SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2019, Vol. 62 032113:1–032113:15
https://doi.org/10.1007/s11432-018-9738-2

Universally composable secure geographic area verification without pre-shared secret

Junwei ZHANG[1*], Ning LU[2*], Jianfeng MA[1] & Chao YANG[1]

[1]*School of Cyber Engineering, Xidian University, Xi'an* 710071*, China;*
[2]*School of Computer Science and Technology, Xidian University, Xi'an* 710071*, China*

Received 31 August 2018/Accepted 11 January 2019/Published online 29 January 2019

Abstract The geographic area information of smart devices is required for realizing efficient area-based operations in 5G networks, Internet of Things, and so on. Because majority of smart devices are unmanned and are deployed in a hostile environment, secure geographic area verification is one of the important security issues for ensuring the accuracy of geographic area information of smart devices. In this study, we investigate the composition security of geographic area verification in a universally composable (UC) framework. First, we design the ideal functionality of geographic area verification; further, we propose a novel pre-shared secret-free secure geographic area verification protocol CAV_δ. We also propose an improved protocol CAV_δ^T exhibiting a smaller false accept ratio than that exhibited by CAV_δ. The proposed protocols can be used for verifying the geographic area information of smart devices without the requirement of any pre-shared secret during the initialization phase and additional key management when the protocols are running. Furthermore, the proposed protocols support the batch verification of multiple smart devices in one run, which is considered to be suitable for several location-critical smart devices. Subsequently, in the UC framework, we proved that our protocols achieve the necessary composition security and that our protocols exhibit an ability to resist colluding attacks.

Keywords geographic area verification, pre-shared secret-free, composition security, colluding attacks, provable security

Citation Zhang J W, Lu N, Ma J F, et al. Universally composable secure geographic area verification without pre-shared secret. Sci China Inf Sci, 2019, 62(3): 032113, https://doi.org/10.1007/s11432-018-9738-2

1 Introduction

Future networks, such as 5G networks, Internet of Things and cyber-physical systems [1], are expected to be extensively applied in several applications, including environmental monitoring, smart community, intelligent manufacturing, smart grids, and intelligent transportation. Because several smart devices that belong to future networks are usually deployed in unattended (even hostile) environments [2], security issues should be considered when smart devices exchange data and perform various tasks [3].

The location of smart devices can provide useful information about their owners. On one hand, the location of smart devices can play an important role in some applications, for instance, an intelligent fire alarm system requires the application of some location-critical devices for securely reporting the fire alarm information. On the other hand, some location-based operations and applications are achieved based on the location information of smart devices, including geographic routing [4], geographic key distribution [5], and so on.

[*] Corresponding author (email: jwzhang@xidian.edu.cn, luning@neuq.edu.cn)

To ensure the security of the location information, both secure localization and secure location verification are required. Secure localization [6], i.e., secure location determination, can be implemented on smart devices to ensure their accurate locations can be obtained with respect to adversarial behaviors. Secure location verification [7] can be used to securely verify the claimed locations of smart devices, i.e., an adversary who is not located at the target location cannot pass the verification test. When smart devices are deployed in hostile environments, adversaries may launch attacks by forging inaccurate locations, which may result in fatal consequences. For instance, a fire alarm report with an incorrect location will cause an irreparable loss of life and property.

Furthermore, it is difficult to efficiently manage large-scale unmanned devices. Therefore, some applications use the geographic area information of smart devices, instead of the location information, to realize effective group management of a large number of smart devices. For instance, in a smart community system, area-based access control must ensure that the devices in a system should be located in an area within the community. Further, an adversarial device may forge its location to gain access rights in an illegal manner. Thus, secure geographic area verification is necessary, especially for applications involving large-scale unmanned devices.

In this study, we investigated the application of secure geographic area verification to massive location-critical smart devices. However, we have faced several challenges while performing this study that can be given as follows.

First, the composition security of geographic area verification must be satisfied to perform area-based tasks such as area-based routing, key exchange, and access control. Thus, the security of geographic area verification must be guaranteed not only in a stand-alone setting but also when it is composed with other protocols.

Second, because it is difficult for large-scale unmanned devices to prepare and manage pre-shared secrets, traditional security mechanisms, including identity authentication [8, 9], password based authentication [10, 11], group key management [12], and multi-factor authentication [13], cannot be applied in such a scenario. Therefore, we did not use any pre-shared secret to realize secure geographic area verification in this study.

Third, the geographic area verification protocols should ensure security against colluding adversaries, i.e., a set of adversaries cannot pass the verification even if they collude together. Because majority of the unmanned smart devices are deployed in a hostile environment with insufficient security protections, a colluding attack, which can be considered to be an example of a typical attack, can be easily launched by adversaries.

Fourth, batch verification, especially geographic area verification, is required for large-scale unmanned devices. For a large number of devices, one-by-one verification is an inefficient method for verifying majority of the applications even although it may be one of the available solutions.

Our contributions. This study focused on the realization of pre-shared secret secure geographic area verification for location-critical devices. The proposed protocols need to satisfy the composition security against colluding adversaries. Thus, the main contributions of this study can be given as follows:

(1) We presented the security definition of circular area verification and designed the ideal functionality of circular area verification $\mathcal{F}^{D}_{\text{CAV}}$ in a universally composable (UC) framework. Simultaneously, we proved that $\mathcal{F}^{D}_{\text{CAV}}$ can imply the security definition.

(2) We proposed CAV_{δ}, which is a novel geographic area verification protocol for smart devices. The CAV_{δ} protocol provides batch verification services without the requirement of any pre-shared secret; further, CAV_{δ} was proved to satisfy the composition security against colluding adversaries.

(3) We proposed an improved protocol CAV^{T}_{δ}. When compared to CAV_{δ}, CAV^{T}_{δ} protocol includes a verification of time-of-arrival feature along with a better false accept ratio performance.

Organization. The remainder of the study can be organized as follows. Section 2 reviews the related work. Section 3 presents the system model, adversary model, and security definition. Section 4 discusses the design of the ideal funcitionality in a UC framework. In Section 5, we discuss the proposed protocols of CAV_{δ} and CAV^{T}_{δ}. The analyses of the security and performance of the proposed protocols are discussed in Sections 6 and 7, respectively. Finally, the conclusion of this study is discussed in Section 8.

2 Related work

2.1 Location verification

Secure location verification has been extensively investigated in several studies related to wireless networks. Vora et al. [14] proposed a secure location verification scheme that can verify the in-region location of provers one-by-one. Sastry et al. [7] presented a protocol that can be referred to as Echo for realizing small circular region verification by each verifier. Du et al. [15] designed a protcol that can be referred to as LAD for detecting abnormal locations based on the deployment information. Capkun et al. [16] studied the secure location verification on hidden and mobile base stations.

Chiang et al. [17] designed a secure location verification scheme against two adversaries. Hasan et al. [18] presented a witness-oriented secure location verification framework for mobile devices. Perazzo et al. [19] discussed drone path planning and secure location verification strategies. Sciancalepore et al. [20] realized a secure location verification based on meteor burst communications. However, the aforementioned schemes cannot resist a colluding attack by multiple adversaries.

Secure distance bounding, which was initially introduced by Brands et al. [21], can securely realize the verification of the upper-bound distance from a verifier to a prover. Rasmussen et al. [22] discussed the privacy leakage of the distance bounding protocol. Tippenhauer et al. [23] investigated the identity-based secure distance bounding and localization protocols. Capkun et al. [24] proposed group distance bounding protocols for multiple provers. Cremers et al. [25] analyzed distance hijacking attacks using distance bounding protocols. Perazzo et al. [26] realized a secure positioning protocol based on non-ideal distance bounding protocols. However, the distance bounding protocol can be only used to verify the upper bound and is not suitable for area verification of devices.

Chandran et al. [27] initially introduced position-based cryptography. Further, the authors proved that secure location verification cannot be realized under collusion attacks in a bare model. They realized a secure location verification protocol in the bounded-retrieval model (BRM). Furthermore, position-based cryptography was investigated in a quantum model [28]. Yang et al. [29] discussed the location privacy of position-based cryptography. Zhang et al. [30] proposed a secure location verification protocol in the UC framework. Obviously, the aforementioned position-based cryptographic protocols are based on location verification and not on area verification.

Thus, the existing location verification schemes cannot realize pre-shared secret-free secure area verification against colluding attacks.

2.2 Composition security

There are two popular composition security models for cryptographic protocols, i.e., the UC framework [31] and protocol composition logic (PCL) [32]. The UC framework is based on computational complexity and has been used to analyze plenty of cryptographic tasks, including signature [33], password-based key exchange [34], and trusted computing [35]. Particularly, Zhang et al. [30] investigated the composition security of secure positioning and proposed a UC secure location verification protocol in BRM. The PCL is based on a logic system and has been used to analyze the composition security of practical authentication protocols such as signature-based key exchange [32], symmetric key based authentication [36], and 802.11i [37].

In this study, the UC framework has been applied to analyze the composition security of geographical area verification.

