Automatika

Journal for Control, Measurement, Electronics, Computing and Communications

ISSN: 0005-1144 (Print) 1848-3380 (Online) Journal homepage: https://www.tandfonline.com/loi/taut20

A mutual authentication and key update protocol in satellite communication network

Congyu Huang, Zijian Zhang, Meng Li, Liehuang Zhu, Zhengjia Zhu & Xiaoxian Yang

To cite this article: Congyu Huang, Zijian Zhang, Meng Li, Liehuang Zhu, Zhengjia Zhu & Xiaoxian Yang (2020) A mutual authentication and key update protocol in satellite communication network, Automatika, 61:3, 334-344, DOI: 10.1080/00051144.2020.1757966

To link to this article: https://doi.org/10.1080/00051144.2020.1757966

Published online: 04 May 2020.

AUTOMATIKA
2020, VOL. 61, NO. 3, 334–344
https://doi.org/10.1080/00051144.2020.1757966

REGULAR PAPER

OPEN ACCESS

A mutual authentication and key update protocol in satellite communication network

Congyu Huang[a], Zijian Zhang[a,b], Meng Li[c], Liehuang Zhu[a], Zhengjia Zhu[a] and Xiaoxian Yang[d]

[a]School of Computer Science, Beijing Institute of Technology, Beijing, People's Republic of China; [b]University of Auckland, Auckland, New Zealand; [c]School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, People's Republic of China; [d]School of Computer and Information Engineering, Shanghai Polytechnic University, Shanghai, People's Republic of China

ABSTRACT

Satellite communication networks have been widely used to provide essential communication services, including voice communication, global positioning, message communication, etc. However, sorts of network attacks are easy to be launched in these networks due to the limited computation capability and communication width, long communication delay, and intermittent link connection. In this paper, we first propose a new [E]ncryption-based [M]utual [A]uthentication and [K]ey [U]pdate (EMAKU) protocol in satellite communication networks. Next we analyze the security of the EMAKU protocol under two classic network attacks which are replay attack and man-in-the-middle attack. Finally, experiments show that the EMAKU protocol is 21.5% faster than the traditional encryption-based authentication protocols, and the average time of key update of the EMAKU protocol is about 450.01 ms.

ARTICLE HISTORY
Received 8 September 2019
Accepted 29 March 2020

KEYWORDS
Satellite communication networks; authentication; key update

1. Introduction

With the increasing development of communication technology, satellite communication systems are becoming more and more prevalent [1–7]. They can provide a variety of essential communication services, including voice communication, global positioning, and message communication [8–11]. These systems usually have to face serious network attacks, such as replay attack or man-in-the-middle attack, because the computation capability and communication width are limited [12], the communication delay is long [13], and the link connection is intermittent [14]. Therefore, a secure satellite communication network is difficult to be built for satellite communication systems.

The essential method to guarantee the security of satellite communication networks is to authenticate each new satellite when it launches into the network and exchange a key among the satellites and the on-ground base stations. Several authentication and key agreement schemes have been proposed to provide security assurance in satellite communication networks. For instances, Wullems et al. [15] proposed a public key cryptosystem-based authentication protocol to improve the security of satellite systems. However, the protocol was unidirectional, so it cannot meet the requirement of mutual authentication. Cruickshank et al. [16] designed a mutual authentication protocol between endpoints and satellites. But the designed protocol had a high maintenance cost and a high failure risk. Sasaki et al. [17] put forward a double-layered inclined orbit constellation to improve the robustness of satellite communication network. But they did not consider the security for the network. Zhang et al. [18] proposed a low-earth orbit satellite and group key agreement protocol based 3GPP authentication and key agreement protocol. But they did not consider key update cases. Zhu et al. [19] proposed an entity authentication and access control scheme in satellite communication networks, but the protocol is not suitable for authentication among satellites.

The main contributions of this paper are summarized as follows:

- This paper first proposes a new [E]ncryption-based [M]utual [A]uthentication and [K]ey [U]pdate (EMAKU) protocol for double-layered satellite communication networks. The protocol considers the limited computation and communication resources of satellites. Meanwhile, it applies geostationary-earth-orbit (GEO) satellites to control the clusters of low-earth-orbit (LEO) satellites, such that the key update process of LEO satellites can always be controlled by the on-ground base stations.
- We take the replay attack and man-in-the-middle attack as examples to demonstrate the security of the EMAKU protocol.
- A simulation platform is implemented, and simulation experiments show that the efficiency of the

CONTACT Xiaoxian Yang ✉ xxyang@sspu.edu.cn

proposed protocol is 21.5% faster than the traditional encryption-based authentication protocols, while the average time of key update is about 450.01 ms.

The rest of the paper is organized as follows. In Section 2, we mainly discuss the related works of authentication protocols and architectures in satellite communication networks. In Section 3, we describe the preliminaries. In Section 4, we discuss the models and goals of this paper. In Section 5, we describe the mutual authentication and key update protocol. In Section 6, we analyse the security and performance of the proposed protocol. In the last section, we summarize the paper.

2. Related works

There are various of authentication and key exchange protocols designed for authenticating entities in wireless communication networks. For instance, Lu et al. [20] proposed an authentication and key agreement protocol based on 3GPP authentication and key agreement protocol. But it is not suitable to use in satellite networks due to its huge resources requirement. Zeng et al. [21] also proposed an efficient anonymous user authentication protocol for mobile Internet of things. However, it took too much computation cost if it was directly used in satellite communication networks. Lin et al. [22] proposed an efficient dynamic authentication protocol. It reduced space storage and key management complexity without using verification table. But the computation cost of the protocol is too heavy to be deployed in satellites with limited computation resources.

There are also several authentication protocols designed for authenticating entities in satellite communication networks. For example, Chang et al. [23] proposed an authentication and key agreement protocol in the satellite communication networks. This protocol aimed to authenticate between endpoints and satellites. Unfortunately, it is difficult to be practical for mutual authentication among satellites. Lee et al. [24] presented an entity authentication protocol which made use of static and dynamic identities in a verification table to lower computation cost. However, the proposed protocol was not secure when the verification table is leaked. Zhibo et al. [25] put forward an end-to-end authentication protocol in the satellite communication networks. This protocol was proposed on the Internet key exchange (IKE) protocol. However, the computation cost of the proposed protocol was heavier than that of the authentication protocols based on private key cryptography, since the fundamental IKE protocol applied public key cryptography.

