schemes are not suitable for NC-UAV communication systems. In this paper, we propose an enhanced DAA scheme with mutual authentication (MA-DAA scheme), which meets the security requirements of NC-UAV communication systems. The proposed MA-DAA scheme, which is based on asymmetric pairings, bundles the identities of trusted platform module (TPM) and Host to solve the malicious module changing attacks. Credential randomization, batch proof and verification, and mutual authentication are realized in the MA-DAA scheme. The computational workload in TPM and Host is reduced in order to meet the low computation and resource requirements in TPM and Host. The entire scheme and protocols are presented, and the security and efficiency of the proposed MA-DAA scheme are proved and analyzed. Our experiment results also confirm the high efficiency of the proposed scheme.

Keywords: network-connected UAV; trusted computing; direct anonymous attestation; mutual authentication; trusted platform module

I. Introduction

In recent years, unmanned aerial vehicles (UAVs) have experienced rapid growth in civil and commercial areas such as precise agriculture, sky monitoring, cargo delivery, traffic control, rescue and search [1]. However, existing UAV systems mainly rely on the traditional direct ground-to-UAV communications over the unlicensed spectrum (e.g., ISM 2.4GHz), which is of limited data rate, unreliable, insecure, and can only operate within the visual line of sight (LoS) range [2]. As the number of UAVs and their related applications increase explosively in the coming years, it is imperative to develop innovative wireless communication technologies not only for supporting the ultra-reliable UAV remote command and

Received: Feb. 14, 2018
Editor: Shi Jin

control to ensure their safe operations, but also for enabling high-capacity mission-related information transmission.

Integrating UAVs into cellular and satellite networks is a promising solution to achieve the above goals, where UAVs with their own missions could be connected with either cellular base stations (BSs) or satellites as special aerial users, referred to as network-connected UAVs (NC-UAVs) [3-4]. Thanks to the almost ubiquitous accessibility worldwide, network-connected UAVs are expected to significantly outperform the traditional direct ground-to-UAV communications.

Based on the cellular and satellite networks, the controlling and accessing to NC-UAVs have been spread all over the world. The security of the NC-UAVs is faced new challenges. Although the security and authentication mechanisms of cellular and satellite networks have ensured the security in the transmission link, the confidentiality, integrity and availability of NC-UAVs are needed to be improved based on high-level security mechanisms.

Some researchers have proposed related security solutions for UAV communication systems. The communication security of UAV is analyzed in [5], while the vulnerability of UAVs to cyber attacks is presented in [6]. GPS spoofing is used to capture and control the UAV [7] and a detection approach for GPS spoofing attacks to UAV is proposed in [8]. Security testing of a UAV is presented in [9]. All the above security analysis and testing are based on the direct ground-to-UAV communication systems. For NC-UAV communication systems, [10] presented a security authentication system using an encrypted channel on UAV network. Some technologies are proposed in [11] to improve the communication security of open source UAVs, while [12] presented a handover key management scheme in an LTE-based UAV system. However, the schemes in [10-12] do not effectively solve the security problems in the network-connected UAV communication systems.

In a decentralized communication system, the enforcement principle of access control is supplemented by trust. An entity can be trusted if it predictably and observably behaves in a manner expected for its intended purpose. Trusted computing has been proposed by the trusted computing group (TCG) and added to the ISO/IEC 11889 standard [13]. Although TCG proposed some implementing methods in [13] with regard to embedded application environments, they are not sufficient for solving all the problems that emerge in embedded application systems, especially in the embedded NC-UAV systems. When trusted computing architecture is applied to NC-UAV communication systems, the computational capability, the security and efficiency requirements of such systems need to be concerned.

With the trusted computing protocol, there are three stages for one node to accomplish trusted network access: user authentication, platform trusted authentication, and trusted network connection (TNC). A platform can be trusted if it predictably and observably behaves in a manner expected for its intended purpose. TPM is the trusted base in a trusted platform. We can use a platform authentication protocol to determine whether one node has the trusted TPM module and all the applications in this node run as expected.

Because there is no human intervention in NC-UAV communication systems, user authentication is mainly replaced by platform authentication, whereas the following factors should be concerned. 1) Platform identity authentication and integrity checking of NC-UAV are critical for such vehicle to be allowed to access in the ground control station (GCS). 2) NC-UAV is always installed outdoors in lack of security, and thus it can easily be destroyed. Authentic trusted platform modules (TPMs) could even be stolen and used in adversary NC-UAV to cheat the authentication server. 3) We know that the computational capabilities of TPMs and Hosts are of low level because they are mainly based on embedded computing hardware. 4) Mutual authentication between NC-UAV and the authentication center is necessary for network-connected UAV

systems.

The solution first developed by TCG uses a trusted third party, privacy certification authority (Privacy CA), to realize platform authentication [14]. In the Privacy CA scheme, each TPM generates a key pair called the endorsement key (EK). Privacy CA is assumed to know EK of all valid TPMs and issues a certificate for a TPM. TPM can then forward this certificate to the Verifier and authenticate itself with regard to this AIK. This solution has the obvious limitation that the Privacy CA server needs to be involved in every transaction. Moreover, if the Privacy CA and the Verifier collude, the Verifier can uniquely identify a TPM.

A new type of scheme, direct anonymous attestation (DAA), was thus developed by Brickell et al.[15] for remote authentication of a trusted computing platform while preserving the privacy of the platform. DAA is a new group signature scheme without the capability of opening a signature, but with the mechanism to detect a rogue member. In the DAA scheme, a suitable signature scheme is employed to issue certificates on a membership public key generated by a TPM. Then, to authenticate as a group member, TPM proves to the Verifier that it possesses a certificate on a public key for which it also knows the secret key. Many researchers have proposed different types of DAA schemes to meet the requirements in different applications and environments.

The DAA scheme that holds anonymous and privacy properties has effectively resolved the limitations of the Privacy CA scheme, and it has the following characteristics: 1) Efficiency. When the platform receives the DAA credential from the Issuer, it can use this credential to conduct the signing and verification processing many times. 2) Anonymity. Because the DAA scheme applies the zero-knowledge proof theory to prove the trust of a new platform that possesses legitimate credentials, it can prevent adversaries from seeking the identity of the real communicating TPM. It is difficult for an adversary to track the identity of the target TPM even when the Verifier can collude with the credential Issuer [16-17]. 3) Privacy. The Issuer is the trusted credential issuer that has the EK lists to determine the legitimacy of the applying TPM, and the Verifier employs the Camenisch-Lysyanskaya (C-L) signature scheme [18] and respective discrete logarithm-based proofs to prove the possession of a certificate, whereas privacy and anonymity are guaranteed under the decisional Diffie-Hellman (DDH) assumption.

The DAA scheme developed by Brickell et al. [15] is based on a strong RSA assumption and it is called the RSA-type DAA, (hereafter referred to as RSA-DAA). Theory analysis results have shown that the protocols and algorithms in RSA-DAA scheme are complicated and inefficient, it is not suitable for TPMs with fewer computational capabilities. In recent years, researchers have worked on how to create DAA schemes with elliptic curves cryptography (ECC) and pairings. We call these, the ECC-DAA schemes. In general, the ECC-DAA scheme is more efficient in both computation and communication than the RSA-DAA scheme. TPM's operation is much simpler and the key/signature length is much shorter in the ECC-DAA scheme than the RSA-DAA scheme.

Many ECC-DAA schemes are based on the q-strong Diffie-Hellman (q-SDH), DDH, and computing Diffie-Hellman (CDH) difficult assumptions [16-17] [19-21]. They have reduced the computational workload in TPM and enhanced security among the Signer, Issuer, and Verifier. However, to the best of our knowledge, there is no exact DAA scheme that has been proposed to meet the requirements of NC-UAV communication systems. Based on the security requirements of less computational workload in TPM and the Host, mutual authentication requirement between the Signer and Verifier, and bundling rogue check requirement of TPM and the Host in NC-UAV communication systems, we propose an enhanced DAA with mutual authentication that can satisfy all these requirements.