3 Problem statement

3.1 System model

The system model used for secure geographic area verification comprises two types of generic entities that can be referred to as the verifier and prover.

(1) Verifier. Usually, a set of verifiers (denoted by $\mathbb{V} = \{V_1, \ldots, V_i, \ldots, V_I\}$ is trusted to verify the claimed circular area of provers. Let v_i denote the location of V_i. Note that there are at least three verifiers for performing two-dimensional area verification.

(2) Prover. There are some provers (denoted by $\mathbb{P} = \{P_1, \ldots, P_j, \ldots, P_J\}$) for verifiers $\mathbb{V}$ to pass verification on a target area. Let p_j denote the location of a prover P_j.

In this study, we emphasize on the circular area (denoted by $\mathrm{Area}(O, R)$), where its center is O and the radius is R. First, the circular area is one of the most convenient and intuitive methods to describe an area roughly. Further, the circular area is the most extensively used in various scenarios. Second, it is well known that any irregular geometric area can be covered by multiple circles, i.e., the so-called circle covering problem. Therefore, we can transform the verification on an irregular area into circular covering problems [38].

3.2 Adversary model

There are some colluding adversaries (denoted by $\mathbb{A} = \{A_1, A_2, \ldots, A_K\}$) for attacking the circular area verification protocol. Let a_k denote the location of adversary A_k.

Definition 1 (D-far colluding adversaries for position p)**.** Let $d(p, q)$ denote the distance from position p to position q. We can assume that a set of colluding adversaries $\mathbb{A}_{D \rhd p} = \{A_1, A_2, \ldots, V_K\}$ is D-far colluding adversaries for position p if the distance from any adversary in $\mathbb{A}_{D \rhd p}$ to position p is greater than D, i.e. $D < \min\{d(a_k, p) | 1 \leqslant k \leqslant K\}$.

In this study, there are colluding adversaries $\mathbb{A}_{D \rhd O} = \{A_1, A_2, \ldots, A_K\}$ that intend to pass the verification on target circular area $\mathrm{Area}(O, R)$. Further, any adversary in $\mathbb{A}_{D \rhd O}$ can access and send any message via the public communication channel.

We assume that the verifiers in $\mathbb{V}$ that are considered in the infrastructure of our protocols are trusted and uncorrupted. At the same time, there is a secure channel for the verifiers in $\mathbb{V}$ to share some information in a secure manner. In addition, we assume that a prover located in a legal area should be honest without exhibiting any adversarial behavior and run the proposed protocol in a faithful manner.

3.3 Security definition

A circular area protocol π comprises two algorithms, i.e., π = (SEL, VER), where the circular area selection algorithm SEL can output a target circular area with center O and radius R and the circular area verification algorithm $\mathrm{VER}(\mathbb{V}, O, R, P)$, given a set of verifiers $\mathbb{V} = \{V_1, V_2, \ldots, V_n\}$, can verify whether a party P is located in $\mathrm{Area}(O, R)$. We state that $\mathbb{V}$ remains uncorrupted when each V_i in $\mathbb{V}$ is uncorrupted.

Definition 2 (Secure circular area verification for D-far colluding adversaries)**.** Given a circular area $\mathrm{Area}(O, R)$ generated by SEL and enclosed in the tetrahedron defined by uncorrupted verifiers $\mathbb{V}$, a circular area verification protocol π is a secure circular area verification protocol against D-far colluding adversaries with respect to security parameter κ if $\mathbb{A}$ can output an party A^* located at position p^* satisfying $d(p^*, O) > D$ and $\mathrm{VER}(\mathbb{V}, O, R, A^*) = 1$ (i.e., Accept) with a negligible function $\varepsilon(\kappa)$ such that κ is sufficiently large for probabilistic polynomial time (PPT) D-far colluding adversaries $\mathbb{A}_{D \rhd O} = \{A_1, A_2, \ldots, A_K\}$. Formally,

$$\begin{aligned} &\mathrm{Prob}[(A^*, p^*) \leftarrow \mathbb{A}_{D \rhd O}^{\mathrm{SEL,VER}} : (O, R) \leftarrow \mathrm{SEL}, \\ &d(p^*, O) > D, \quad \mathrm{VER}(\mathbb{V}, O, R, A^*) = 1] < \varepsilon(\kappa). \end{aligned} \tag{1}$$

The definition of secure circular area verification ensures that the minimum requirements are satisfied; therefore, it exhibits the following characteristics:

(1) The D-far colluding adversaries. Any party that has a distance greater than D from O cannot succeed during the verification procedure. Obviously, the false accept event that a prover located outside $\mathrm{Area}(O, R)$ can pass the verification may occur when $D > R$.

Functionality $\mathcal{F}_{\mathbf{CAV}}^{D}$

Initialization

A set of verifiers $\mathbb{V} = \{V_1, V_2, \ldots, V_n\}$ at positions pos_1, pos_2, …, pos_n respectively.

A set of colluding adversaries $\mathbb{A}_{D \rhd O}(\mathrm{sid})$ controlled by adversary $\mathcal{S}$ in the sid session, where $\mathbb{A}_{D \rhd O}(\mathrm{sid}) = \emptyset$ and sid is the session identity.

The center $\mathrm{Cen(sid)} = \perp$ and the radius $\mathrm{Rad(sid)} = \perp$.

Circular area selection

Upon receiving the request (Circular Area Select, sid, $\mathbb{V}$) from verifiers $\mathbb{V}$:

If $\mathrm{Cen(sid)} \neq \perp$ or $\mathrm{Rad(sid)} \neq \perp$, then sends AREA_SELECTED to verifiers $\mathbb{V}$; else sends (Circular Area Select, sid, $\mathbb{V}$) to adversary $\mathcal{S}$.

Upon receiving the response (Circular Area Selected, sid, $\mathbb{V}$, O,R) from adversary $\mathcal{S}$:

(1) If $\mathrm{Area}(O, R)$ is not enclosed by verifiers $\mathbb{V}$, then sends AREA_INVALID to adversary $\mathcal{S}$.

(2) Else, sets $\mathrm{Cen(sid)} = O$, $\mathrm{Rad(sid)} = R$ and $\mathrm{SID} = (\mathrm{sid}, \mathbb{V}, O, R)$, then broadcasts (Circular Area Selected, SID).

Upon receiving (AdvPos, sid, A_k, $a\mathrm{pos}_k$) from adversary $\mathcal{S}$ $(1 \leqslant k \leqslant K)$:

If $\mathrm{Cen(sid)} \neq \perp$ and $A_k \notin \mathbb{A}_{D \rhd O}(\mathrm{sid})$ and $d(a\mathrm{pos}_k, \mathrm{Cen(sid)}) > D$, then sets $\mathrm{Pos}(\mathrm{sid}, A_k) = a\mathrm{pos}_k$ and $\mathbb{A}_{D \rhd O}(\mathrm{sid}) = \mathbb{A}_{D \rhd O}(\mathrm{sid}) \cup \{A_k\}$; else ignores this message.

Circular area verification

Upon receiving (Circular Area Verified, SID, P, f) from adversary $\mathcal{S}$ where $p = \mathrm{Pos}(\mathrm{sid}, p)$ and $f \in$ {Accept, Reject}:

(1) If $d(p, \mathrm{Cen(sid)}) > D$, then outputs (Circular Area Verified, SID, P, Reject) to $\mathbb{V}$;

(2) Else, outputs (Circular Area Verified, SID, P, f) to $\mathbb{V}$.

Figure 1 Ideal functionality $\mathcal{F}_{\mathrm{CAV}}^{D}$.

(2) Non-adaptive (static) position of adversary. This only provides the circular area verification services in a static position setting, i.e., the position of an adversary should remain constant when the protocol is running.

(3) Without completeness. This does not guarantee the completeness of the verification results, i.e., the false reject event that a prover located in $\mathrm{Area}(O, R)$ cannot pass the verification may occur without completeness.

4 Circular area verification in the UC framework

4.1 The UC framework

The UC framework [31], which is one of the computational complexity models, can be used to analyze and ensure the compositional security of cryptographic protocols. Ideal functionality plays a central role in the UC framework. Actually, ideal functionality can be treated as an incorruptible trusted party who is capable of realizing the security requirements and tasks of cryptographic protocols.

Note that the position-based cryptographic protocols that consider the time of arrival are not robust with respect to the processing and computing delays. Therefore, the execution model and proposed protocols, which are similar to those mentioned in [27], are based on the assumption that all the participants can immediately send, receive, and process data without any delay.

Further, the execution model and BRM functionality of circular area verification in the UC framework are the same as those of secure positioning [30]. Because of space limitation, only the ideal functionality of circular area verification will be presented, as depicted in Figure 1.