In summary, the existing works cannot meet all the requirements of security, efficiency, and limited computation and storage cost for mutual authentication and key update for satellites communication and satellite-endpoint communication, simultaneously.

3. Preliminaries

In this paper, we modify a reliable maintenance protocol proposed in [26] to update secret encryption and integrity keys between Ground Control Center (GCC) and satellites. Here the specification of the reliable maintenance protocol is shown in Figure 1, where the *Enc* is an encryption algorithm that can resist against chosen plain text attack, and *MAC* is a message authentication code algorithm that is secure under chosen message attack.

The reliable maintenance protocol mainly contained two steps. In the first step, mutual authentication between GCC and a satellite that neighbours to the targeted satellite required to update an encryption key *CK* and an integrity key *IK*. In the second step, the GCC passes new keys to the targeted satellite via two secure communication channels which are (1) between GCC and the neighbouring satellite, and (2) between the neighbouring satellite and the targeted satellite. Our key update protocol is based on the reliable maintenance protocol.

4. Models and goals

4.1. System model

Figure 2 depicts the system model of satellite communication networks. It consists of User Terminals(UT), GCC, GEO satellites and LEO satellites. Since LEO satellite networks cannot keep connection with GCC all the time, and parts of GEO satellites are out of the communication range with GCC, it is of great importance to build a secure satellites–satellites communication channel by which GCC can communicate with every GEO and LEO satellite. Here, each pair of neighbouring GEO/LEO satellites is assumed to have a communication channel. Specifically, each GEO satellite can communicate with LEO satellites when the LEO satellites run into the communication range.

- *GEO satellites*. A GEO satellite GV is regarded as a 3-tuple $< n^{GV}, s^{GV}, c^{GV} >$, where n^{GV} is the number of GEO satellites, s^{GV} stands for the security parameter, and c^{GV} represents the control information for GEO satellites.
- *LEO satellites*. LV represents a LEO satellite, which can be denoted by a 3-tuple $< n^{LV}, s^{LV}, c^{LV} >$. Here n^{LV} is the number of LEO satellites, s^{LV} stands for the security parameter, and c^{LV} represents the control information for LEO satellites.
- *GCC*. GCC mainly contain an identity management module, a control module and a security module.

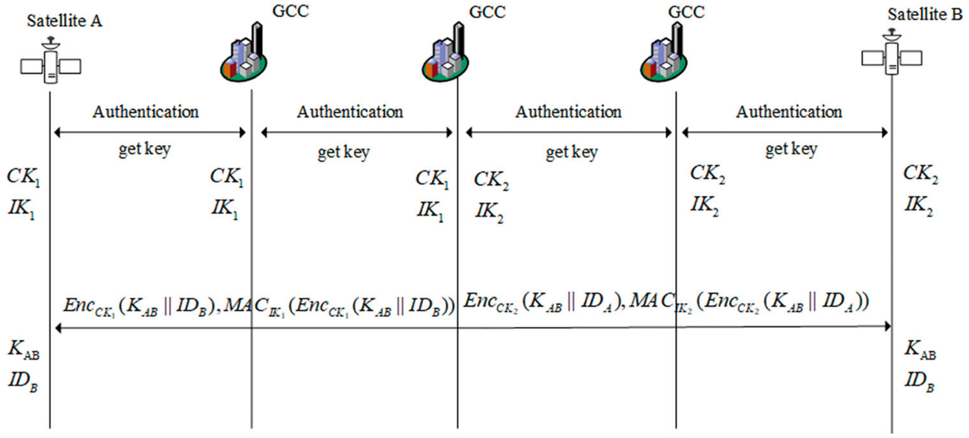

Figure 1. Reliable maintenance.

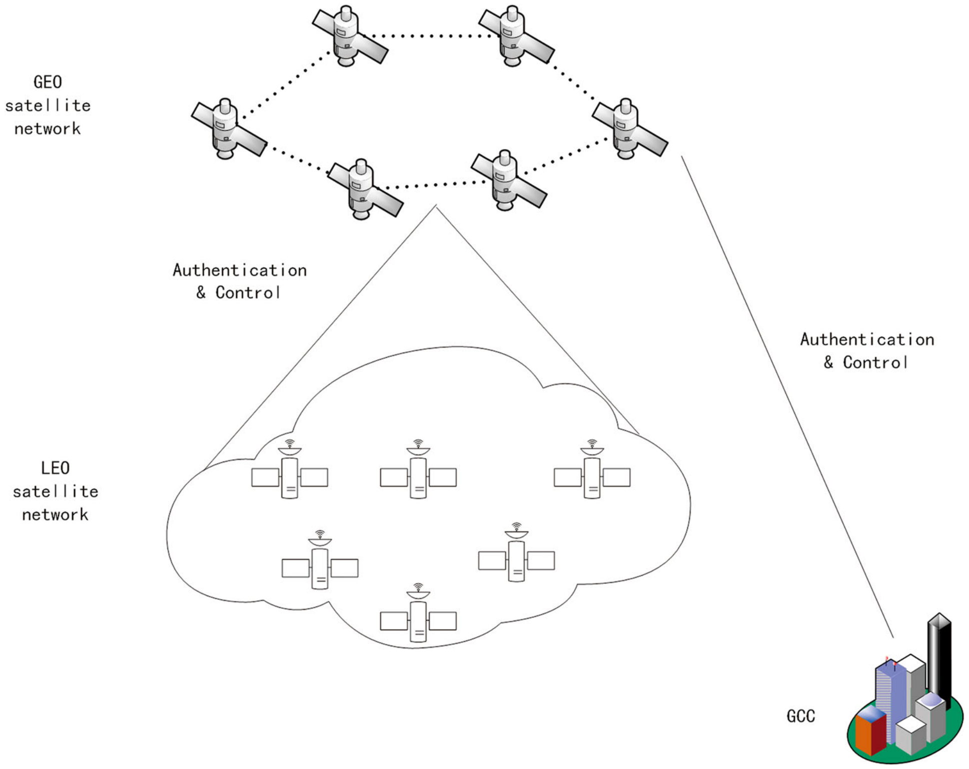

Figure 2. System model of satellite communication networks.

GCC can assign pre-shared keys, authenticate and manage GEO/LEO satellites.