Our main contributions include 1) We present a security prototype for NC-UAV communication systems firstly. The security flaws and risks in NC-UAV platform authentication are analyzed, and a new trusted NC-UAV remote DAA scheme is proposed to meet the security requirements in UAV communication systems. 2) A new mutual authentication-direct anonymous attestation (MA-DAA) scheme with less computation and mutual authentication for NC-UAV communication systems is proposed. The significant advantages of the new scheme are described as follows:

a) We put off the J, K pairs and the computations in the Sign/Verify stage in [17], while the computation of variable D is transferred from TPM to the Issuer. Moreover, an efficient batch proof and verification scheme is used to reduce not only the computational workload of TPM, but also that of the Host. This is critical for NC-UAV systems because the Hosts in such systems are also mainly based on embedded hardware without a high-level of computational capabilities.

b) The identity of TPM and the Host are authenticated in bundle and rogue-checked by the Issuers and Verifiers. This technique can avoid the security flaw where one NC-UAV uses a trusted TPM that is stolen from another trusted NC-UAV in order to pass verification and launch malicious attacks on the GCS center. This compromise problem is not concerned in the existing DAA schemes.

c) We propose a new variable, $c_2 = H_2\left(f \| bsn\right)$, and related algorithms to replace the J, K variables from [17] and perform rogue list checking and user-controlled linkability. The Verifier can check the received c_2 against the rogue list to find the rogue TPMs. Meanwhile, with the same bsn value, the Verifier can find the messages that originate from the same TPM by obtaining the same c_2 value. This scheme can reduce one multiplication computation induced by the J, K pair algorithm.

d) We add mutual authentication to the DAA scheme. NC-UAV can also verify the trust of the Verifier with the help of the MAC mechanism and secret key between these entities. In NC-UAV communication systems, data are mostly transmitted from NC-UAV to GCS. In the existing DAA schemes, when there is an adversary that combines the function of the Verifier and GCS, NC-UAV's privacy is compromised, and obtaining critical data from NC-UAV is facilitated. Such security vulnerability and flaw are dangerous for critical NC-UAV data.

e) In addition to enhancements in security, the MA-DAA scheme also reduces the computation workload of TPM and the Host at the Join, Sign, and Verify stage, and the communication payload. This is important for the NC-UAV mostly powered by battery.

The rest of the paper is organized as follows: Section 2 presents the network-connected UAV communication system and constructs a remote DAA scheme for NC-UAV to realize secure and efficient access. Then, an MA-DAA scheme is proposed in Section 3 that meets the security requirements in NC-UAV communication systems. In Section 4, performance analysis of the proposed MA-DAA scheme is presented. Experiment results also confirm the analysis conclusions. Finally, we conclude the paper in Section 5.

II. NC-UAV Communication Systems and Security

UAV technology has emerged as a cutting-edge technology for next-generation. Various UAV applications have already started to emerge in various fields. Based on the cellular and satellite network connection facility, the number of network-connected UAV units will become huge, and many new characters are accompanied with the emerging of NC-UAV communications.

In a normal NC-UAV communication system, NC-UAV, GCS, and transmission link are the three major components. Because NC-UAV is mainly built on embedded hardware and run with embedded operating systems, its limited computational capability is the largest

constraint for NC-UAV to conduct sophisticated computation and processing. Huge and intermittent data are transmitted downlink from NC-UAV to GCS. GCS is located on the core network side that collects and stores UAV data, and the authenticated NC-UAVs can connect to GCS and get the command and control.

Because network-connected UAV units are widely spread in different environments and there are vulnerabilities in wireless access links, NC-UAV communication systems encounter the following security challenges. First, because NC-UAV units are unattended in most of the time, adversaries can damage them easily. Second, NC-UAV communication is based on wireless links, so eavesdropping in-the-middle is extremely simple to apply. Finally, because most NC-UAV units are characterized by low capabilities of power supply and computing resources, novel privacy and authentication algorithms cannot be easily implemented in such units. [9-12] have described different types of attacks that exist in NC-UAV communication systems.

In order to establish a trusting relationship in dispersed systems, we must collect security-relevant elements and capabilities to form a trust boundary. These include methods that extend the trust boundary and convey trust to an external entity. A TPM or trusted environment (TRE) provides a hardware security root of trust, allowing the system construction to combine the characteristics of trust and enforcement. TPM or TRE performs a secure start-up process that ensures the TPM or TRE reaches a determined trusted state. Then this trusted state can be transferred from TPM or TRE to the Host. The benefits of trust in NC-UAV systems include: 1) A trusted running environment is provided against attacks on NC-UAV units. 2) The privacy of the data transmitted between the NC-UAV units and GCSs is ensured. 3) We can realize user authentication, platform authentication, and platform integrity verification based on the trusted condition in the NC-UAV units to construct a trusted network access architecture.

For NC-UAV communication systems, platform authentication is more important than user authentication. Thus, in this paper, we focus on platform authentication research and propose a new DAA scheme to authenticate the trust of NC-UAV platforms securely and efficiently. In the DAA scheme, a TPM in NC-UAV has to obtain a legitimate DAA credential from a trusted Issuer before processing the trusted platform verification with the Verifier. When NC-UAV obtains the legitimate DAA credential, it signs the DAA credential with its secret key. Finally, the Verifier validates the trust of the signed credential from NC-UAV. In general, NC-UAV units obtain public secure parameters Par_{pk} and EKs from the NC-UAV developers or suppliers in a secure environment or secure channel. Then they are delivered to the operators and deployed in the designated working fields. The installed NC-UAV units connect and transmit data between them and the GCS. However, before connecting to the GCS, NC-UAV units have to perform remote anonymous attestation processing with platform validation authority (PVA) entities and the access control function model in the operator. Here, PVA entities are the verification system used to verify the trust of NC-UAV units. A practical DAA prototype for NC-UAV communication systems is presented in Figure 1.

In this DAA prototype, prior to the start of trusted proof processing, TPM in NC-UAV first needs to create secret key f. Once the

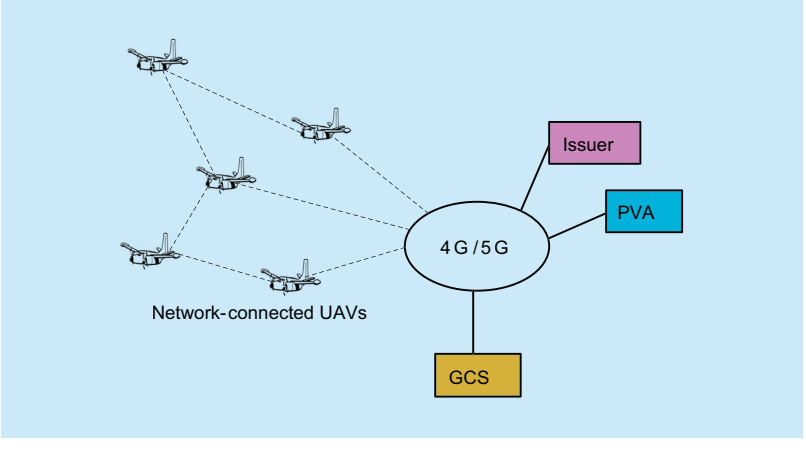

Fig. 1. *DAA prototype for NC-UAV communication systems.*

DAA credential is received, the TPM platform signs message m or the Hash value of the NC-UAV platform integrity information with secret key f. In general, these processes can realize trusted proof of the platform and integrity check. Finally, this DAA signature is sent to the PVA verification server for PVA to verify and conduct rogue-checking of the received DAA signature.