4.2 Ideal functionality of circular area verification

Initialization. Functionality $\mathcal{F}_{\mathrm{CAV}}^{D}$ deploys a set of verifiers $\mathbb{V}$. Let sid denote the unique session identity of a circular area verification protocol run. Let $\perp$ denote a NULL value. Let Pos(sid, P) denote the location of party P in the sid session. Let Cen(sid) and Rad(sid) denote the center and the radius of the target circular area in the sid session. Adversary $\mathcal{S}$ can control a set of colluding adversaries $\mathbb{A}_{D \rhd O}$.

Determine the target circular area. The functionality forwards the request (Circular Area Select, sid, $\mathbb{V}$) to adversary $\mathcal{S}$ if a circular area has not been selected, i.e. $\mathrm{Cen(sid)} = \perp$ or $\mathrm{Rad(sid)} = \perp$. If a target circular area exists, the functionality rejects the request and send AREA_SELECTED to verifiers.

Adversary $\mathcal{S}$ determines $\text{Area}(O, R)$ as a target circular area enclosed by verifiers $\mathbb{V}$ by sending the response (Circular Area Selected, sid, $\mathbb{V}$, O, R). Thus, the security of circular area verification does not rely on the selection of target area.

D-far colluding adversaries. When $\text{Cen(sid)} \neq \perp$ (i.e., the target circular area has been determined), adversary $\mathcal{S}$ can deploy some D-far adversaries for center O according to the target circular area in the sid session. If $d(a\text{pos}_k, \text{Cen(sid)}) \leqslant D$, the functionality will not proceed this request.

Non-adaptive (static) position of adversary. The location of an adversary in $\mathbb{A}_{D \rhd O}(\text{sid})$ is static in this session. If Pos(sid, A_k)$\neq \perp$ (which means that the position of A_k has been determined), the request (AdvPos, sid, A_k, $a\text{pos}_k$) will be ignored.

Without pre-shared secret. There is non pre-shared knowledge between verifiers $\mathbb{V}$ and a prover P in functionality $\mathcal{F}^D_{\text{CAV}}$. One of the challenges of designing secure circular area verification protocols is that how to ensure security without any pre-shared secret.

Adversary behavior. Adversary $\mathcal{S}$ controls the adversaries in $\mathbb{A}_{D \rhd O}(\text{sid})$ to send any message and read all the messages when they pass by these adversaries. Further, the verification result is generated by the request (Circular Area Verified, $\text{SID}, P, p = \text{Pos}(\text{sid}, P), f$) from adversary $\mathcal{S}$. Prover P may be an honest prover located in the target area, or may be an adversary controlled by adversary $\mathcal{S}$ for passing the verification. Therefore, $\mathcal{F}^D_{\text{CAV}}$ cannot ensure the completeness of the verification.

Security. The D-far colluding adversaries cannot output D-far party P for passing the verification successfully. Functionality $\mathcal{F}^D_{\text{CAV}}$ outputs (Circular Area Verified, SID, P, Reject) if $d(p, \text{Cen(sid)}) > D$.

4.3 $\mathcal{F}^D_{\text{CAV}}$ implies Definition 1

Theorem 1. If protocol π can securely realize the ideal functionality $\mathcal{F}^D_{\text{CAV}}$, then protocol π satisfies Definition 1.

Proof. If protocol π does not satisfy Definition 1, we can construct an environment $\mathcal{Z}$, which can differentiate the real protocol running (REAL) and the ideal procedure (IDEAL) with a non-negligible probability.

If protocol π is not secure, then there is a real adversary $\mathcal{G}$, which controls a set of the colluding adversaries $\mathbb{A}$. Further, we have that $\text{Prob}[(A^*, p^*) \leftarrow \mathcal{G}^{\text{SEL,VER}} : (O, R) \leftarrow \text{SEL}; d(p^*, O) > D; \text{VER}(\mathbb{V}, O, R, A^*) = 1] \geqslant \varepsilon(\kappa)$, where $\varepsilon(\kappa)$ is a non-negligible probability with respect to security parameter κ. Based on adversary $\mathcal{G}$, environment $\mathcal{Z}$ can be constructed as follows.

(1) When $\mathcal{G}$ requests to run SEL, $\mathcal{Z}$ hands (Circular Area Select, sid, $\mathbb{V}$) to verifiers $\mathbb{V}$.

(2) When the simulated protocol outputs (Circular Area Selected, sid, $\mathbb{V}$, O, R), $\mathcal{Z}$ hands it to $\mathcal{G}$.

(3) When $\mathcal{G}$ generates a party P located at position p, $\mathcal{Z}$ activates the verification procedure for party P in the simulated protocol.

(4) When $\mathcal{G}$ controls A_i to send any message, $\mathcal{Z}$ hands this instruction to A_i. Further, $\mathcal{Z}$ hands the messages from all the adversaries to $\mathcal{G}$.

(5) When the simulated protocol outputs (Circular Area Verified, SID, P, f): If $d(p, \text{Cen(sid)}) > D$ and f = Accept, then $\mathcal{Z}$ outputs 1; otherwise, $\mathcal{Z}$ outputs 0.

REAL. When $\mathcal{G}$ can launch a successful attack with non-negligible probability $\varepsilon(\kappa)$, π generates a false result, i.e., (Circular Area Verified, SID, P, Accept), where $d(p, \text{Cen(sid)}) \leqslant D$. Therefore, the probability that environment $\mathcal{Z}$ outputs 1 is equal to the probability that $\mathcal{G}$ attacks successfully when $\mathcal{Z}$ is interacting with the real protocol running, i.e., $\text{Prob}[\mathcal{Z}{=}\ 1|\ \text{REAL}] = \varepsilon(\kappa)$.

IDEAL. When $\mathcal{Z}$ is interacting with $\mathcal{F}^D_{\text{CAV}}$ in the ideal procedure, functionality $\mathcal{F}^D_{\text{CAV}}$ never outputs (Circular Area Verified, $\text{SID}, P, \text{Accept}$) if $d(p, \text{Cen(sid)}) \leqslant D$, thus $\mathcal{Z}$ outputs 0. Therefore, $\text{Prob}[\mathcal{Z} = 1 \mid \text{IDEAL}] = 0$.

Therefore, $\mathcal{Z}$ can differentiate between REAL and IDEAL with a non-negligible probability $\varepsilon(\kappa)$ with non-negligible probability $|\text{Prob}[\mathcal{Z} = 1|\text{REAL}] - \text{Prob}[\mathcal{Z} = 1|\text{IDEAL}]| = \varepsilon(\kappa)$. Furthermore, we have a contradiction that π cannot securely realize functionality $\mathcal{F}^D_{\text{CAV}}$.

Table 1 Notations

Notation	Description
V_i	The ith verifier located at v_i
P_j	The jth prover located at p_j
A_k	The kth adversary located at a_k
Area(O, R)	The circular area with center O and radius R
X_i	The ith BRM message
n_i	The ith random number
C	The traveling speed of messages
$d(p, q)$	The distance between position p and position q
$F(\cdot)$	A secure BSM pseudorandom generator
$g(\cdot)$	A secure MAC function

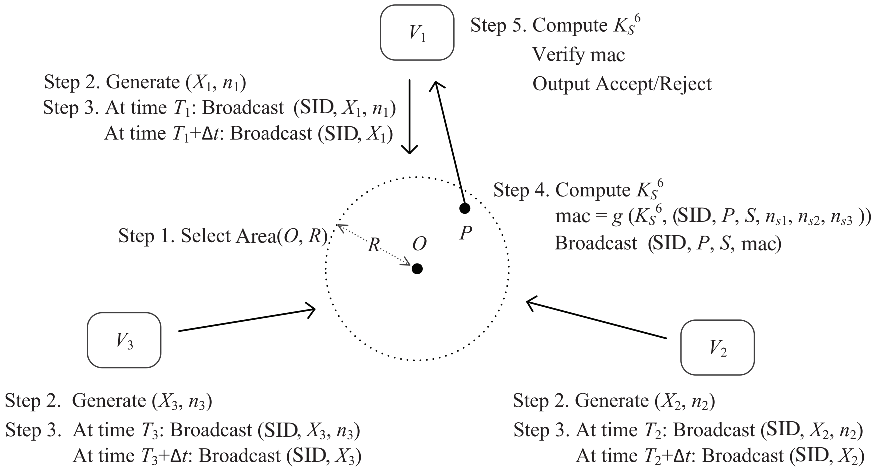
V_1
Step 5. Compute K_S^6
Verify mac
Output Accept/Reject
Step 2. Generate (X_1, n_1)
Step 3. At time T_1: Broadcast (SID, X_1, n_1)
At time $T_1+\Delta t$: Broadcast (SID, X_1)
Step 4. Compute K_S^6
mac = g (K_S^6, (SID, P, S, n_{s1}, n_{s2}, n_{s3}))
Broadcast (SID, P, S, mac)
Step 1. Select Area(O, R)
R
O
P
V_3
V_2
Step 2. Generate (X_3, n_3)
Step 3. At time T_3: Broadcast (SID, X_3, n_3)
At time $T_3+\Delta t$: Broadcast (SID, X_3)
Step 2. Generate (X_2, n_2)
Step 3. At time T_2: Broadcast (SID, X_2, n_2)
At time $T_2+\Delta t$: Broadcast (SID, X_2)

Figure 2 Protocol CAV.