- *GEO satellite networks*. *GSN* denotes the GEO satellite networks, which consist of GEO-GEO satellite communication channels. The GEO satellite networks can be regarded as an attribute graph $GSN = (GV, GE)$, where GV denotes a GEO satellite as a vertex in graph, and GE denotes a GEO-GEO satellite communication channel as an edge in the graph.
- *LEO satellite networks*. *LSN* denotes the LEO satellite networks, which contain LEO-LEO satellites communication channels. The LEO satellite can be regarded as an attribute graph $LSN = (LV, LE)$, where LV denotes a LEO satellite as a vertex in graph, and LE denotes an LEO-LEO satellite communication channel as an edge in the graph.
- *Communication*. *Send* denotes message that is delivered from an entity to other one.
- *Authentication*. *Auth* denotes the authentication protocol between two entities in satellite communication networks. That is, an authentication protocol between two GV-GV, or two LV-LV, or GV-LV, or GCC-GV.

In this paper, the proposed protocol can be divided into two parts: mutual authentication and key update. The mutual authentication is among satellites, and between satellite and GCC. The key update is accomplished by three components of the GCC, GEO satellite and LEO satellite. The procedure is start with the GCC. More concretely, the GCC *Send* messages to the GEO satellite, and then the GEO satellite *Send* messages to the LEO satellite.

4.2. Threat model

The threat is presumed to have the ability of launching active attacks such as replay attack or man-in-the-middle attack, etc. Specifically, since the communication channels of the satellite communication networks are wireless, all the messages received can be regarded as generating or forwarding by adversaries theoretically. In other words, messages occur on any satellite communication channels can be assumed to be intercepted or replaced by adversary.

4.3. Goal and challenge

Our goal is to build three secure satellite communication channels (1) between UT and GCC, (2) between two GEO/LEO satellites, and (3) update *CK* and *IK* between GCC and a GEO/LEO satellite, in the satellite communication networks defined in the system model under various attacks defined in the threat model.

There are three challenges to attain our goal in satellite communication networks. First, the computational and bandwidth resources of satellite communication networks are limited. Second, each satellite communication channel is public and vulnerable to be attacked. Third, the topology of *LSN* is not stable from the viewpoint of GCC.

5. An encryption-based mutual authentication and key update protocol in satellite communication networks

In this section, we first propose two mutual authentication sub-protocols to establish secure communication channels (1) among satellites and (2) between satellite and GCC. Next, we propose a key update sub-protocol for updating the *CK* and *IK* for LEO satellites.

5.1. Mutual authentication between a GEO/LEO satellite and the GCC

GEO/LEO satellites utilize the symmetric keys, which are used for authenticating *Auth*. Specifically, the former satellite executes the symmetry K_{G_i} for itself, and the symmetric key $K_{G_{ij}}$ utilized for authentication between satellites is presented by the GCC. The symmetric key is sent by the original satellite first, when the satellite in orbit received the key, it executes the authentication process through the key $K_{G_{ij}}$. The protocol specification is shown in Figure 3, and the process is depicted in algorithm 1. A detailed description is shown as follows.

Algorithm 1 Satellite and GCC networking authentication

```
Require: $ID_{G_i}, ID_G$
Ensure: authentication result
$AuthMessage \leftarrow MSG$
$Send\ \ AuthMessage\ \ to\ \ ID_{G_i}$
$ID_{G_i}\ \ Compute\ \ r_G$
$AuthMessage \leftarrow MSG$
$Send\ \ AuthMessage\ \ to\ \ ID_G$
$ID_G\ \ Compute\ \ r_G\ \ and\ \ XMAC$
if $MAC == XMAC$ then
    $AuthMessage \leftarrow MSG$
    $Send\ \ AuthMessage\ \ to\ \ ID_{G_i}$
    $generates\ \ sk||CK||IK$
else
    return $fail$
$ID_{G_i}\ \ Compute\ \ XMAC$
if $MAC == XMAC$ then
    $generates\ sk||CK||IK$
    return $pass$
else
    return $fail$
```

(1) Firstly, ID_G generates authentication message which contain *MSG*. Then *Send* the message to ID_{G_i}, ID is the identity of entity of the satellite communication network.

(2) When ID_{G_i} receives the message, ID_{G_i} computes r_A, and then saves the value of r_G. Next, ID_{G_i} generates its own authentication message which contains $MSG_1||MSG_2$, and sends the message to ID_G.

(3) Once ID_G receives the response, ID_G computes rG_i. And then it checks whether XID_G is correct. If the message content is correct, then generates the second message which contains *MSG*. At the same time, ID_G generates $sk||CK||IK$. If the message is not correct, the protocol aborts.

(4) Once ID_{G_i} receives the response, ID_{G_i} computes *XMAC*, and the protocol will continue to check *XMAC*. If its content is correct, ID_{G_i} generates a message $sk||CK||IK$ that is used in the future communication.

5.2. Mutual authentication between two GEO/LEO satellites

All the satellites are assumed to be launched one by one and gradually build a satellite communication network. Thus, the authentication of different satellites is not exact the same. When the first satellite is launched, the network has not been built yet. The authentication for the first satellite is authenticated through the proposed