In the existing DAA schemes, the DAA Sign protocol between NC-UAV and the Issuer/Verifier is a single-side authentication protocol. There is no authentication program in NC-UAV to authenticate the identity and authority of the Issuer and Verifier. This is a security flaw that can threaten the privacy of NC-UAV units. For example, when one adversary constructs a device that contains the function of Issuer and Verifier, such device can easily pass the NC-UAV units' DAA attestation processing, and then receive the critical data transmitted from those NC-UAV units. This is very dangerous for NC-UAV units used for critical tasks. Therefore, a new DAA scheme with mutual authentication should be proposed to solve this security problem, and the other vulnerabilities and inefficiency problems of the existing DAA schemes.

III. THE PROPOSED MA-DAA SCHEME

3.1 Preliminary knowledge

We present the preliminary knowledge on the pairings and elliptic curve cryptography. Here, G_1, G_2 and G_T are the cyclic groups of prime order p, g_1 is the generator of G_1, g_2 is the generator of G_2. Computing the discrete logarithm on group G_1, G_2, and G_T is difficult.

If the mapping $e: G_1 \times G_2 \to G_T$ can meet the following constraint: 1) $g_1 \in G_1, g_2 \in G_2, 1 \in G_T$, and $e(g_1, g_2) \neq 1$, 2) $x \in G_1, y \in G_2$, then $e(x, y)$ can be computed in polynomial time, 3) for $x \in G_1, y \in G_2$ and $a, b \in Z_p$, $e(x, y)^{ab} = e(x^a, y^b)$, we call $e: G_1 \times G_2 \to G_T$ is a bilinear mapping.

In the pairings $\hat{t}: G_1 \times G_2 \to G_T$, G_1 and G_2 are the additive groups and G_T is the multiplicative group whose prime order is p. However, when we want to evaluate the computational cost of different DAA scheme with respect to each player, G_1 is used to donate the cost of exponentiation computation in group G_1, G_2 donates the cost of exponentiation computation in group G_2, and G_T donates the cost of exponentiation computation in group G_T. Meanwhile, G_x^m donates the operation costs of multi-exponentiation with m times in group G_x (x is 1, 2, or T). We also let P denote the cost of a pairing computation, and P^m is the multiple operation costs of m pairings.

According to [16], when asymmetric bilinear pairings and the Barreto-Naehrig curves are applied, and the bit sizes of G_1 and G_T are the same, the computational cost of one G_1 is approximately $1/14$ of the computational cost of one G_T. This is based on the fact that G_1 is an elliptic curve over F_q and G_T is a subgroup of $F_{q_{12}}$ which requires approximately $10F_q$ operations. For the sake of TPM efficiency, it is important to reduce the G_T group computations in TPM and the Host, or replace them with the G_1 group computations when we want to improve the performance of the DAA schemes in network-connected UAV systems.

3.2 Proposed MA-DAA scheme

A DAA scheme includes three partners: Issuer, Signer, and Verifier. The Issuer is responsible for verifying the legitimacy of the Signers and issues a DAA credential to each Signer. A Signer, which is a pair of Host and associated TPM, can prove membership to a Verifier by providing a DAA signature. The Verifier can verify the membership credential from the signature, but it cannot learn the identity of the Signer.

In network-connected UAV communication systems, NC-UAV adopts the detection and alarm mechanism for vandalism resistance in order to protect NC-UAV from attack. In this

paper, we adopt r' instead of the (J, K) pair to detect connectivity such that the computation load of TPM Sign is reduced from $1G_1/2G_1$ [20] to $1G_1$. We change the way of DAA certificate generation, and transfer $D = f \cdot B$ computed in TPM to the Issuer. The Issuer can then determine value D using input value F based on the B-bLRSW assumption. The computation cost in TPM is no longer necessary, and thus the computation amount of TPM Join is reduced from $3G_1$ [20] to $2G_1$. Therefore, not only the performance of TPM Sign, but also that of TPM Join are improved in the proposed MA-DAA scheme.

The overall MA-DAA scheme includes the Setup protocol that established the system parameters, the Join protocol that obtains the certificate, and the Sign/Verify protocol.

Setup protocol

Assuming that: G_1, G_2, G_T are cyclic groups with order of prime q, bilinear pairs $\hat{t}: G_1 \times G_2 \mapsto G_T$, $H_1: \{0,1\}^* \mapsto Z_q$, we obtain the parameter set par_C as $\left(G_1, G_2, G_T, \hat{t}, P_1, P_2, q, H_1\right)$.

In setup protocol, we define the Issuer parameter set as par_I, which includes Issuer public key and private key as ipk and isk, isk: $x, y \leftarrow Z_q$, $ipk: (X, Y)$, $X = xP_2 \in G_2$, and $Y = yP_2 \in G_2$. In addition, the Issuer generates a pair of keys (SK_I, PK_I) for mutual authentication. Moreover, the Issuer provides a unique value K_I to generate secret value f. Then the parameter set par_I is (ipk, K_I, PK_I).

Suppose that the parameter par_R for TPM is H_2, $H_2: \{0,1\}^* \mapsto Z_q$. The public and private key pair for TPM is (SK_T, PK_T), par_T: PK_T. The public and private key pair of the Host is (SK_H, PK_H), par_H: PK_H. The public and private key pair of the Verifier is (SK_V, PK_V), par_V: PK_V. And the parameter for Sign/Verify is par_S: $H_3: \{0,1\}^* \mapsto Z_q$, $H_4: \{0,1\}^* \mapsto Z_q$.

After Setup protocol, the system public parameter set par is constructed as: $(par_C, par_I, par_R, par_S, par_T, par_H, par_V)$.

Join protocol

The Join protocol is realized based on the request/response interaction between the Issuer and TPM/Host. We divide the Join protocol into four parts in this order: Issuer request, TPM response, Issuer response, and Host verify. The overall process framework of the Join protocol is shown in Figure 2.

The operation for the Issuer request is shown in Figure 3. First, the Issuer needs to confirm that it is a legitimate trusted platform that issues DAA certificates, whereas TPM needs to check the legitimacy of the Issuer. That is to say, an authentication channel should be established between the Issuer and TPM in advance. The establishment is completed with random numbers n_{I_1} and n_{I_2} chosen by the Issuer, which encrypts n_{I_2} with SK_I and then encrypts $n_{I_1} \| n_{I_2} \| E_{SK_I}(n_{I_2})$ with PK_T for the Host. Similarly to the Issuer, the Host generates a random number $RAND$,