5 The proposed protocols

5.1 Notations

The notations used in this study are shown in Table 1.

5.2 Protocol CAV$_\delta$

The protocol CAV$_\delta$ (shown in Figure 2) is as follows.

Initialization. A set of verifiers $\mathbb{V} = \{V_1, V_2, V_3\}$ located at v_1, v_2, v_3 respectively.

BSM PRG $F : \{0,1\}^n \times \{0,1\}^l \rightarrow \{0,1\}^l$. MAC $g : \{0,1\}^* \times \{0,1\}^l \rightarrow \{0,1\}^m$.

Circular area selection. When receiving (Circular Area Select, sid, $\mathbb{V}$), where sid is the session identity:

Step 1. Verifiers $\mathbb{V}$ select Area(O, R) as a target circular area, where O is center and R is radius.

Step 2. Verifier V_i $(1 \leqslant i \leqslant 3)$ in $\mathbb{V}$ selects a random number n_i and messages $(X_i,\ X_{i+3})$ with high min-entropy $(\delta + \beta)n$, then shares (X_i, X_{i+3}, n_i) with other verifiers in $\mathbb{V}$ over a secure channel. Verifier V_i sets $M_i = (\text{SID}, X_i, n_i)$ and $M_{i+3} = (\text{SID}, X_{i+3})$, where $\text{SID} = (\text{sid}, \mathbb{V}, O, R)$. Let T and T' denote the time at which M_i and M_{i+3} reach the center O. We have that $T' = T + \Delta t$, where $\Delta t = 2\delta R/C$. Then, verifier V_1 outputs (Circular Area Selected, sid, $\mathbb{V}$, O, R).

Circular area verification. When receiving (Circular Area Verify, SID, P):

Step 3. Let $d(p, q)$ denote the euclidean distance between position p and position q. At time $T_i = T - d(v_i, O)/C$, verifier V_i in $\mathbb{V}$ broadcasts M_i. At time $T_i' = T_i + \Delta t$, V_i in $\mathbb{V}$ broadcasts M_{i+3}.

Note that in the UC framework, functionality $\mathcal{F}_{\text{BRM}}$ can be applied to broadcast X_i, i.e., V_i sends (Broadcast BRMessage, sid, X_i, i) to $\mathcal{F}_{\text{BRM}}$.

Step 4. When receives M_i at time t_i and M_{i+3} at time t_{i+3}, prover P sets $S = s_1s_2s_3$ as the sequence of receiving messages $\{M_i\}$, where $s_i \in \{1, 2, 3\}$ and $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3}$. If located in Area$(O, R)$, prover P can compute k_S^6, where

$$k_S^j = \begin{cases} F(X_S^j, k_S^{j-1}), & 2 \leqslant j \leqslant 6, \\ n_{s_1}, & j = 1, \end{cases}$$

and

$$X_S^j = \begin{cases} X_{s_j}, & 1 \leqslant j \leqslant 3, \\ X_{s_{j-3}+3}, & 4 \leqslant j \leqslant 6. \end{cases}$$

Then, prover P computes mac $= g(k_S^6, (\mathrm{SID}, P, S, n_{s_1}, n_{s_2}, n_{s_3}))$, and broadcasts (SID, P, S, mac).

Prover P can only obtain a notice $(\mathsf{Broadcasted\ BRMessage}, \mathrm{sid}, V_i, i)$, but not receive the value of X_i in the UC framework. Further, prover P can request a computation service about X_i by sending $(\mathsf{Compute}, \mathrm{sid}, i, Y)$ to $\mathcal{F}_{\mathrm{BRM}}$, where Y is an function of X_i, and obtain the computation result y from $\mathcal{F}_{\mathrm{BRM}}$.

Step 5. When verifier V_1 in $\mathbb{V}$ receives (SID, P, S, mac): V_1 computes k_S^6, and verifies the correctness of mac using k_S^6. If success, verifier V_1 outputs (Circular Area Verified, SID, P, Accept). Otherwise, V_1 outputs (Circular Area Verified, SID, P, Reject).

5.3 Protocol CAV_δ^T

In this part, we present an improved version of CAV_δ with time verification, i.e., protocol CAV_δ^T. The procedure of protocol $\mathrm{CAV}_{\delta=1}^T$ from Step 1 to 4 is same as that of protocol CAV_δ except Step 5 in the circular area verification phase.

Step 5. When each V_i in $\mathbb{V}$ receives (SID, P, S, mac) at time RT_i: If $RT_i \leqslant T + (2(\delta+1)R + d(O, v_i))/C$ and the mac is valid, verifier V_i sets $b_i = 1$; otherwise $b_i = 0$. If $b_1 = b_2 = b_3 = 1$, verifier V_1 outputs (Circular Area Verified, SID, P, Accept). Otherwise, verifier V_1 outputs (Circular Area Verified, SID, P, Reject).

Compared to CAV_δ, the only additional operation of protocol CAV_δ^T is that each verifier V_i in protocol CAV_δ^T should verify the time of receiving the response (SID, P, S, mac) sent by prover P.

5.4 Discussion

Without any pre-shared secret. Both CAV_δ and CAV_δ^T are pre-shared secret-free geographic area verification protocols. In the initialization phase, there is on pre-shared symmetric key, public/private key pair between verifier V_i in $\mathbb{V}$ and prover P. During the circular area selection phase and the circular area verification phase, prover P only uses BSM PRG F and MAC g to compute the response for verifiers. Thus, both CAV_δ and CAV_δ^T are not dependent on key management technology.

Batch verification. Both CAV_δ and CAV_δ^T can realize batch verification on several smart devices in a run. In the circular area verification phase, each P_i in $\mathbb{P}$ can compute one K_S^6 and generate a valid response $(\mathrm{SID}, P_i, S, \mathrm{mac})$ to pass the verification if P_i is within Area(O, R) (See the completeness analysis in Section 6). Therefore, the proposed protocols support batch verification for a large number of location-critical smart devices.

BSM PRG and $\mathcal{F}_{\mathrm{BRM}}$. $\mathcal{F}_{\mathrm{BRM}}$ can ensure that all adversaries retrieve part of X_i with high min-entropy information $(\alpha + \beta)n$ and the total retrieval information S_i about X_i is bounded by βn, i.e., $|S_i| \leqslant \beta n$. Given S_i and r, BSM PRG can ensure that any adversary cannot compute $y = F(X_i, r)$. However, prover P within Area(O, R) can compute K_S^6 and pass geographic area verification in the proposed protocols. The detailed analysis can be referred to Section 6.

Composition security against colluding adversaries. On one hand, both CAV_δ and CAV_δ^T can securely realize ideal functionality $\mathcal{F}_{\mathrm{CAV}}^D$ in the UC framework. Thus, the proposed protocols can be applied to construct geographic area-based tasks, such as area-based message authentication, area-based group key exchange, and so on. On the other hand, the colluding adversaries cannot compute K_S^6 to pass geographic area verification because these adversaries only retrieve the information S_i about X_i where $|S_i| \leqslant \beta n$ in BRM. Section 6 presents the detailed analysis of composition security against colluding adversaries.

CAV$_\delta$ vs CAV$_\delta^T$. Protocol CAV$_\delta^T$ has a better performance for the view of false accept ratio because a prover with a false accept event in CAV$_\delta$ may not pass time verification in CAV$_\delta^T$. Section 7 presents the comparison results on the false accept ratio of CAV$_\delta$ and CAV$_\delta^T$.

Extension in 3-dimensions. The proposed protocols can be extended to perform three-dimensional space verification. First, four verifiers, i.e., $\mathbb{V} = \{V_1, V_2, V_3, V_4\}$ are necessary in the Initialization phase. Second, prover P should compute K_S^8 according to receiving messages $\{M_i\}$, where $S = s_1s_2s_3s_4$ is the receiving sequence. Then, prover P computes mac based on K_S^8, and verifiers also verify the correctness of mac using K_S^8.

Location privacy. In the proposed protocols, verifier V_i broadcasts $M_i = (\mathrm{SID}, X_i, n_i)$ and $M_{i+3} = (\mathrm{SID}, X_{i+3})$, and prover P sends the response (SID, P, S, mac). Obviously, the above messages do not reveal any information about the location of prover P. Therefore, the colluding adversaries cannot obtain the location information of prover P based on the messages in our protocols. In other words, the proposed protocols provide location privacy protection from the view of information security.

However, the colluding adveraires can infer the origin of a message with the help of the characteristic of broadcast communications. If prover P at (x, y) broadcasts (SID, P, S, mac) at time t, adversary A_i at (x_i, y_i) can receive the message at time t_i and have an equation $(x - x_i)^2 + (y - y_i)^2 = C^2(t_i - t)^2$. Then, the colluding adversaries can construct a equation set to compute (x, y).