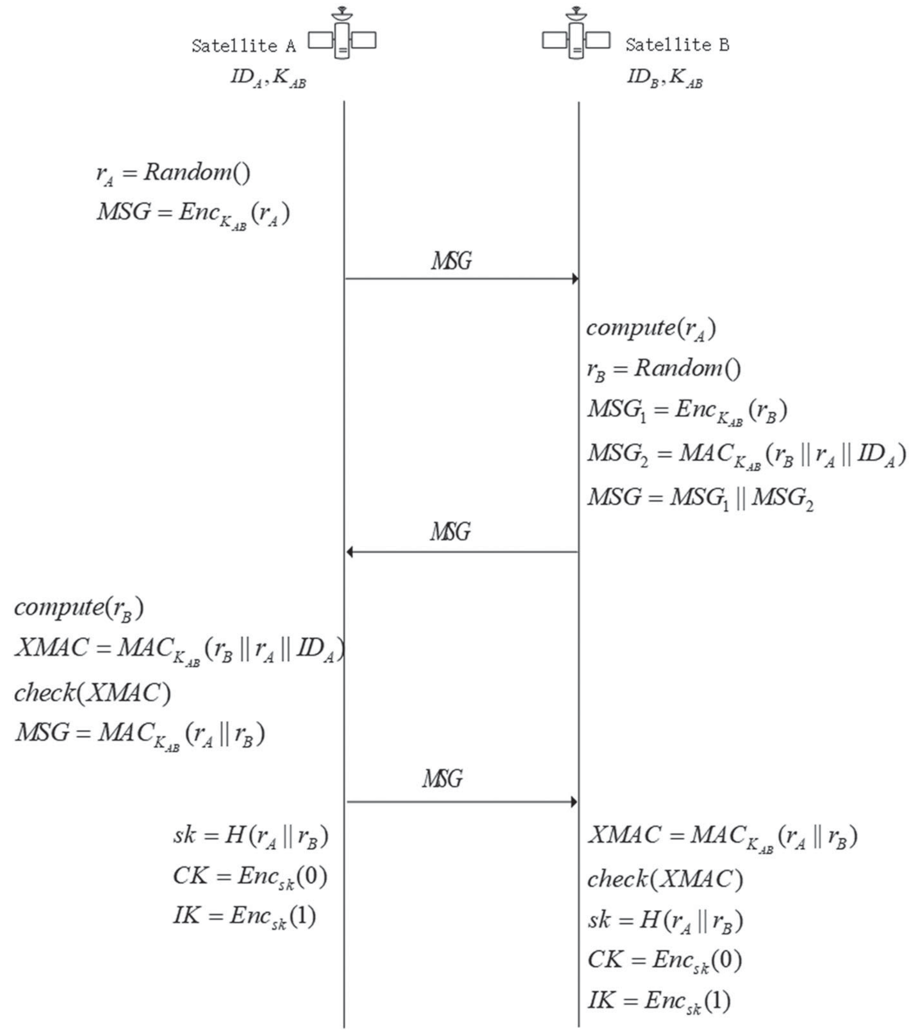

Figure 3. Authentication protocol between a GEO/LEO Satellite and the GCC.

mutual authentication protocol between a GEO/LEO Satellite and the GCC. Upon completing the authentication of the first satellite, the second satellite can be deployed in a similar way.

Besides mutual authentication between a GEO/LEO Satellite and the GCC, it is also vital to build a secure communication channel for neighbouring satellites, since there must exists GEO/LEO satellites that cannot directly communicate with the GCC. So these satellites can only authenticate with the GCC, when there are secure communication channel among neighbouring satellites. The steps of the mutual authentication protocol between two GEO/LEO satellites (called A and B for short) are defined as below:

(1) A first calculates the authentication massage MSG based on its own key K_{AB} to initiate a challenge, where the authentication message is made up of the following three elements which are the identity ID_A, the encrypted ciphertext of random number r_A and the MAC of random number r_A. The generation process of the message MSG is constructed as $MSG = Enc_{K_{AB}(r_A)}$. A then sends the message MSG to B.

(2) Once B receiving the MSG from A, it obtains r_A by decrypting $Enc_{K_{AB}(r_A)}$ with K_{AB}. B then generates a random number r_B, and a message $MSG_1 = Enc_{K_{AB}(r_B)}$, $MSG_2 = MAC_{K_{AB}}(r_B||r_A||ID_A)$, $MSG = MSG_1||MSG_2$. After that, the message MSG is sent to A.

(3) Upon A receives B's MSG, $MSG_1||MSG_2$ is computed by MSG. A obtains r_B by decrypting MSG_1 and computes $XMAC = MAC_{K_{AB}}(r_B||r_A||ID_A)$. Then A will check $XMAC$. While it is correct, A calculates $MSG = MAC_{K_{AB}}(r_A||r_B)$ and sends MSG to B. A computes $sk = H(r_A||r_B)$, $CK = Enc_{sk}(0)$, $IK = Enc_{sk}(1)$ for future communication.

(4) When B receives A's MSG, B computes $XAMC = MAC_{K_{AB}}(r_A||r_B)$. And then checks $XAMC$. If all of the parameters above are correct, B computes $sk = H(r_A||r_B)$, $CK = Enc_{sk}(0)$, $IK = Enc_{sk}(1)$.

5.3. Key update for LEO satellites

The key update is designed for LEO satellites. In the GEO/LEO satellite communication networks, GEO satellites can be used to control some LEO satellites when those satellites are compromised by an adversary. The key update protocol contains two steps. First, mutual authentication between GEO and LEO satellites. Second, using the GEO satellites to update the key of the compromised LEO satellites.

After establishing *GSN* and *LSN*, the key update for LEO satellites can be implemented with the help of *GSN*. Specifically, if the GCC wants to update key for a LEO satellite which is out of the communication range, it can use some GEO satellites as bridge. The protocol specification is shown in Figure 4.

The following example illustrates the failure of a high-orbiting satellite to describe the process to update keys and re-build a secure communication channel is shown in Figure 5 and the next two procedures.

Algorithm 2 Key Update Procedure 1

```
Require: $K_{AB}$
Ensure: update state
Satellite and Ground Authenticate
get $CK_1, IK_1, CK_2, IK_2$
encrypt $K_{AB}$
Send $K_{AB}$ to A and B
Send $ID_A$ to B and $ID_B$ to A
if Success then
    return Success
else
    return Fail
```

Algorithm 3 Key Update Procedure 2

```
Require: $ID_A, ID_B$
Ensure:
$CK = f2_{K_{AB}}(RAND_1)$    $IK = f3_{K_{AB}}(RAND_1)$
$MSG_1 = Enc_{CK_{AB}}(SQN||ID_A)$    $MSG_2 = MAC_{IK_{AB}}(SQN||ID_A)$
$AV = MSG_1||MSG_2||RAND_1$
Send AV to B
$CK = f2_{K_{AB}}(RAND_1)$    $IK = f3_{K_{AB}}(RAND_1)$
check SQN and ID
$MSG_1 = Enc_{CK_{AB}}(SQN + 1||ID_A)$    $MSG_2 = MAC_{IK_{AB}}(SQN + 1||ID_A)$
$AV = MSG_1||MSG_2$
get $SQN, ID$
check SQN and ID
if Check Success then
    return Success
else
    return Fail
```

(1) Satellite A is first authenticated with GCC and builds a secure communication channel by the encryption key CK_1, and the integrity key IK_1, while Satellite B is also authenticated with GCC and builds a secure communication channel by the encryption key CK_2, and the integrity key IK_2.

(2) The GCC allocates the symmetric key K_{AB} for satellites A and B on both secure communication channels, and sends ID_A to satellites B, and ID_B to satellites A.

(3) Satellite A calculates CK_{AB} and IK_{AB}, encrypts the *MAC* values of *SQN*, ID_A, and *SQN* and ID_A, and generates the *AV* and sends it to Satellite B.