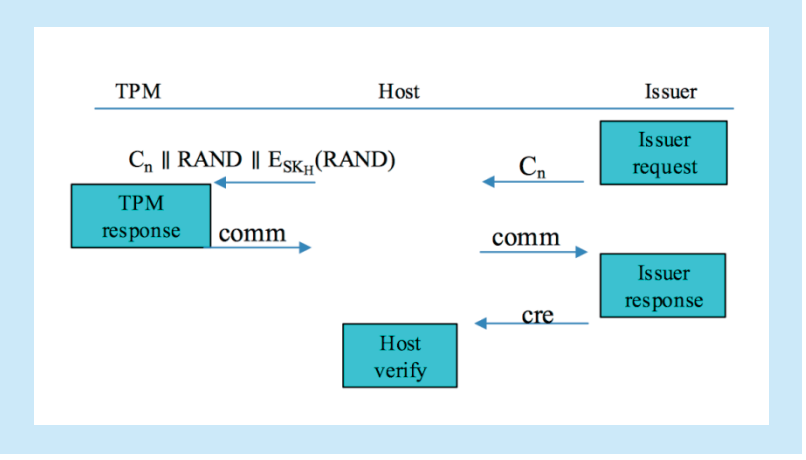

Fig. 2. *Overall process framework for Join protocol.*

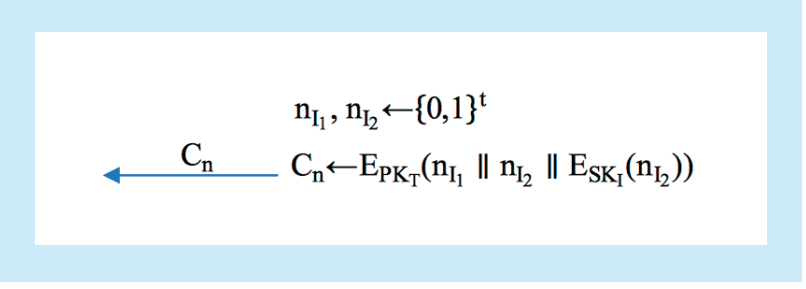

Fig. 3. *Issuer request operation.*

encrypts $RAND$ with pre-shared private key SK_H, and sends $C_n \| RAND \| E_{SK_H}(RAND)$ to TPM. TPM decrypts $E_{SK_H}(RAND)$ with pre-shared public key PK_H to obtain $RAND'$. The result of the comparison between $RAND'$ and $RAND$ indicates whether the Host is legitimate. Only when $RAND'$ is equal to $RAND$, does the TPM continue the protocol. Next, if TPM can successfully decrypt C_n with SK_T, we can obtain the value of $n_{I_1} \| n_{I_2} \| E_{SK_I}(n_{I_2})$, and return n_{I_1} to the Issuer, which indicates that TPM owns its legitimate EK private key. In addition, TPM decrypts $E_{SK_I}(n_{I_2})$ with the legitimate EK public key of the Issuer and compares the decrypted result n_{I_2}' with n_{I_2}. If n_{I_2}' is equal to n_{I_2}, this indicates the legitimacy of the Issuer. Therefore, mutual authentication between the Issuer and TPM is completed. The operation process for the TPM response is shown in Figure 4.

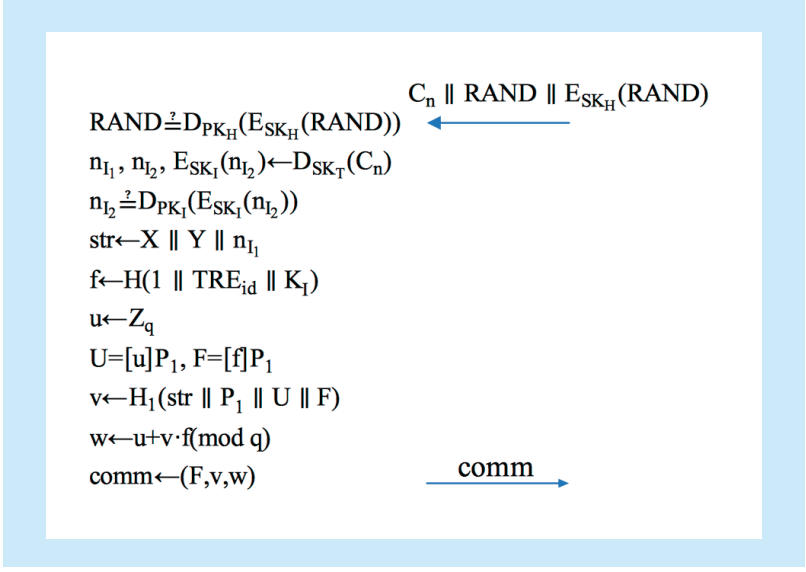

Fig. 4. *TPM response operation.*

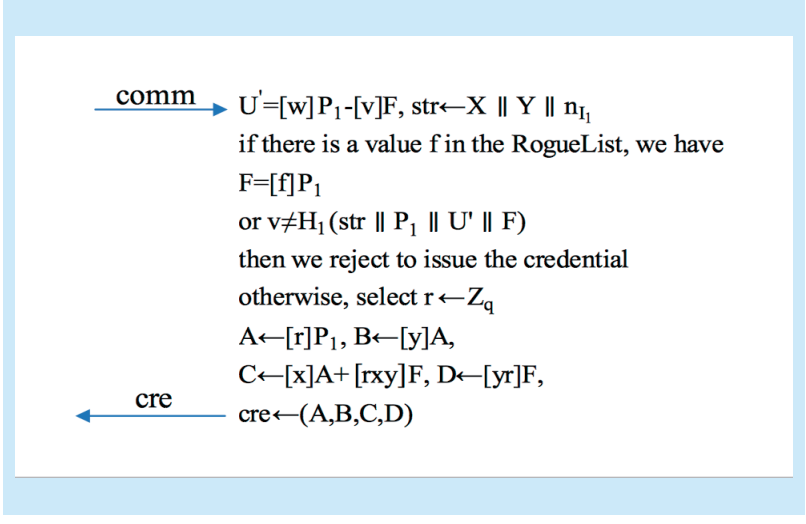

Fig. 5. *Issuer response operation.*

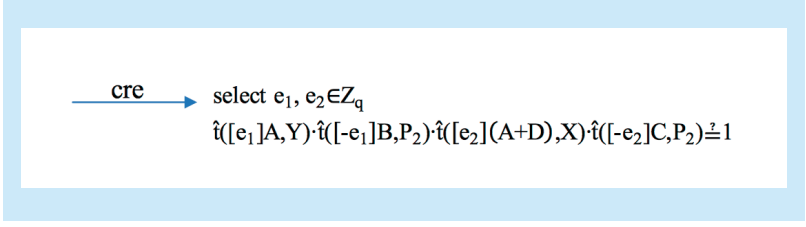

Fig. 6. *Host verify operation.*

TPM generates secret value f with K_I and TRE_{id} (the internal ID of TRE in NC-UAV), where $f = H(1 \| TRE_{id} \| K_I)$. TPM generates $comm$ from f, and sends it to the Issuer. The operation of the Issuer response is shown in Figure 5.

The Issuer authenticates the $comm$ value. That is to say, it examines the zero-knowledge proof result of TPM in order to check whether TPM owns a legitimate f. If the authentication is successful, the Issuer generates a DAA certificate with F in $comm$. It is important to note that in the proposed scheme, the computation of D in certificate cre is calculated as $D=[yr]F$, rather than the computation of $D=[f]B$ using f as is done in [10]. This is mainly because F is generated from value f. Finally, we generate DAA certification (A,B,C,D) based on the blind-bilinear B-bLRSW assumption. To summarize, the computation amount of TPM Join in the MA-DAA scheme is reduced to $2G_1$, which is caused by the change of computation of D value. The computation amount is the lower than those existing DAA schemes based on the LRSW and DDH assumptions.

The operation for the Host verify is shown in Figure 6. After receiving certification cre, the Host verifies the correctness of cre. By adopting the batch authentication technology,

the Host determines whether the certificate in the signature is correct using a P^4 computation, i.e., the Host handles four bilinear pairings in one process. The P^4 computation cost is less than four independent bilinear pair computation cost ($4P$) [22].

Sign/Verify protocol

The Sign/Verify protocol refers to the process that the TPM, along with the Host, performs the knowledge sign of message msg, generates DAA signature σ. , and then delivers σ to the Verifier. According to various operations that occur in chronological order in the Sign/Verify protocol, it can be divided into three parts: Host Sign, TPM Sign, and Verify. The overall process of the Sign/Verify protocol is shown in Figure 7.

The Verifier generates a random number $RAND_V$ and encrypts it with its private key SK_V, and then sends $n_v \| bsn \| RAND_V \| E_{SK_V}(RAND_V)$ to the Host. bsn is a base name selected by the Verifier and n_v is also generated by the Verifier.

The operation of the Host Sign is shown in Figure 8. The Host performs a blind computation of the DAA certification value (A,B,C,D) after receiving n_v, base name bsn, $RAND_V$, and $E_{SK_V}(RAND_V)$ from the Verifier in order to generate (R,S,T,E). Meanwhile, the Host generates a random number $RAND$ and encrypts it with pre-shared private key SK_H, which is similar to the procedure in the Join protocol. Next, the Host sends $S \| bsn \| c_1 \| RAND \| E_{SK_H}(RAND) \| RAND_V \| E_{SK_V}(RAND_V)$ to TPM.