6 Security analysis

In this section, we first analyze the correctness of the proposed protocols. Then, we show that the proposed protocols securely realize the circular area verification functionality. For simplicity, we set $\delta = 1$.

6.1 Completeness

Theorem 2. Given Area(O, R) and verifiers $\mathbb{V}$, protocol CAV$_{\delta=1}$ can output (Circular Area Verified, SID, P, Accept) if $p = \mathrm{Pos}(\mathrm{sid}, P)$ and $d(p, O) \leqslant R$.

Proof. If prover P within Area(O, R) can compute k_S^6 and send a valid message authentication code mac $= g(k_S^6, (\mathrm{SID}, P, S, n_{s_1}, n_{s_2}, n_{s_3}))$, protocol CAV$_{\delta=1}$ outputs (Circular Area Verified, SID, P, Accept).

In protocol CAV$_{\delta=1}$,

$$k_S^j = \begin{cases} F(X_S^j, k_S^{j-1}), & 2 \leqslant j \leqslant 6, \\ n_{s_1}, & j = 1. \end{cases}$$

Let t_i denote the time when X_i passes prover P in the running of protocol CAV$_{\delta=1}$. Prover P can compute k_S^6 if and only if the inequality $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3} \leqslant t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$ holds.

In protocol CAV$_{\delta=1}$, M_i or M_{i+3} is broadcasted by V_i at time T_i or T_i', and they reach center O at time T or T', respectively.

(1) If prover P is located at center O, we have that $t_{s_1} = t_{s_2} = t_{s_3} = T < T' = t_{s_1+3} = t_{s_2+3} = t_{s_3+3}$. Obviously, prover P can set $S = 123$ in general and compute k_{123}^6, then generate a valid mac.

(2) If prover P is located in Area(O, R) except center O, we have that $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3}$, where $S = s_1s_2s_3$. Furthermore, $t_{s_1} < T < t_{s_3}$ due to that if $t_{s_1} \geqslant T$ and $t_{s_3} \leqslant T$ leads to $t_{s_1} = t_{s_2} = t_{s_3}$, which is a contradiction that prover P is in Area(O, R) but not center O. if prover P is in Area(O, R), $T - R/C \leqslant t_{s_1} \leqslant T - d(p, O)/C < T < T + d(p, O)/C \leqslant t_{s_3} \leqslant T + R/C$. Because V_i broadcasts M_i and M_{i+3} at time T_i and T_i', respectively, there is a similar result that $t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$ and $T' - R/C \leqslant t_{s_1+3} \leqslant T' - d(p, O)/C < T' < T' + d(p, O)/C \leqslant t_{s_3+3} \leqslant T' + R/C$ when $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3}$. Therefore, $t_{s_3} \leqslant T + R/C = T' - R/C \leqslant t_{s_1+3}$ can be derived because $T' = T + 2R/C$ in protocol CAV$_{\delta=1}$. The inequality $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3} \leqslant t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$ holds.

In summary, prover P in Area(O, R) can pass area verification successfully.

Similarly, protocol $\mathrm{CAV}^{T}_{\delta=1}$ also ensures the completeness because prover P in $\mathrm{Area}(O,R)$ can also compute k^{6}_{S} and send the valid response. Thus, we have the following lemma.

Lemma 1. Given $\mathrm{Area}(O,R)$ and verifiers $\mathbb{V}$, protocol $\mathrm{CAV}^{T}_{\delta=1}$ can output ($\mathsf{Circular\ Area\ Verified}$, SID, P, Accept) if $p = \mathrm{Pos}(\mathrm{sid}, P)$ and $d(p,O) \leqslant R$.

Note that the completeness of CAV_{δ} and $\mathrm{CAV}^{T}_{\delta}$ may be unsatisfied and generate the false reject events when $\delta < 1$. The analysis on false reject ratio is shown in Section 7.

6.2 UC security

Theorem 3. Given $\mathrm{Area}(O,R)$, there exists D such that protocol $\mathrm{CAV}_{\delta=1}$ can realize $\mathcal{F}^{D}_{\mathrm{CAV}}$ in the $\mathcal{F}_{\mathrm{BRM}}$-hybrid model.

Proof. The proving process have three phases in the following parts. In part (1), we find an unique D in a running of protocol $\mathrm{CAV}_{\delta=1}$ given verifiers $\mathbb{V}$ and $\mathrm{Area}(O,R)$. In part (2), we construct an adversary $\mathcal{S}$ to simulate dummy adversary $\mathcal{A}$ for environment $\mathcal{Z}$. In part (3), we analyze that environment $\mathcal{Z}$ cannot distinguish between REAL and IDEAL with a non-negligible probability.

(1) Determine the value of D.

In the proof of Theorem 2, prover P can complete the verification on $\mathrm{Area}(O,R)$ if and only if $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3} \leqslant t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$.

According to the messages M_i and M_{i+3} from V_i in protocol $\mathrm{CAV}_{\delta=1}$, we have

$$t_{s_1} = T_{s_1} + d(p, v_{s_1})/C = T - d(v_{s_1}, O)/C + d(p, v_{s_1})/C;$$

$$t_{s_2} = T_{s_2} + d(p, v_{s_2})/C = T - d(v_{s_2}, O)/C + d(p, v_{s_2})/C;$$

$$t_{s_1} = t_{s_2} \Longleftrightarrow d(p, v_{s_2}) - d(p, v_{s_1}) = d(v_{s_2}, O) - d(v_{s_1}, O).$$

It is well known that a hyperbola $\mathbb{H}$ is a set of points, such that for any point P in $\mathbb{H}$, given two fixed points (F_1, F_2), $||PF_1| - |PF_2||$ is constant, usually denoted by $c(c>0)$, i.e., $\mathbb{H} = \{P \mid ||PF_2| - |PF_1|| = c\}$.

Thus, $t_{s_1} = t_{s_2}$ can be represented as one branch of hyperbola $\mathbb{H}_{t_{s_1}=t_{s_2}}$ when (v_{s_1}, v_{s_2}) are the two fixed points and $d(p, v_{s_2}) - d(p, v_{s_1}) = d(v_{s_2}, O) - d(v_{s_1}, O)$ is constant. By the way, $\mathbb{H}_{t_{s_1}=t_{s_2}}$ is transformed into a straight line $\mathbb{L}_{t_{s_1}=t_{s_2}}$ if $d(v_{s_2}, O) - d(v_{s_1}, O) = 0$. In fact, because there is no difference between $\mathbb{L}_{t_{s_1}=t_{s_2}}$ and one branch of $\mathbb{H}_{t_{s_1}=t_{s_2}}$ from the view of our analysis here, we will not consider the situation about $\mathbb{L}_{t_{s_1}=t_{s_2}}$ in the following part.

Similarly, $t_{s_2} = t_{s_3}$, $t_{s_3} = t_{s_1+3}$, $t_{s_1+3} = t_{s_2+3}$ and $t_{s_2+3} = t_{s_3+3}$ can be shown as $\mathbb{H}_{t_{s_2}=t_{s_3}}$, $\mathbb{H}_{t_{s_3}=t_{s_1+3}}$, $\mathbb{H}_{t_{s_1+3}=t_{s_2+3}}$ and $\mathbb{H}_{t_{s_2+3}=t_{s_3+3}}$, respectively. The only difference is that for $\mathbb{H}_{t_{s_3}=t_{s_1+3}}$, the two fixed points are (v_3, v_1), and the constant value of $d(p, v_{s_3}) - d(p, v_{s_1})$ is $d(v_{s_3}, O) - d(v_{s_1}, O) + 2R$ in $\mathbb{H}_{t_{s_3}=t_{s_1+3}}$. Note that $\mathbb{H}_{t_{s_1}=t_{s_2}} = \mathbb{H}_{t_{s_1+3}=t_{s_2+3}}$ and $\mathbb{H}_{t_{s_2}=t_{s_3}} = \mathbb{H}_{t_{s_2+3}=t_{s_3+3}}$ because $t_{s_1} = t_{s_2}$ is equal to $t_{s_1+3} = t_{s_2+3}$ and $t_{s_2} = t_{s_3}$ is equal to $t_{s_2+3} = t_{s_3+3}$ in protocol $\mathrm{CAV}_{\delta=1}$.

Finally, according to the six values of $\{S \mid S = s_1 s_2 s_3, t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3}\}$, the inequality $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3} \leqslant t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$ constructs an area (denoted by $\mathrm{Area}_{\mathbb{H}}$, see Figure 3) which is enclosed by six branches of $\mathbb{H}_{t_3=t_4}$, $\mathbb{H}_{t_2=t_4}$, $\mathbb{H}_{t_2=t_6}$, $\mathbb{H}_{t_1=t_6}$, $\mathbb{H}_{t_1=t_5}$ and $\mathbb{H}_{t_3=t_5}$. Six intersection points $\{h_j \mid h_j = \mathbb{H}_{t_j=t_u} \cap \mathbb{H}_{t_j=t_w}, j \in [1,6]\}$ as the vertexes of $\mathrm{Area}_{\mathbb{H}}$ are generated by six hyperbolic branches. Further, $\mathbb{H}_{t_1=t_2}$, $\mathbb{H}_{t_2=t_3}$ and $\mathbb{H}_{t_1=t_3}$ divide $\mathrm{Area}_{\mathbb{H}}$ into six sub-areas, and prover P located in anyone of these sub-areas can compute a valid k^{6}_{S}.