(4) When satellite B receives satellite A's message *AV*, B calculates CK_{AB} and IK_{AB} to decrypt *SQN* and ID_A, and checks whether it is equal to the previously received *SQN* value is within reasonable limits, if there is a verification that fails to decline authentication. If both are verified successfully, it encrypts $SQN + 1$ and ID_B, calculates the MAC values of $SQN + 1$ and ID_B to form the *AV* vector, and sends it to satellite A.

(5) When satellite A receives the message *AV*, it verifies *SQN* and ID_B, and checks if it is valid. If so, the satellite authentication is successfully completed. Otherwise, the authentication fails and access is denied.

The completion of the above steps will enable the satellites A and B to update their keys.

6. Security analysis and performance analysis

6.1. Security analysis

The Encryption-based Mutual Authentication and Key Update (EMAKU) protocol can accomplish mutual authentication and key update. Specifically, the EMA KU protocol is used a symmetric key encryption scheme to ensure the confidentiality of the protocol. Message authentication code is used to ensure the integrity of the protocol. Thus, attacks such as counterfeiting and forgery can be resisted. We use the random number instead of timestamps to protect against replay attacks. In the process of satellite communication, key update is run in the secure communication channel, which can effectively resist against man-in-the-middle attacks.

Moreover, the two entities in the communication channel perform mutual authentication and key update to obtain the encryption key and the integrity key, respectively. GCC will update both keys between the compromised neighbouring LEO satellites. Specifically, the EMAKU protocol uses a symmetric key generation function to derive an encryption key and an integrity key for providing the confidentiality and integrity. Through the proposed protocol above, the traditional attacks such as counterfeiting and forgery in the satellite communication networks can be resisted. Also, the EMAKU protocol uses the *SQN* to defend against replay attacks.

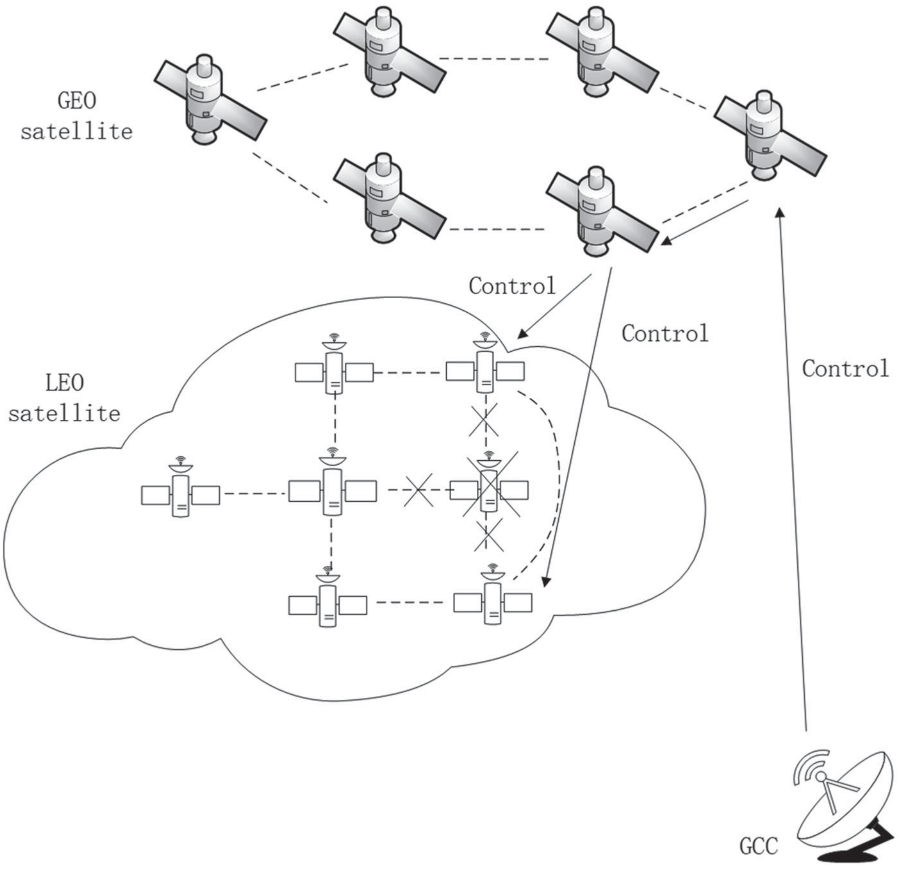

Figure 4. Control the invalid LEO satellite.

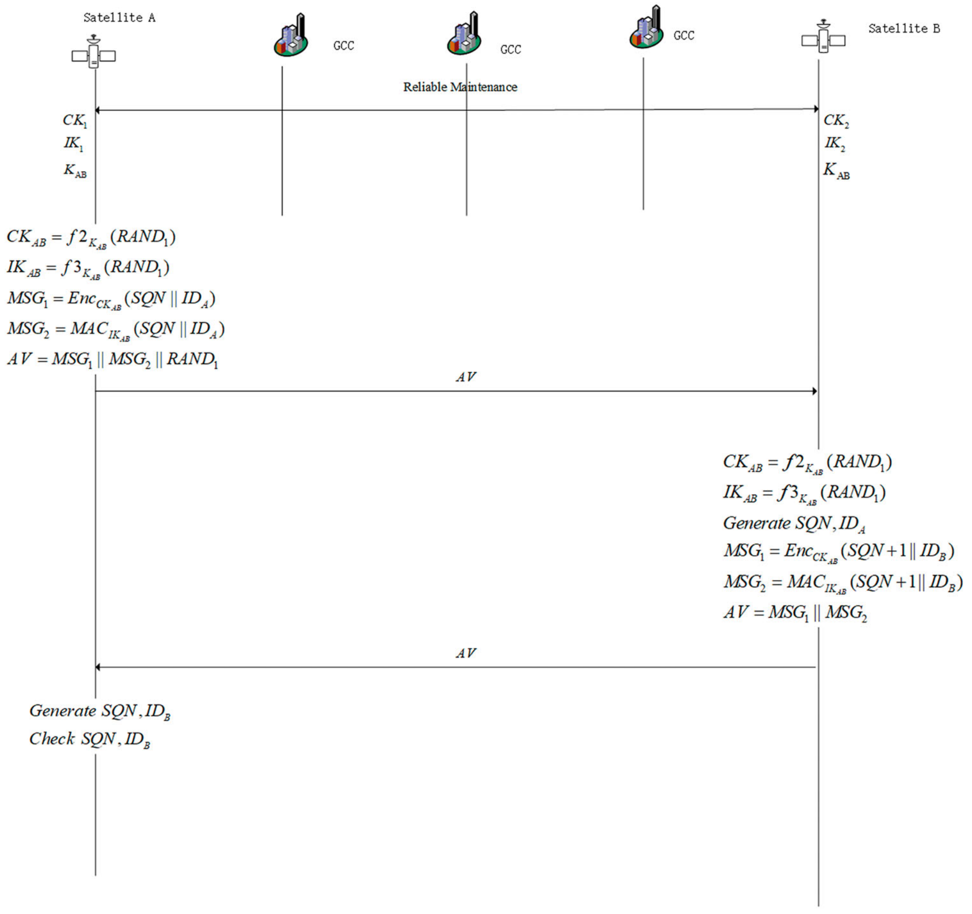
Satellite A
GCC
GCC
GCC
Satellite B
Reliable Maintenance
CK_1
IK_1
K_{AB}
CK_2
IK_2
K_{AB}
$CK_{AB} = f2_{K_{AB}}(RAND_1)$
$IK_{AB} = f3_{K_{AB}}(RAND_1)$
$MSG_1 = Enc_{CK_{AB}}(SQN \| ID_A)$
$MSG_2 = MAC_{IK_{AB}}(SQN \| ID_A)$
$AV = MSG_1 \| MSG_2 \| RAND_1$
AV
$CK_{AB} = f2_{K_{AB}}(RAND_1)$
$IK_{AB} = f3_{K_{AB}}(RAND_1)$
$Generate\ SQN, ID_A$
$MSG_1 = Enc_{CK_{AB}}(SQN+1 \| ID_B)$
$MSG_2 = MAC_{IK_{AB}}(SQN+1 \| ID_B)$
$AV = MSG_1 \| MSG_2$
AV
$Generate\ SQN, ID_B$
$Check\ SQN, ID_B$