The operation of TPM Sign is shown in Figure 9. After receiving message from the Host, TPM verifies the legitimacy of the Verifier and Host using the same method as the Join protocol. TPM calculates $RAND_V$' and $RAND$' similarly to decrypting $E_{SK_V}(RAND_V)$ and $E_{SK_H}(RAND)$ with pre-shared keys PK_V and PK_H, respectively, and compares $RAND_V$' with $RAND_V$ and $RAND$' with $RAND$. Only when these two pairs of values are respectively equal, are the Verifier and Host proven to be legitimate.

TPM continues to finish the remaining computation of the signature value after checking the legitimacy of the Host. TPM generates independent value c_2 for relevance detection.

$$c_2 = H_2(f \| bsn) \tag{1}$$

Considering c_2 in equation (1) as a public signature member value, we perform the zero-knowledge proof for the possession of

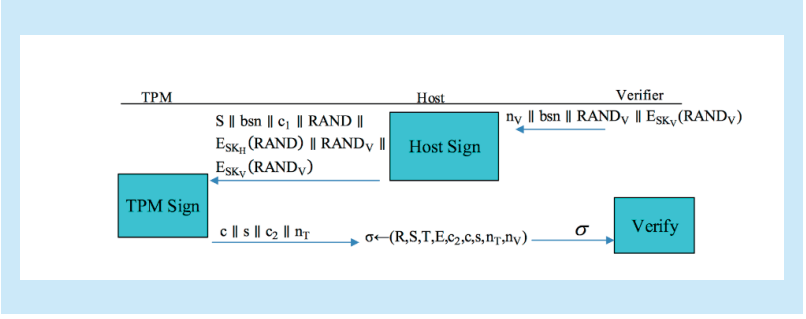

Fig. 7. *Overall process of Sign/Verify protocol.*

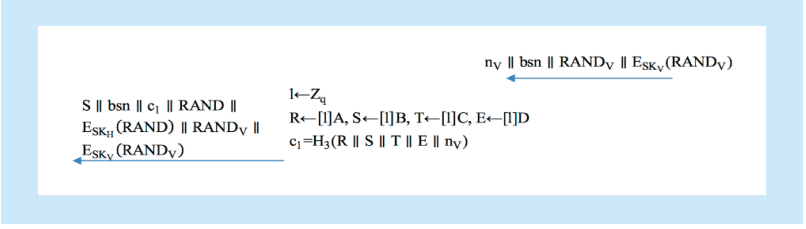

Fig. 8. *Operation of Host sign.*

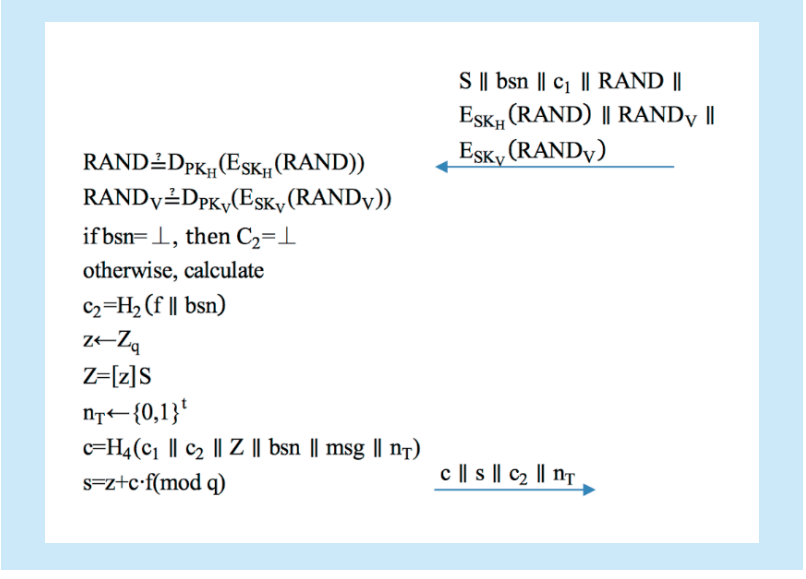

Fig. 9. *Operation of TPM sign.*

a legitimate DAA certification and generate c and s, whereas c_2 is added to the Hash computation of c. TPM sends c, s, c_2, and random number n_T together to the Host, and the final signature σ generated by the Host is $(R,S,T,E,c,s,n_2,n_T,n_v)$.

The operation of Verify is shown in Figure 10. The Verifier performs the verify operation after receiving signature σ. First, the Verifier adds counterfeit f on the rogue list to the blind certificate as $E=[f]S$ to determine whether the f value used in the signature has been disclosed, and verifies whether the blind signature value (R,S,T,E) is correct. Then, the Verifier determines whether the zero-knowledge proof of possessing a legitimate DAA certification in signature σ is correct. If it is correct, it indicates that the Signer owns a legitimate secret value f and legitimate DAA certificate based on the same f. If the Verifier has provided a specific *bsn* in advance, the relevance detection of the signature is also required. Relevance detection can be performed using signature member value c_2 generated from secret value f and base name *bsn* from the Verifier. Such detection plays a role similar to the *J*, *K* mechanism. The entire Verify process is successful only if all these verification steps are correctly completed.

3.3 Security analysis of MA-DAA scheme

Firstly, we find that the security risks in [16] have been solved in the MA-DAA scheme. c_2 is kept within TPM before generating signature values s and c and not exposed to the Host. Therefore, there is no possibility for the attacker to obtain the c_2 value even if the Host is captured. At the time of c_2 generation and delivery of signature values s and c to the Host, the Host cannot change the c_2 value anymore. Because c_2 has been added to the Hash computation of the c value, any change from the Host is discovered by the Verifier.

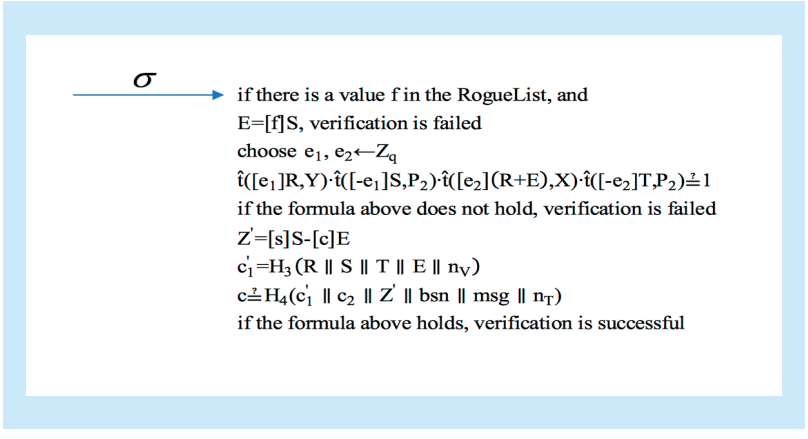

Fig. 10. *Operation of Verify processing.*

At the same time, TPM adds *bsn* to Hash verification in order to keep the Host from changing *bsn* maliciously and harm the relevance of the signature. As a result, the situation where the attacker might control the Host to undermine the relevance does not occur in the MA-DAA scheme. However, once TPM is captured, it is easy for TPM to damage the signature's relevance. Physical protection measures are adopted for NC-UAV in order to protect the embedded TPM from destroyed. Compared with external NC-UAVs, the embedded TPM cannot easily be attacked. It is appropriate for the Verifier to trust the credibility of TPM and believe it can compute the value of c_2 correctly, instead of casually guessing a value to be c_2. For this reason, c_2 can play a key role in performing relevance detection that is as good as *J*, *K* in existing schemes [19] [20]. The use of c_2 is sufficiently secure.