Therefore, we can set D as $\max\{d(h_j, O) \mid j \in [1,6]\}$. For example, D should be equal to $d(h_3, O)$ in Figure 3. It is obvious that $\mathrm{Area}(O,R) \subset \mathrm{Area}_{\mathbb{H}} \subset \mathrm{Area}(O,D)$ and any point outside $\mathrm{Area}(O,D)$ is not satisfy the inequality $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3} \leqslant t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$.

In the following part, we prove that for all the colluding adversaries $\mathbb{A}_{D \rhd O}$, environment $\mathcal{Z}$ can distinguish between REAL and IDEAL with only a negligible probability.

(2) Construct adversary $\mathcal{S}$.

Let $\mathcal{A}$ be dummy adversary that controls the colluding adversaries $\mathbb{A}_{D \rhd O} = \{A_1, A_2, \ldots, A_K\}$. Adversary $\mathcal{S}$ (shown in Figure 4) activates adversary $\mathcal{A}$ in real process. $\mathcal{S}$ forwards the instruction from $\mathcal{Z}$

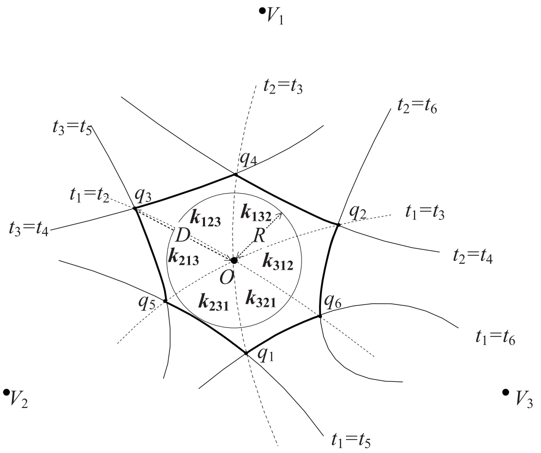

Figure 3 Area$_{\mathbb{H}}$ derived from the inequality.

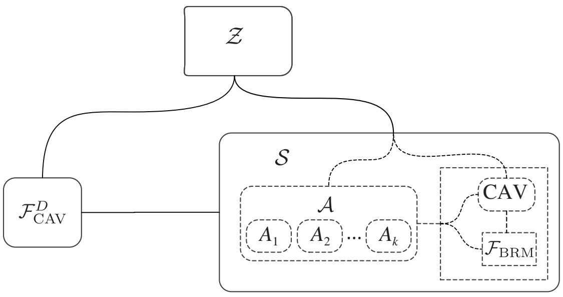

Figure 4 Adversary $\mathcal{S}$.

to $\mathcal{A}$ and copies the output of $\mathcal{A}$ to $\mathcal{Z}$. Further, $\mathcal{S}$ simulates for $\mathcal{A}$ the running of protocol $\text{CAV}_{\delta=1}$ with $\mathcal{F}_{\text{BRM}}$.

The detailed description of adversary $\mathcal{S}$ is as follows:

(i) When receiving (Circular Area Select, sid, $\mathbb{V}$) from $\mathcal{F}^D_{\text{CAV}}$, runs a simulated copy of protocol $\text{CAV}_{\delta=1}$ with the input (Position Initialize, sid, $\mathbb{V}$) for $\mathcal{A}$.

(ii) When receiving (Circular Area Selected, sid, $\mathbb{V}, O, R$) from $\mathcal{A}$, hands (Circular Area Selected, sid, $\mathbb{V}, O$, R) to $\mathcal{F}^D_{\text{CAV}}$.

(iii) When receiving (Circular Area Verify, SID, P) from $\mathcal{Z}$, runs a copy of $\text{CAV}_{\delta=1}$ with input (Circular Area Verify, SID, P).

(iv) When V_i sends M_i or M_{i+3}, simulates message M_i or M_{i+3} from V_i for adversary $\mathcal{A}$.

(v) When receiving (Broadcast BRMessage, sid, X_i, i) from V_i, sends (Broadcast BRMessage, sid, X_i, i) to $\mathcal{F}_{\text{BRM}}$.

(vi) When receiving (Retrieve BRMessage, sid, i, F) from $\mathcal{A}$, sends (Retrieve BRMessage, sid, i, F) to $\mathcal{F}_{\text{BRM}}$ and hands the response (Retrieve BRMessage, sid, i, f) from $\mathcal{F}_{\text{BRM}}$ to $\mathcal{A}$.

(vii) When $\mathcal{F}_{\text{BRM}}$ outputs (Broadcasted BRMessage, sid, V_i, i) to party P, hands it from $\mathcal{F}_{\text{BRM}}$ to P.

(viii) When party P requests (Compute, sid, i, Y), hands it from P to $\mathcal{F}_{\text{BRM}}$. If $\mathcal{F}_{\text{BRM}}$ outputs (Computed, sid, i, Y, y), forwards the response to P.

(ix) When P broadcasts (SID, P, S, mac), simulates message (SID, P, S, mac) from P for adversary $\mathcal{A}$.

(x) When the simulated $\text{CAV}_{\delta=1}$ outputs (Circular Area Verified, SID, P, f) (f = Accept or Reject), sends (Circular Area Verified, SID, P, f) to $\mathcal{F}^D_{\text{CAV}}$.

(3) REAL and IDEAL are indistinguishable.

Let FALSE_ACCEPT denote the event that protocol $\text{CAV}_{\delta=1}$ outputs (Prover Verified, SID, P', Accept), where $d(p' = \text{Pos}(\text{sid}, P'), \text{Cen}(\text{sid})) > D$. Thus, environment $\mathcal{Z}$ cannot distinguish between REAL and IDEAL unless FALSE_ACCEPT happens. Then, we analyze that FALSE_ACCEPT occurs with only a negligible probability.

FALSE_ACCEPT occurs only if adversary $\mathcal{A}$ can forge a valid (SID, P', S, mac) to pass the verification of verifiers $\mathbb{V}$, where $d(p' = \text{Pos}(\text{sid}, P'), \text{Cen}(\text{sid})) > D$. Assuming that adversary $\mathcal{A}$ can forge such a message (SID, P', S, mac) with probability ϵ, it may be one of the following two cases.

Case 1. Adversary $\mathcal{A}$ can compute anyone mac key k^6_S.

Let X^*_i denote the total retrival information about X_i with high min-entropy $(\alpha+\beta)n$, where $|X^*_i| \leqslant \beta n$. Let B_1 denote a non-colluding adversary in $\mathbb{A}_{D \rhd O}$, i.e., adversary B_1 can only receive the messages from verifiers.

For adversary B_1 in $\mathbb{A}_{D \rhd O}$, we have that $t_{s_1} < T - R/C < t_{s_1+3} = t_{s_1} + 2R/C < T + R/C < t_{s_3}$. Because $t_{s_1+3} < t_{s_3}$, adversary B_1 can only compute k^3_S later than t_{s_1+3} but not obtain the valid k^4_S based on

Table 2 Comparison with related studies

Protocol	RB	DB	SP	SPreg	Ours
Area verification	✓	×	×	×	✓
Batch verification	✓	✓	×	×	✓
Resist colluding attacks	×	✓	✓	✓	✓
Without pre-shared key	✓	×	✓	×	✓
Composition security	–	–	–	✓	✓

$X^*_{s_1+3}$. Otherwise, there is a contradiction of the definition of BSM PRG [27]. Therefore, adversary B_1 cannot compute k^6_S because $t_{s_1} \leqslant t_{s_2} \leqslant t_{s_3} \leqslant t_{s_1+3} \leqslant t_{s_2+3} \leqslant t_{s_3+3}$ does not hold.

Let B_2 denote a colluding adversary in $\mathbb{A}_{D \rhd O}$. Adversary in B_2 can obtain and share the computation results with other colluding adversaries in $\mathbb{A}_{D \rhd O}$.

Lemma 2. If adversary B_2 receives $f(X_i)$ from other adversaries at time t'_i and adversary B_2 directly receives information X_i from a verifier at time t_i, then $t_i \leqslant t'_i$.

Proof sketch. According to the triangle inequality, we have that Lemma 2 holds.

According to Lemma 2, adversary B_2 can only receive k^j_S or any function about X^j_S from other adversaries in $\mathbb{A}_{D \rhd O}$ later than the time receiving X^j_S directly from verifiers. Further, adversary B_2 cannot compute k^4_S because adversary B_2 can receive k^3_S later than t_{s_3} ($> t_{s_1+3}$). Therefore, adversary B_2 cannot compute k^6_S with a non-negligible probability.