Figure 5. Key update protocol.

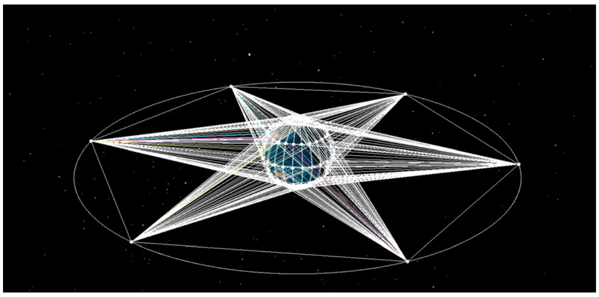

Figure 6. Simulation in STK.

6.2. Performance analysis

We simulate the EMAKU protocol under a computer which have an Intel (R) Core i7-7700HQ CPU@2.80 GHz processor to test its performance. We uses openssl open source library security algorithm in the simulation. In the experiments, we use virtual machines to simulate satellites and use Satellite Tool Kit 9.0(STK for short) [27] to calculate satellite network delay. The simulation in STK is shown in Figure 6.

In order to test the performance of the EMAKU protocol, we carried out three experiments. The first experiment is to compare the performance of our protocol with that of the traditional mutual authentication protocol. In the first experiment, we put the protocols into the satellite simulation environment to measure the communication delay and computation delay of the protocol. The second experiment is to test the performance of authentication protocols under different key lengths in the simulation network. The last experiment is to show the performance of key update in the simulation environment.

One hundred tests of network authentication were compared with the traditional mutual authentication protocol which is based on 3GPP AKA protocol [20]. The total delay results are shown in Figure 7. The total communication and computation time of the EMAKU protocol is less than the traditional mutual authentication protocol. Because the communication delay is too large and there is little difference between them, we mainly compare the computational delay between the two protocols (Figure 8).

In the first experiment, the maximum computation time of the EMAKU protocol is approximately 0.328 ms, the minimum time is about 0.053 ms, and the average computation time is about 0.073 ms. The maximum computation time of the traditional authentication scheme is approximately 0.345 ms, the minimum computation time is about 0.071 ms, and the average computation time is about 0.093 ms. The experiment shows that the average efficiency of the EMAKU protocol is 21.5% higher than that of the traditional authentication protocol.

In the second experiment, we conduct comparison between 128 bits, 192 bits, 256 bits key symmetric encryption, as shown in Figure 9. The average encryption time for 128-bits is 378.48 ms, the average time for 192-bits is 380.55 ms, the average time for 256-bits is 380.61 ms, and the fluctuation range is within 10 ms. The mutual authentication protocol is stable in the simulation environment. Because the mutual authentication protocol in this paper requires less environment, it has little impact on different key length.

In a word, the EMAKU protocol works stably in satellite communication networks.

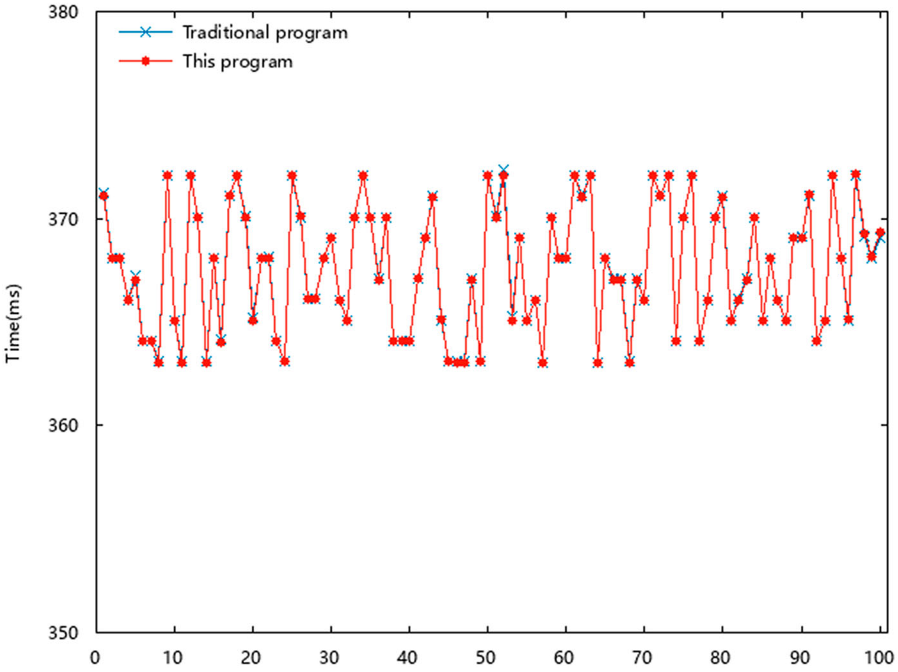

Figure 7. Total delay comparison of mutual authentication.