Secondly, we analyze the correctness and security of the overall processes in the MA-DAA scheme. Athe time of verifying, the equation

$$\hat{t}([e_1]R,Y)\cdot\hat{t}([-e_1]S,P_2)\cdot\hat{t}([e_2](R+E),X)\cdot\hat{t}([-e_2]T,P_2)=1 \quad (2)$$

guarantees the correctness of the MA-DAA scheme. If equation (2) is true, $\hat{t}(R,Y)=\hat{t}(R,yP_2)=\hat{t}(yR,P_2)=\hat{t}(S,P_2)$ and $\hat{t}(R+fS,X)=\hat{t}(R+fS,xP_2)=\hat{t}(x(R+fS),P_2)=\hat{t}(T,P_2)$ must be true. This indicates that

the DAA certificate of the signature is generated correctly. The security of the MA-DAA scheme is mainly defined as: provided that secret value f and the DAA certification have not been disclosed, an attacker cannot succeed.

The security analysis is presented as follows. In the circumstance when no secret value for TPM or the DAA certification is disclosed, without the Verifier, an attacker should provide (R,S,T,E) alone to make equation (2) be true, which expects $S=[y]R$, $T=[x](R+E)$ to be true. Assume that attacker $\mathcal{A}$ selects $R=[\alpha]P_1$, $S=[\beta]P_1$, $T=[\gamma]P_1$, and $E=[\delta]P_1$, and the public key of the Issuer is known as $X=[x]P_2$, $Y=[y]P_1$. $S=[y\cdot\alpha]P_1$ is required for making $S=[y]R$ true, which means that given $[\alpha]P_1$ and $[y]P_1$, the attacker must be able to compute $[y\cdot\alpha]P_1$ in order to solve the CDH problem in G_1. This is obviously impossible for the CDH problem in group G_1 of a non-symmetric bilinear pair because it is generally considered as terribly difficult to be resolved in cryptography.

In addition, unlike the existing DAA schemes, the MA-DAA scheme has the mutual authentication feature. Assume that, before any system setup, each Issuer has its private endorsement private key SK_I, and each TPM gets the corresponding public key PK_I. The Issuer generates a random number n_{I_2} and encrypts it with SK_I. TPM admits the legitimacy of the Issuer if the result of decrypting $E_{SK_I}(n_{I_2})$ with PK_I is equal to the received n_{I_2}. Considering that the Issuer has checked TPM in the Join protocol, mutual authentication is completed.

In order to prevent a corrupted Host from taking an honest TPM to sign an illegal message, it is necessary to bind TPM and the Host when manufacturing the devices. A pair of pre-shared public/private key PK_H / SK_H is embedded into TPM and the Host. During the Join and Sign protocols, the Host needs to generate a random number RAND and send $RAND \| E_{SK_H}(RAND)$ to TPM. TPM checks the consistency between $D_{PK_H}(E_{SK_H}(RAND))$. and $RAND$ in order to verify the Host's legitimacy.

Based on the above analysis results, it can find out that the MA-DAA scheme is sufficiently secure on the premise that TPM in NC-UAV is secure and credible. Compared with the existing DAA schemes based on the LRSW and DDH assumptions, there is no security weakness, but an efficiency improvement in the proposed MA-DAA scheme. MA-DAA scheme also has the highest running efficiency among all the DAA schemes based on the LRSW and DDH assumptions. Both the Join and Sign protocols in the MA-DAA scheme have been improved, the computation quantity of TPM Join has been reduced to $2G_1$, and that of TPM Sign is as low as $1G_1$. The benefits of security, low cost, high efficiency make the MA-DAA scheme satisfy the dual requirements of security and efficiency for NC-UAV systems. The following section provides a detailed analysis and experimental verification of the efficiency enhancement.

3.4 Performance analysis of the MA-DAA scheme

We compare the MA-DAA scheme with the existing DAA schemes in terms of the computation cost of all entities during the procedures of the Join and Sign/Verify protocols. A comparison of the computation cost in the Join protocol operation stage is listed in table 1.

Table I. *Computation cost comparison among MA-DAA and existing schemes in Join protocol.*

Stage	Schemes	TPM	Host	Issuer
Join	BCC-DAA	$2G_N^3+3G_T$	$1G_N^2+1G_\Gamma+1P_v$	$nG_\Gamma+2G_N+1G_N^4+1G_\Gamma^2+1P_c$
	BCL-DAA	$3G_1$	$6P$	$2G_1+2G_1^2$
	scheme in [17]	$3G_1$	$4P$	$2G_1+2G_1^2$
	scheme in [23]	$3G_1$	$6P$	$2G_1+1G_1^2$
	scheme in [24]	$2G_1$	$4P$	$1G_1+2G_1^2$
	scheme in [20]	$3G_1$	$1P^4$	$2G_1+2G_1^2$
	MA-DAA	$2G_1$	$1P^4$	$3G_1+1G_1^2$

We can find that the TPM Join computation cost of the MA-DAA scheme is $2G_1$, which is the lowest among all the DAA schemes based on the LRSW and DDH assumptions, and the computation cost of the Host is only $1P^4$. However, at the same time, we can find that TPM offloads $1G_1$ to the Issuer, and the computation cost of the Issuer is increased from $2G_1+2G_1^2$ [20] to $3G_1+1G_1^2$. Considering the fact that the role of the Issuer is played by a server with powerful computing ability, offloading the computation cost can greatly improve the performance of the entire Join protocol.

The comparison of the computation cost in the Sign/Verify protocol operations of the MA-DAA and the existing DAA schemes is listed in table 2.

We can find from table 2 that in scheme [20], the computation cost of TPM Sign is $2G_1$ when the signature is associated. With regard to the MA-DAA scheme, the computation cost of TPM Sign remains $1G_1$ and it is the lowest among all the schemes without increasing the burden on the other protocol entities. This is because the MA-DAA scheme introduces equation (1) for signature relevance detection, and there is no need for TPM to perform any G_1 operations of *J*, *K* as in [20]. The use of equation (1) not only reduces the entire Sign computation cost in the MA-DAA scheme, but it also maintains the same level of security as the scheme in [20].

Furthermore, considering the communication bandwidth performance in NC-UAV communication systems and storage capacity of the vehicle, we need to analyze the size of the DAA certificates and signatures in the MA-DAA scheme. A smaller certificate and signature size indicate that smaller communication bandwidth and fewer storage resources are required. The comparison of the certificate and signature sizes in the MA-DAA and the other DAA schemes is listed in table 3.

In table 3, q represents the order of finite field Z_q, and h is the output of the Hash function. In $G_i\{i=1,2,T\}$, G_i constitutes asymmetric bilinear pairing $G_1\times G_2\to G_T$; G and $\mathbf{G}$ constitute symmetric bilinear pairing $G\times G\to \boldsymbol{G}$. We find out that the MA-DAA scheme improves the overall operational efficiency without additional demand for storage capacity or communication bandwidth. The certificate size of MA-DAA is $3G_1$ and the signature size is $1q+4G_1+1h$, both of which are small among all the DAA schemes.

Assume that the required RSA key length is 3072 bits, the element sizes of G_1, G_2, and G_T are 3072 bits [25], and the order of the finite field q is 256 bits. We can then compute the output signature size of the MA-DAA scheme as 12704 bits, which represents a 40% reduction in length compared with the signature size of the original BCC-DAA scheme (21707 bits in [26]). Therefore, the MA-DAA scheme has an obvious advantage in both security and performance over the BCC-DAA scheme and maintains the same security level

Table II. *Computation cost comparison among MA-DAA and the existing schemes in Sign/Verify protocol.*

Stage	Schemes	TPM	Host	Verify
Sign/Verify	BCC-DAA	$2G_N^3+3G_T$	$1G_\Gamma+1G_N+1G_N^2+2G_\Gamma^3+1G_N^4$	$4G_\Gamma+2G_N^4+1G_N^6+nG_\Gamma$
	BCL-DAA	$3G_T$	$3G_1+1G_T+3P$	$1G_T^2+1G_T^3+5P+(n+1)G_1$
	scheme in [17]	$2G_1+G_T$	$3G_1+1P$	$1G_1^2+1G_T^2+5P+nG_1$
	scheme in [23]	$4G_T$	$3G_1+2G_T+3P$	$1G_T+2G_T^2+1G_T^3+5P$
	scheme in [24]	$3G_1$	$3G_1$	$2G_1+2P+nG_1$
	scheme in [20]	$2G_1$	$4G_1$	$1G_1^2+1P^4+nG_1$
	MA-DAA	$1G_1$	$4G_1$	$1G_1^2+1P^4+nG_1$

while increasing operational efficiency and reducing communication bandwidth. We can conclude that the MA-DAA scheme is suitable for the network-connected UAV communication system with limited computing and storage resources and low transmission bandwidth.