Similar to the proof of position-based key exchange in [27], the security of protocol $\mathrm{CAV}_{\delta=1}$ can also be proven by a reduction to an intrusion resilient random secret sharing protocol. We omit the detailed analysis due to space limitations.

In a word, adversary $\mathcal{A}$ can compute k^6_S with only a negligible probability in Case 1.

Case 2. Given a message m^*, adversary $\mathcal{A}$ can forge a valid message authentication code $\mathrm{mac}^* = g(k, m^*)$ without k.

Obviously, adversary $\mathcal{A}$ can forge such a valid mac with only a negligible probability in Case 2 if g is a secure MAC function. Otherwise, we can construct an adversary to attack the MAC function g which is contract to the definition of secure MAC function [31].

Thus, protocol $\mathrm{CAV}_{\delta=1}$ can securely realize $\mathcal{F}^D_{\mathrm{CAV}}$ in the $\mathcal{F}_{\mathrm{BRM}}$-hybrid model.

Lemma 3. Given $\mathrm{Area}(O, R)$, there exists D such that protocol $\mathrm{CAV}^T_{\delta=1}$ can realize $\mathcal{F}^D_{\mathrm{CAV}}$ in the $\mathcal{F}_{\mathrm{BRM}}$-hybrid model.

In the proof of Lemma 3, the value of D in $\mathrm{CAV}^T_{\delta=1}$ is less than that of $\mathrm{CAV}_{\delta=1}$. Let $\mathbb{C}_i$ denote the upper bound of the equality $d(p, v_i) = d(O, v_i) + R$. There is an area $\mathrm{Area}_{\mathbb{C}}$ which is enclosed by $\mathbb{C}_1$, $\mathbb{C}_2$ and $\mathbb{C}_3$. Therefore, the value of D in $\mathrm{CAV}^T_{\delta=1}$ should be $\max\{d(p, O) \mid p \in \mathrm{Area}_{\mathbb{H}} \wedge p \in \mathrm{Area}_{\mathbb{C}}\}$. Because the proving of Lemma 3 is similar to that of Theorem 3, we omit the detail proof.

7 Performance analysis

7.1 Comparison

This section compares the proposed protocols with related studies including protocol RB [14], protocol DB [24], protocol SP [27] and protocol SPreg [30].

Table 2 shows the comparison results, where "✓" means satisfied totally, "×" means dissatisfied and "–" means uninvolved. Obviously, protocol RB cannot resist colluding attacks. Protocol DB only verifies the upper bound but not a target area. Further, protocol DB should deploy pre-shared key. Protocol SP and protocol SPreg can only realize location verification but not support batch verification. Except protocol SPreg, the aforementioned protocols are not analyzed in the UC framework. Compared to related studies, the proposed protocols can satisfy all the required properties.

Table 3 compares the communication and computation overhead, where $|X|$ denotes the length of message X with high min-entropy, n denotes the length of random strings, F denotes one operation of BSM PRG, and g denotes one operation of MAC. Obviously, the communication overhead of a verifier in

Table 3 Communication and computation overhead

	SP		SPreg		Ours	
	Verifier	Prover	Verifier	Prover	Verifier	Prover
Communication overhead	$\|X\|+\|n\|$	$\|n\|$	$\|X\|+2\|n\|$	$2\|n\|$	$2\|X\|+\|n\|$	$\|n\|$
Computation overhead	$3F$	$3F$	$3F+g$	$6F+3g$	$6F+g$	$6F+g$

the proposed protocols is $2|X|+n$, which is slight worse than that of SP and SPreg. The communication overhead of a prover in the proposed protocols is n, which is better than that of SPreg. The computation overhead of a verifier and the computation overhead of a prover in the proposed protocols are same, i.e. $6F+g$. Compared to SPreg, the proposed protocols have better performance in computation overhead. Therefore, our protocols can be applied for secure area verification on large-scale unmanned devices.

7.2 False accept and false reject

From the security analysis, $\mathrm{Area}(O,R) \subset \mathrm{Area}_{\mathbb{H}}$ in protocol $\mathrm{CAV}_{\delta=1}$. Thus, there are some false accept positions which are outside $\mathrm{Area}(O,R)$ but in $\mathrm{Area}_{\mathbb{H}}$. In this section, we present the error analysis of the proposed protocols. In fact, given radius R ($R \ll \min\{d(v_i,v_j)|i \neq j\}$), $\mathrm{Area}_{\mathbb{H}}$ is determined by the followed two factors: (1) The position of center O in the triangle enclosed by verifiers $\mathbb{V}$. (2) The value of δ. In order to get a lower false accept ratio, an available solution to reduce the area of $\mathrm{Area}_{\mathbb{H}}$ is to set $\delta < 1$. While the false reject event, i.e., a position within $\mathrm{Area}(O,R)$ cannot pass the circular area verification on $\mathrm{Area}(O,R)$, may occur when $\delta < 1$ and $\mathrm{Area}(O,R) \nsubseteq \mathrm{Area}_{\mathbb{H}}$.

Let FA and FR denote the false accept area and the false reject area respectively. We define the false accept ratio FAR and the false reject ratio FRR as FAR = FA/ $\mathrm{Area}(O,R)$ and FRR = FR/$\mathrm{Area}(O,R)$. Thus, the total false ratio TFR should be FAR + FRR.

Note that the false reject event violates the correctness of area verification, while the false accept event also deviates from the original intention of security requirement. Therefore, we first analyze the false accept ration FAR, and then show the relationship between FAR and FRR when $\delta < 1$.

7.3 FAR in $\mathrm{CAV}_{\delta=1}$

Our experiments are implemented on a PC (CPU: Intel Core i5-7200U 2.5 GHz, RAM: 8 G, OS: Windows 10) using Matlab R2014b. For simplicity, we set the coordinates for verifiers as $v_1 = (0, 40/\sqrt{3})$, $v_2 = (-20, -20/\sqrt{3})$, $v_3 = (20, -20/\sqrt{3})$, i.e., verifiers $\mathbb{V} = \{V_1, V_2, V_3\}$ can form an equilateral triangle. We divide the triangle region into grids equally and calculate the values of FAR when center O is located at the intersection points of these grids.

Figure 5 indicates that the values of FAR are different under different center O. Particularly, FAR can achieve the minimum value 0.4718 when center O is located at $P_0 = (0,0)$, i.e., the center of the equilateral triangle. Obviously, FAR increases when center O moves away the center of the triangle. For example, FAR will be 0.4929, 0.6064, and 0.7157 when center O is located at $P_1(0,-3)$, $P_2(0,-7)$, and $P_3 = (0,-9)$, respectively.

7.4 False ratio under different δ

We set center O as $(0,0)$ simply. Then, we select different δ and evaluate the false ratio of CAV_δ including FAR, FRR and TFR. Figure 6 shows that FAR increases with the increase of δ, while FRR decreases oppositely. In this experiment, FAR is 0 when $\delta \leqslant 0.73$; FRR is 0 when $\delta \geqslant 0.9$. If TFR = FAR+ FRR, the minimum value of TFR = FAR+ FRR is 0.0912 when $\delta = 0.8$ and TFR is near 0.092 when $0.8 \leqslant \delta \leqslant 0.85$. Note that the X-axis in Figure 6 is not uniform in order to illustrate the details.

Note that TFR can be reconsidered as μFAR$+\nu$FRR based on coefficients μ and ν, where $0 \leqslant \mu, \nu \leqslant 1$ in different scenarios. For example, a false reject event in a fire alarm system will be fatal, but a false accept may be tolerant. Thus, we can set $\mu < \nu$ in this scenario.

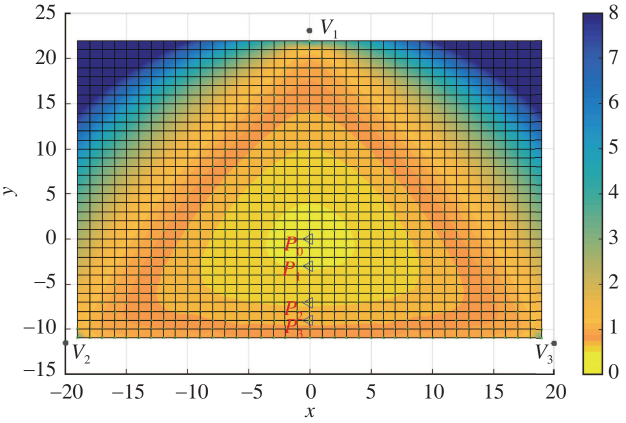

Figure 5 (Color online) FAR under different O in protocol $\mathrm{CAV}_{\delta=1}$.

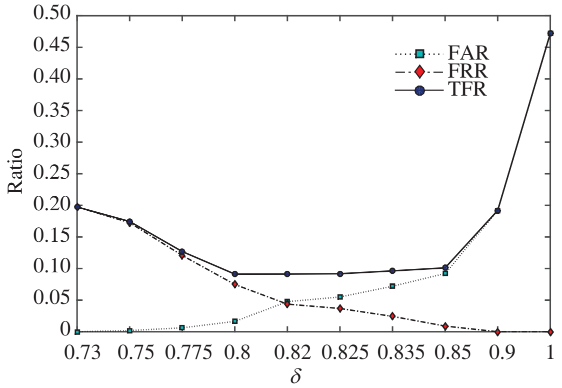

Figure 6 (Color online) FAR and FRR under different δ.