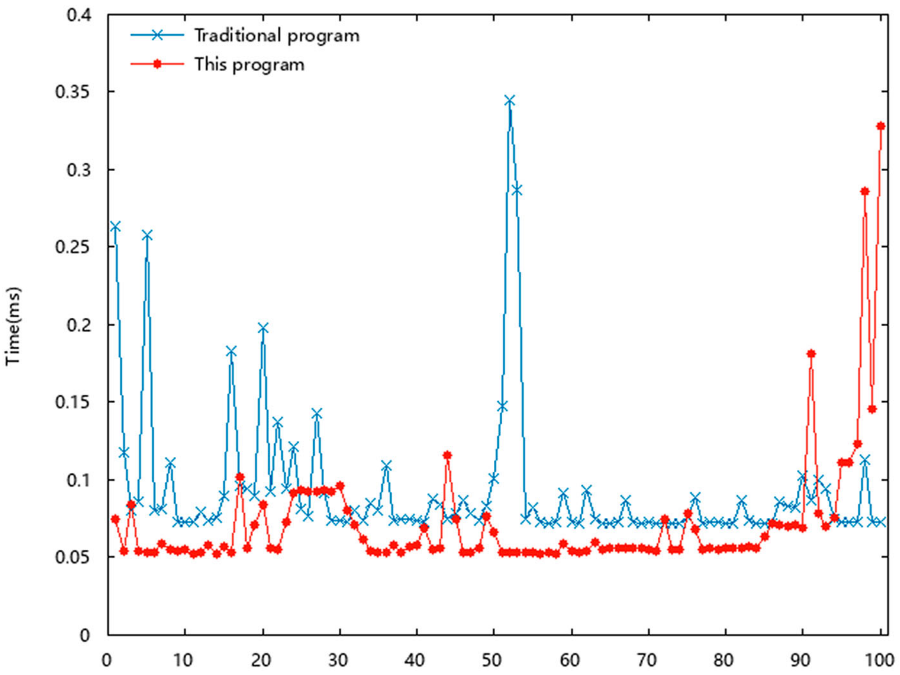

Figure 8. Computation cost comparison of mutual authentication.

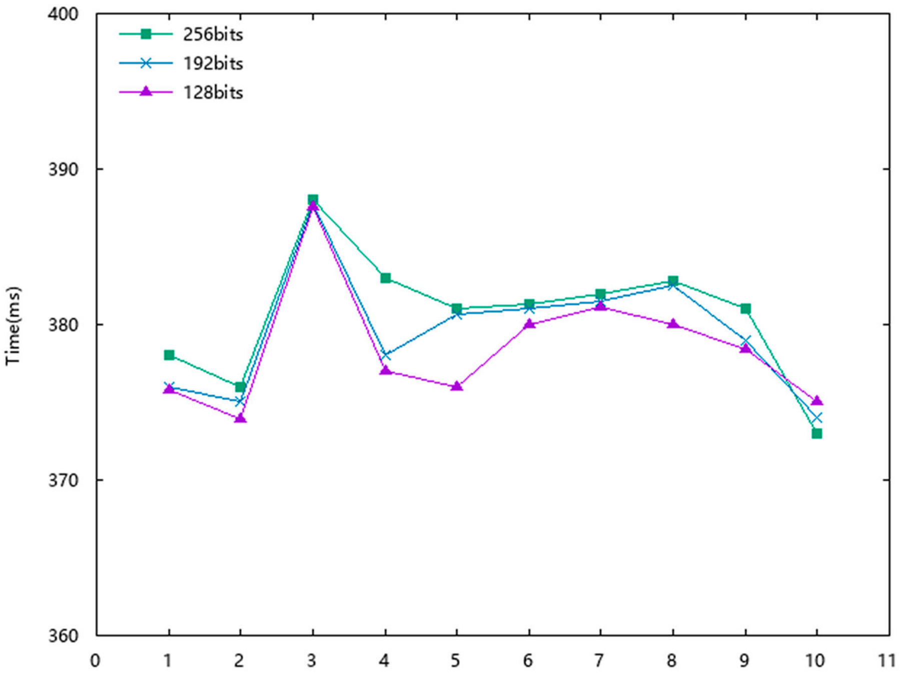

Figure 9. Performance test results of different digit number of network authentication key.

Finally, we tests the performance of key update. Since the router for key update will pass through 1–4 GEO satellites, the experiments in this paper have done 10 experiments for different paths. The test results are shown in Figure 10. As is shown in the experiments, when key update gets though 1 GEO satellite, the maximum computation time of the EMAKU protocol is 255.60 ms, the minimum time is 269.57 ms, the average time is 262.47 ms. When it turns to 2 GEO satellites, the maximum computation time of the EMAKU protocol is 392.73 ms, the minimum time is 382.10 ms, the average time is 386.03 ms. When it needs to pass 3 GEO satellites, the maximum computation time of the EMAKU protocol for key update is 519.03 ms, the minimum time is 506.77 ms, the average time is 512.32 ms. When it needs to pass 4 GEO satellites, the maximum computation time of the EMAKU protocol is 644.21 ms, the minimum time is 630.48 ms, the average time is 639.21 ms. The total average time is 450.01 ms.

The delay of key update protocol varies slightly in different paths. The large delay between different paths is due to the fact that every additional satellite passes through will have an additional time delay between high orbit satellites.