IV. Experiment Analysis of MA-DAA Scheme

Here, we compare the experimental performance of the MA-DAA scheme with the existing scheme in [20] based on the NC-UAV computation environment. Here, an alternative simulation is used for the experiments. This means that, without considering the communication time between NC-UAV and the remote certificate Issuer and the communication time between NC-UAV and the remote Verifier, we simply focus on the time overhead on independent protocol operations of each protocol entity.

Based on this assumption, we set the Host, Issuer, and Verifier to work on the same computer. The software simulation technique [27] is used to internally install TPM on the same computer, which communicates with TPM via the hardware interface. Statistics on the time overhead of each protocol are provided in the standalone simulation environment. The scheme's parameters, such as par, are chosen according to [19].

4.1 Experimental environment

We construct the network-connected UAV experimental environment. The Host platform is the embedded computer with 2.8 GHz Intel Core(TM) i5 CPU, 4 GB of memory. The operating system is Ubuntu 9.10. The installed kernel is Linux 2.6.25. TPM simulator is tpm_emulator-0.5.1. TSS software stack [28] is trousers_0.3.1-7_i386.deb. Linux command line tools that TPM uses is tpm-tools_1.3.1-4_i386.deb. The other software packages include OpenSSL, opencryptoki, libtool, libtspi1, and libopencryptoki.

Table III. *Credential and Signature sizes among the MA-DAA scheme and the existing schemes.*

DAA schemes	Credential size	Signature size
BCL-DAA	$3\boldsymbol{G}$	$2q+3G_1+1h+2\boldsymbol{G}$
scheme in [17]	$3G_1$	$1q+5G_1+1h$
scheme in [23]	$3G_1$	$2q+2G_T+3G_1+1h$
scheme in [24]	$3G_1$	$1q+6G_1+1h$
scheme in [20]	$3G_1$	$1q+5G_1+1h$
MA-DAA	$3G_1$	$1q+4G_1+1h$

4.2 Experiment results

The Host computer exchanges messages with software TPM via the kernel simulated module /dev/tpm0. Here, the input data for one protocol is transferred by previously saved .txt files, whereas the protocol output data is stored in similar .txt files to allow other protocols to call them. For example, the DAA certificate, which NC-UAV obtains after the Join protocol, is stored in the form of credential.txt and is called in the Sign protocol. In addition, the public key data is pre-stored in the local public document for each entity's visit.

As described in Section 4.2, according to the operation order of the Join protocol in the MA-DAA scheme, it is divided into four parts: Issuer requirement, TPM response, Issuer response, and Host verification. In accordance with the Join protocol process of the MA-DAA scheme and use of the cryptographic algorithm library software package provided by OpenSSL, we employ the C language to write the client program edaa_join.c in Ubuntu 9.10. The main jobs of edaa_join.c include the Issuer requirement, Issuer response, and Host verification. The TPM response is fulfilled by software TPM: it starts when the Host inputs data to device /dev/tpm0 and ends when the Host obtains the output from /dev/tpm0.

The program computes the time overhead of each protocol in microseconds and runs by calling the timing function of the operating system. The experiment results after running edaa_join are shown in Figure 11.

From Figure 11, we can find out that the time overhead requirement of the Issuer in the Join protocol of the MA-DAA scheme is (585782-584653) μs, namely, 1129 μs; the time overhead of the TPM response is (3306207-586079) μs, namely, 2720128 μs; the time overhead of the Issuer response is (3311957-3306799) μs, namely, 5158 μs; and the time overhead of the Host verification is (3357951-3311957) μs, namely, 45994 μs. In addition, we find that some time overhead is required when TPM exchanges data with the Host. For example, if the Issuer asks for the Sign, it writes data to TPM, which costs (586079-585782) μs, namely, 297 μs. Because the Host and Issuer modules originate from the same computer, their communication time is neglected. Similarly, the communication time between the Host and Verifier in the Sign protocol is also neglected. In this experiment, we simply consider the communication time between TPM and the Host.

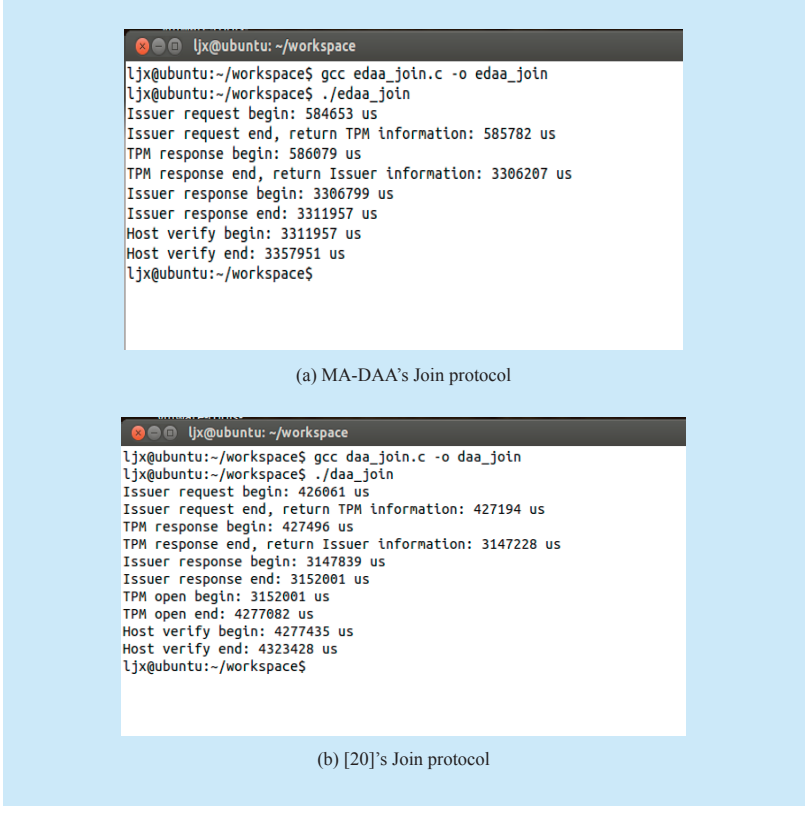

Fig. 11. *Time overhead testing results for MA-DAA's and [20]'s Join protocol.*

Table IV. *Computation cost comparison between MA-DAA scheme and scheme of [20] in Join (unit μs).*

scheme	Join steps					
MA-DAA	Issuer Request	TPM Response	Issuer Response		Host Verification	Total times
	1129	2720128	5158		45994	2772409
Scheme in [20]	Issuer Request	TPM Response	Issuer Response	TPM Open	Host Verification	Total times
	1133	2719732	4162	1125081	45993	3896101

According to table 4, the main differences between the MA-DAA scheme and the scheme in [20] can be seen in two aspects. First, the MA-DAA scheme reduces the TPM Open process. Second, the time overhead of the Issuer response in the MA-DAA scheme is 996(5158-4162) μs larger than scheme [20]. The reason is that in the MA-DAA scheme, TPM does not have the TPM Open operation, whereas the Issuer makes the operation on behalf of TPM. This way, the Issuer fulfills the group G_1 exponential operation originally conducted by TPM. With the use of batch technology, the Issuer response costs only 996 μs more than the scheme in [20], which is small relative to the TPM Open cost of 1125081 μs. By computing the total time of the Join process of the MA-DAA scheme and the scheme in [20], we can find out that the Join protocol's total time of the MA-DAA scheme is 2772409 μs. Compared with the time overhead of scheme [20] of 3896101 μs, the performance of the MA-DAA Join improves by up to approximately 29%.