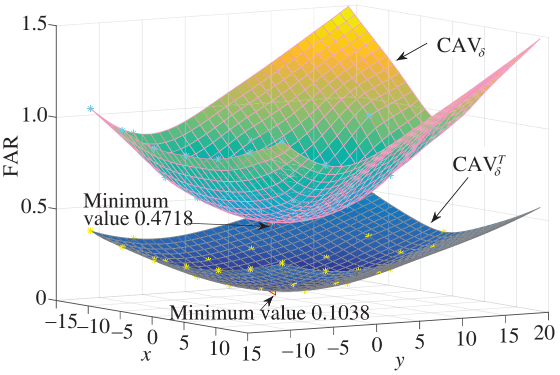

Figure 7 (Color online) FAR in $\mathrm{CAV}_{\delta=1}$ and $\mathrm{CAV}^{T}_{\delta=1}$.

7.5 $\mathrm{CAV}_{\delta=1}$ vs. $\mathrm{CAV}^{T}_{\delta=1}$ in FAR

Figure 7 focuses on FAR in both $\mathrm{CAV}_{\delta=1}$ and $\mathrm{CAV}^{T}_{\delta=1}$, where FAR is coordinate axis Z. It is obvious that the value of FAR in $\mathrm{CAV}^{T}_{\delta=1}$ is much less than that of $\mathrm{CAV}_{\delta=1}$. For instance, when center O is (0, 0), the minimum value of FAR in $\mathrm{CAV}^{T}_{\delta=1}$ is 0.1038 which is less than 0.4718, i.e., the minimum value of FAR in $\mathrm{CAV}_{\delta=1}$.

8 Conclusion

In this study, we investigate the composition security of geographic area verification in the UC framework. We formalize the ideal functionality of geographic area verification and propose two secure geographic area verification protocols, i.e., CAV_{δ} and $\mathrm{CAV}^{T}_{\delta}$. Compared to the existing schemes, the proposed protocols not only realize geographic area verification without any pre-shared secret, but also support batch verification on large-scale smart devices. The proposed protocols satisfy the completeness and the composition security in the UC framework.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61472310, U1536202, 61672413, 61672415, 61601107, U1708262) and China 111 Project (Grant No. B16037).

References

1 Yang G, Zhou X S. Intelligent CPS: features and challenges. Sci China Inf Sci, 2016, 59: 050102
2 Chen J, Zhang F, Sun J. Analysis of security in cyber-physical systems. Sci China Technol Sci, 2017, 60: 1975–1977
3 Ji X S, Huang K Z, Jin L, et al. Overview of 5G security technology. Sci China Inf Sci, 2018, 61: 081301

4 Li B, Wang W J, Yin Q Y, et al. An energy-efficient geographic routing based on cooperative transmission in wireless sensor networks. Sci China Inf Sci, 2013, 56: 072302

5 Kwon T, Lee J H, Song J S. Location-based pairwise key predistribution for wireless sensor networks. IEEE Trans Wirel Commun, 2009, 8: 5436–5442

6 Zhang Y C, Liu W, Fang Y G, et al. Secure localization and authentication in ultra-wideband sensor networks. IEEE J Sel Areas Commun, 2006, 24: 829–835

7 Sastry N, Shankar U, Wagner D. Secure verification of location claims. In: Proceedings of the 2nd ACM Workshop on Wireless Security, 2003. 1–10

8 He D B, Zeadally S, Wu L B. Certificateless public auditing scheme for cloud-assisted wireless body area networks. IEEE Syst J, 2018, 12: 64–73

9 Shen J, Shen J, Chen X F, et al. An efficient public auditing protocol with novel dynamic structure for cloud data. IEEE Trans Inform Forensic Secur, 2017, 12: 2402–2415

10 Wang D, Cheng H B, Wang P, et al. Zipf's law in passwords. IEEE Trans Inform Forensic Secur, 2017, 12: 2776–2791

11 Wang D, Wang P. Two birds with one stone: two-factor authentication with security beyond conventional bound. IEEE Trans Depend Secure Comput, 2018, 15: 708–722

12 Shen J, Zhou T Q, Chen X F, et al. Anonymous and traceable group data sharing in cloud computing. IEEE Trans Inform Forensic Secur, 2018, 13: 912–925

13 He D B, Zeadally S, Kumar N, et al. Anonymous authentication for wireless body area networks with provable security. IEEE Syst J, 2017, 11: 2590–2601

14 Vora A, Nesterenko M. Secure location verification using radio broadcast. IEEE Trans Depend Secure Comput, 2006, 3: 377–385

15 Du W L, Fang L, Ningi P. LAD: localization anomaly detection for wireless sensor networks. In: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, 2005. 874–886

16 Capkun S, Cagalj M, Srivastava M. Secure localization with hidden and mobile base stations. In: Proceedings of IEEE INFOCOM, 2006. 1–10

17 Chiang J T, Haas J J, Hu Y C. Secure and precise location verification using distance bounding and simultaneous multilateration. In: Proceedings of the 2nd ACM Conference on Wireless Network Security, 2009. 181–192

18 Hasan R, Khan R, Zawoad S, et al. WORAL: a witness oriented secure location provenance framework for mobile devices. IEEE Trans Emerg Top Comput, 2016, 4: 128–141

19 Perazzo P, Sorbelli F B, Conti M, et al. Drone path planning for secure positioning and secure position verification. IEEE Trans Mobile Comput, 2017, 16: 2478–2493

20 Sciancalepore S, Oligeri G, Di P R. Shooting to the stars: secure location verification via meteor burst communications. In: Proceedings of IEEE Conference on Communications and Network Security, 2018. 1–9

21 Brands S, Chaum D. Distance-bounding protocols. In: Advances in Cryptology-EUROCRYPT. Berlin: Springer, 1993. 344–359

22 Rasmussen K B, Capkun S. Location privacy of distance bounding protocols. In: Proceedings of the 15th ACM Conference on Computer and Communications Security, 2008. 149–160

23 Tippenhauer N O, Capkun S. Id-based secure distance bounding and localization. In: Proceedings of Computer Security-ESORICS, 2009. 621–636

24 Capkun S, El D K, Tsudik G. Group distance bounding protocols. In: Proceedings of International Conference on Trust and Trustworthy Computing, 2012. 302–312

25 Cremers C, Rasmussen K B, Schmidt B, et al. Distance hijacking attacks on distance bounding protocols. In: Proceedings of IEEE Symposium on Security and Privacy, San Francisco, 2012. 113–127

26 Perazzo P, Dini G. Secure positioning with non-ideal distance bounding protocols. In: Proceedings of IEEE Symposium on Computers and Communication (ISCC), Larnaca, 2015. 907–912

27 Chandran N, Goyal V, Moriarty R, et al. Position based cryptography. In: Advances in Cryptology-CRYPTO. Berlin: Springer, 2009. 391–407

28 Buhrman H, Chandran N, Fehr S, et al. Position-based quantum cryptography: impossibility and constructions. In: Proceedings of the 31st Annual Conference on Advances in Cryptology, Santa Barbara, 2011. 429–446

29 Yang R P, Xu Q L, Au M H, et al. Position based cryptography with location privacy: a step for fog computing. Future Gener Comput Syst, 2018, 78: 799–806

30 Zhang J W, Ma J F, Yang C, et al. Universally composable secure positioning in the bounded retrieval model. Sci China Inf Sci, 2015, 58: 110105

31 Canetti R. Universally composable security: a new paradigm for cryptographic protocols. In: Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science, 2001. 136–145

32 Datta A, Derek A, Mitchell J C, et al. A derivation system and compositional logic for security protocols. J Comput Sec, 2005, 13: 423–482

33 Zhang J W, Ma J F, Moon S J. Universally composable one-time signature and broadcast authentication. Sci China Inf Sci, 2010, 53: 567–580

34 Hu X X, Zhang J, Zhang Z F, et al. Universally composable anonymous password authenticated key exchange. Sci China Inf Sci, 2017, 60: 52107

35 Zhang J W, Ma J F, Moon S J. Universally composable secure TNC model and EAP-TNC protocol in IF-T. Sci China Inf Sci, 2010, 53: 465–482

36 Zhang J W, Ma J F, Yang C. Protocol derivation system for the Needham-Schroeder family. Sec Commun Netw, 2015, 8: 2687–2703

37 He C H, Sundararajan M, Datta A, et al. A modular correctness proof of ieee 802.11i and TLS. In: Proceedings of the 12th ACM Conference on Computer and Communications Security, 2005. 2–15

38 Naszódi M. On some covering problems in geometry. In: Proceedings of the American Mathematical Society, 2016. 3555–3562