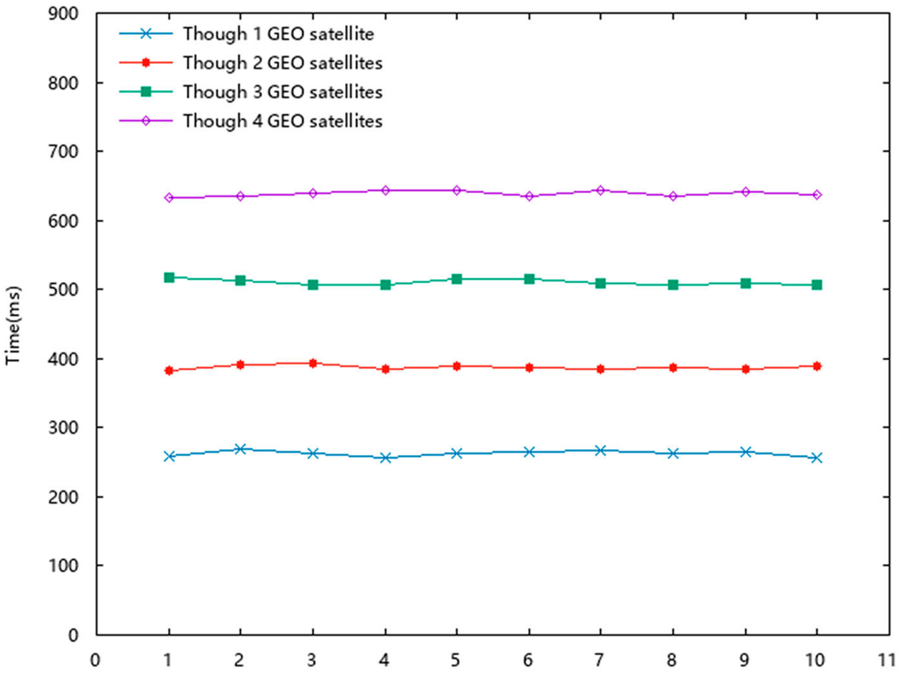

Figure 10. Test results of key update.

7. Conclusion

A new encryption-based mutual authentication and key update protocol in satellite communication networks is proposed in this paper. The security of the EMAKU protocol is proved by security analysis and the performance of the EMAKU protocol is also compared with the traditional authentication protocols. In the future, the computing power of satellite is probably more powerful, and the difficulty based on computation power will eventually be solved. As a result, the use of public key cryptography system on satellite communication networks will be a potential research direction.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This paper is supported by National Natural Science Foundation of China [Grant Number 61872041] and National Key R&D Program of China [Grant Number 2016YFB0800301].

References

[1] Yin Y, Chen L, Lu Y, et al. QoS prediction for service recommendation with deep feature learning in edge computing environment. Mobile Networks Appl. 2019:1–11. doi:10.1007/s11036-019-01241-7

[2] Gao H, Huang W, Yang X, et al. Towards service selection for workflow reconfiguration: an interface-based computing. Future Gen Comput Syst (FGCS). 2018;87:298–311.

[3] Gao H, Chu D, Duan Y, et al. The probabilistic model checking based service selection method for business process modeling. J Software Eng Knowl Eng. 2017;27(6):897–923.

[4] Gao H, Duan Y, Miao H, et al. An approach to data consistency checking for the dynamic replacement of service process. IEEE Access. 2017;5(1):11700–11711.

[5] Gao H, Huang W, Yang X. Applying probabilistic model checking to path planning in an intelligent transportation system using mobility trajectories and their statistical data. Intell Autom Soft Comput (Autosoft). 2019;25(3):547–559.

[6] Gao H, Duan Y, Shao L, et al. Transformation-based processing of typed resources for multimedia sources in the IoT environment. Wirel Networks. 2019:1–17. doi:10.1007/s11276-019-02200-6

[7] Gao H, Xu Y, Yin Y, et al. Context-aware QoS prediction with neural collaborative filtering for internet-of-things services. IEEE IoT J. 2019. doi:10.1109/JIOT.2019.2956827

[8] Amirshahi P, Grippando S. Radio frequency interference monitoring system for weather satellite ground stations: challenges and opportunities. In: 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN). IEEE; 2017.

[9] Berman E. Movable window support device for a satellite TV dish. U.S. Patent No. 6,731,250. 2004 May 4.

[10] De Sanctis M, Cianca E, Araniti G, et al. Satellite communications supporting internet of remote things. IEEE IoT J. 2016;3(1):113–123.

[11] Ashjaee J, Rapoport LB, Kinkulkin D, et al. Satellite differential positioning receiver using multiple base-rover antennas. U.S. Patent No. 9,035,826. 2015 May 19.

[12] Li FH, Yin LH, Wu W, et al. Research status and development trends of security assurance for space-ground integration information network. J Commun. 2016;37(11):156–168.

[13] Jiang C, Wang X, Wang J, et al. Security in space information networks. IEEE Commun Mag. 2015;53(8):82–88. Zheng, Gan, Pantelis-Daniel Arapoglou, and Bjorn Ottersten.

[14] Zheng G, Arapoglou PD, Ottersten B. Physical layer security in multibeam satellite systems. IEEE Trans Wirel Commun. 2012;11(2):852–863.

[15] Wullems C, Pozzobon O, Kubik K. Signal authentication and integrity schemes for next generation global navigation satellite systems. In: European Navigation Conference GNSS. 2005, p. 1.

[16] Cruickshank HS. A security system for satellite networks. In: Proceedings of the Fifth International Conference on Satellite Systems for Mobile Communications and Navigation, London, UK. 1996.

[17] Sasaki T, Katoh T. Dual layer satellite communications system and geostationary satellite therefor. U.S. Patent No. 6,023,605. 2000 Feb 8.

[18] Zhang ZJ, Zhou Q, Zhang C, et al. New low-earth orbit satellites authentication and group key agreement protocol. J Commun. 2018;39(372(06)): 150–158.

[19] Zhu LH, Wang L, Li JS, et al. New entity authentication and access control scheme in satellite communication network. J Commun. 2018;39(372(06)): 77–84.

[20] Lu F, Zheng KF, Niu XX, et al. Security analysis of 3GPP authentication and key agreement protocol. J Software. 2010;21(7):1768–1782.

[21] Zeng X, Xu G, Zheng X, et al. E-AUA: an efficient anonymous user authentication protocol for mobile IoT. IEEE IoT J (Early Access). Jun 2018;6(2):1506–1519. doi:10.1109/JIOT.2018.2847447

[22] Lin H-Y. Efficient dynamic authentication for mobile satellite communication systems without verification table. Int J Satellite Commun Network. 2016;34(1):3–10.

[23] Chang CC, Cheng TF, Wu HL. An authentication and key agreement protocol for satellite communications. Int J Commun Syst. 2014;27(10):1994–2006.

[24] Lee CC, Li CT, Chang RX. A simple and efficient authentication scheme for mobile satellite communication systems. Int J Satellite Commun Network. 2012;30(1):29–38.

[25] Zhibo X, Ma H. Design and simulation of security authentication protocol for satellite network. Comput Eng Appl. 2007;43(17):130–132.

[26] Huang C, Zhu L, Li C, et al. A new satellite constellation networking certification and reliable maintenance protocol (DISA). In: 30th International Conference on Software Engineering & Knowledge Engineering. 2018.

[27] https://www.agi.com/