Moreover, two other source programs are written: edaa_sign.c and edaa_verify.c. In accordance with Section 4, the Sign/Verify in the MA-DAA scheme is divided into three parts: Host Sign, TPM Sign, and Verify. The main jobs of edaa_sign.c include Host Sign operations and data exchange with TPM, whereas edaa_verify.c mainly fulfills the Verify operations independently. Based on a comparison of the Join time overhead, Table 5 provides the final statistics comparison of the time overhead in the Sign/Verify of DAA.

In table 5, the time overhead of the scheme in [20] is large. The improvement of the MA-DAA scheme is that, regardless of whether

the signature has correlation, the computation of each entity is smaller than the signature in scheme [20]. The reason is that MA-DAA uses a new equation instead of J and K in [20] for signature correlation detection, such that TPM Sign makes fewer group G_1 exponential operations than [20], which is where the main advantage of the MA-DAA scheme lies. Comparing the total time of the two schemes, we find that the MA-DAA scheme costs 1200602 μs. It is obvious that the performance of the MA-DAA scheme's Sign/Verify improves by up to 49%.

Table V. *Computation cost comparison between MA-DAA scheme and scheme of [20] in Sign/Verify (unit μs).*

schemes	Sign/Verify steps			
MA-DAA	Host Sign	TPM Sign	Verify	Total times
	4088	1149213	47301	1200602
Scheme in [20]	Host Sign	TPM Sign	Verify	Total times
	6312	2292876	48310	2347498

V. Conclusions

Given that many NC-UAV units operate without human intervention, user authentication will be replaced by platform identity and integrity authentication in network-connected UAV communication systems. In this paper, we proved that, with the benefits of mutual authentication improvement in security, less computational cost, and higher efficiency, the proposed MA-DAA scheme is suitable for network-connected UAV communication systems and meets the requirements of NC-UAV which is limited in computation and bandwidth resources. The MA-DAA scheme balances the requirements of security and efficiency and provides an effective anonymous platform-trusted authentication solution for NC-UAV communication systems. It is the first time that a trusted scheme is proposed in NC-UAV communication systems. With the support of this effective and secure MA-DAA scheme, the application of NC-UAVs will be extended.

Acknowledgements

This work was supported in part by the European Commission Marie Curie IRSES project "AdvIOT" and the National Natural Science Foundation of China (NSFC) under grant No.61372103.

References

[1] L. Gupta, R. Jain, G. Vaszkun, "Survey of Important Issues in UAV Communication Networks", *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, 2016, pp. 1123-1152.

[2] I. Bekmezci, O. K. Sahingoz, S. Temel, "Flying Ad-hoc Networks (FANETs): A Survey", *Ad Hoc Networks*, vol. 11, no. 3, 2013, pp. 1254–1270.

[3] K. Daniel, C. Wietfeld, "Using Public Network Infrastructures for UAV Remote Sensing in Civilian Security Operations", *the IEEE Conference on Technologies for Homeland Security*, 2011.

[4] T. C. Hong, K. Kang, K. Lim, et al., "Network Architecture for Control and Non-payload Communication of UAV", *International Conference on Information and Communication Technology Convergence, IEEE*, 2016, pp. 762-764.

[5] D. He, S. Chan, M. Guizani, "Communication Security of Unmanned Aerial Vehicles", *IEEE Wireless Communications*, no. 99, 2016, pp. 2-7.

[6] K. Hartmann, C. Steup. "The Vulnerability of UAVs to Cyber Attacks - An Approach to The Risk Assessment", *International Conference on Cyber Conflict,* IEEE, 2013, pp. 1-23.

[7] A. J. Kerns, D. P. Shepard, J. A. Bhatti, et al., "Unmanned Aircraft Capture and Control via GPS Spoofing", *Journal of Field Robotics*, vol. 31, no. 4, 2014, pp. 617-636.

[8] G. Panice, S. Luongo, G. Gigante, et al., "A SVM-based Detection Approach for GPS Spoofing Attacks to UAV", *International Conference on Automation and Computing*, 2017, pp. 1-11.

[9] S. Hagerman, A. Andrews, S. Oakes. "Security Testing of An Unmanned Aerial Vehicle (UAV)", *Cybersecurity Symposium,* IEEE, 2017, pp. 26-31.

[10] K. Yoon, D. Park, Y. Yim, et al., "Security Authentication System Using Encrypted Channel on UAV Network", *International Conference on Robotic Computing,* IEEE, 2017, pp. 393-398.

[11] M. Podhradsky, N. Hoffer, C. Coopmans, "Improving Communication Security of Open Source UAVs: Encrypting Radio Control Link", *International Conference of Unmanned Aircraft Systems, IEEE*, 2017.

[12] G. Wang, K. Lim, B. S. Lee, et al., "Handover Key Management in An LTE-based Unmanned Aerial Vehicle Control Network", *International Con-*

ference on Future Internet of Things and Cloud, IEEE, 2017, pp. 200-205.
[13] ISO/IEC 11889: Information Technology-Security Techniques-Trusted Platform Module, *http://www.trusted computinggroup.org*, 2009.
[14] TCG specification architecture overview, Reversion 1.4, *http://www.trusted-computinggroup.org*, 2011.
[15] E. Brickell, J. Camenisch, and L. Chen, "Direct Anonymous Attestation", *The 11th ACM Conference on Computer and Communications Security*, 2004, pp. 132-145.
[16] L. Chen, P. Morrissey, N. P. Smart, "Pairings in Trusted Computing", *Galbraith, Paterson (eds.) Pairing 2008, LNCS*, vol. 5209, 2008, pp. 1-17.
[17] L. Chen, P. Morrissey, N. P. Smart, "Fixing the Pairing Based Protocols", *Cryptology ePrint Archive*, Report 2009/198, http://eprint.iacr.org/2009/198.
[18] J. Camenisch, A. Lysyanskaya, "Signature Schemes and Anonymous Credentials from Bilinear Maps", *CRYPTO 2004, LNCS*, vol. 3152, 2004, pp. 56-72.
[19] L. Chen, D. Page, N. P. Smart, "On the Design and Implementation of an Efficient DAA scheme", *CARDIS 2010, LNCS*, vol. 6035, 2010, pp. 223-238.
[20] L. Chen, "A DAA Scheme Using Batch Proof and Verification", *TRUST 2010, LNCS*, vol. 6101, 2010, pp. 166-180.
[21] E. Brickell, L. Chen, J. Li, Ssimplified Security Notions for Direct Anonymous Attestation and a Concrete Scheme from Pairings", *International Journal of Information Security*, vol. 8, 2009, pp. 315-330.
[22] R. Granger, N. P. Smart, "On Computing Products of Pairings", *Cryptology ePrint Archive*, Report 2006/172, http://eprint.iacr.org/2006/172.
[23] L. Yang, J. Ma, W. Wang, "Multi-domain Direct Anonymous Attestation Scheme from Pairings", *Network and System Security, Springer International Publishing*, 2014, pp. 566-573.
[24] L. Tan, M. Zhou, "A New Process and Framework for Direct Anonymous Attestation Based on Asymmetric Bilinear Maps", *Wuhan University Journal of Natural Sciences*, vol. 16, no. 5, 2011, pp. 369-375.
[25] N. Koblitz, A. Menezes, "Pairing-based Cryptography at High Security Levels", *Cryptography and Coding LNCS*, vol. 3796, 2005, pp. 13-36.
[26] X. Chen, D. Feng, "Direct Anonymous Attestation for Next Generation TPM", *Journal of Computers*, vol. 3, 2008, pp. 43-50.
[27] M. Strasser, H. Stamer, "A software-based Trusted Platform Module Emulator", *TRUST 2008 LNCS*, vol. 4968, 2008, pp. 33-47.
[28] TCG, "TCG Software Stack (TSS) Specification", *https://www.trustedcomputing-group.org*, 2005.