Effective Authentication Mechanism for Vehicular Fog Infrastructure

[1]Ta Thi Kim Hue, Nguyen Gia Duoc Nguyen Duy Tuan, An Braeken *, Kris Steenhaut *, Kieu-Ha Phung
SEEE, Hanoi University of Science and Technology, No.1, Dai Co viet street, Hanoi, Vietnam
*Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussel, Belgium
[1]Email: hue.tathikim@hust.edu.vn

***Abstract*—Recently, a tangible vehicular fog computing (VFC) platform based on oneM2M, called oneVFC, has been proposed. This platform enabled federated AI learning, but did not include sufficient attention to security. Therefore, this paper proposes an effective authentication and key agreement protocol based on the Elliptic curve cryptography (ECC) to enhance privacy preservation in an authenticated way in order to ensure safer AI training. Experimental results show that the proposed authentication mechanisms make reliable transmissions in highly-mobile vehicular networks possible, resulting in a desirable quality of service (QoS). The performance and security strength of two authenticated encryption modes of AES, GCM and CCM, are also compared and evaluated.**

***Index Terms*—vehicular fog computing, Federated Learning, GCM, CCM, Elliptic Curve cryptography.**

I. Introduction

Vehicular fog computation (VFC) is one of the Internet of Vehicles (IoV) infrastructures, which has been introduced to overcome limitations of the bandwidth and latency in cloud computing models in applications for vehicle networks. VFC mechanisms are applied for bringing data to the network edge, and for deploying data processing, management, and storage to be close to the users' devices. VFC ensures that traffic data has been collected, organized, stored and processed in real time, to make sure that the responses to the user's applications are more efficient. Advantages of VFC infrastructures enable a wide range of vehicle-based services on road safety applications, smart traffic control, entertainment services, etc.

However, an optimal balance between performance, security, and privacy is required to ensure the high quality of service (QoS). In this direction, several security solutions using authentication protocols have been proposed. In [1], a privacy-preserving querying and verifiable scheme is presented to enable the vehicles (end-users) to acquire local data from the road side units (RSUs). In another research, a privacy-preserved pseudonym scheme to address the location privacy is proposed by Kang et al. [2]. Similarly, authors in [3] presented a three-layered framework to obtain a trade-off between reliability, security and performance. Although authentication schemes have been proposed, most of them lack an estimation of the computational cost and the resource usage to confirm compliance with the VFC. On the other hand, a poorly designed lightweight cryptography solution not only leads to an inefficient secure scheme for data storage in the constrained fog nodes but also incurs the high computational cost overhead due to a large number of interactions among users and data centers [6]. To the best of our knowledge, the assessments of security levels are also ambiguous.

Recently, the design of a tangible vehicular fog computing (VFC) platform based on oneM2M, called oneVFC, is proposed. oneVFC is able to manage the distributed resources that help to reduce the processing time for AI-driven applications. It has been assessed on a hardware based testbed appropriate for on-vehicle computation-enabled devices. The feasibility of leveraging computing and communication resources of slowly moving cars in cross-section regions or of parked cars for supporting advanced vehicular applications has been investigated. Figure 1 shows the 3-layer hierachical architecture of VFC. The role of computation are taken up by the layer of fog nodes which are mini, portable computers attached on vehicles. Communications among vehicles or vehicles to infrastructure are maintained by Base Transceiver Station (BTS) units or road-side units (RSU) as shown in Fig. 1. This oneVFC platform will inherit the security properties of the one M2M infrastructure. oneVFC has been demonstrated to be very efficient in deploying AI-applications and especially, very appropriate for AI model training in a distributed manner.

Federated Learning (FL) is a novel distributed training approach [7], [8], in which the global model can be achieved by aggregating the locally trained models. The local devices train local models by local data sets. Afterwards, only model parameters are exchanged between servers and clients to preserve data privacy. Therefore, the communication efficiency and data privacy in both distributed learning and federated learning machines are mandatory requirements.

II. Authenticated encryption techniques and system model

A. Authenticated encryption techniques

The security goal of an authenticated encryption (AE) is to provide simultaneously confidentiality, integrity, and authenticity over processed data, in which the ciphertext derived from the plaintext comes along with an authenticated tag. Authenticated decryption is an invertible process for retrieving the integrity of the plaintext. Moreover, anonymity and unlinkability of the requestors is also a very important security requirement. The work in [4] defined an efficient Identity and Access Management (IAM) system to authenticate and authorize clients accessing sensor data source from the OM2M

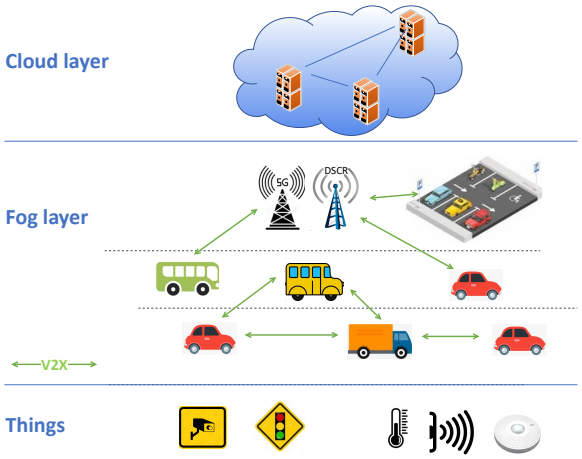

Fig. 1. OneVFC Architecture

platform in an anonymous way. It also analyzed the authenticated encryption using AES with the CCM operation mode over the OM2M platform for wireless sensor networks. The research evaluated the payload encryption and link layer encryption for time and memory efficiency. Moreover, the protocol has been shown to resist replay attacks, man-in-the-middle attacks, impersonation attacks, denial of service and retrieval of identity attacks.

We apply the feasibility of the protocol of [4] to our scenario of the oneVFC platform such that clients willing to share protected features of the AI model training between the oneVFC Fog managers and data owners in an anonymous way are securely authenticated and authorized. In addition, we extend the solution of oneVFC [4] by also including the Galois/Counter Mode (GCM), besides the Counter/CBC-MAC (CCM) mode and evaluate the proposed authenticated encryption schemes on a lab scale testbed. Experimental results ensure that the AI model training in the Federated Learning approach is secured in an effective way on the oneVCF platform and obtains the desirable quality of service (QoS).

B. System model

The architecture shown in Fig. 2 has four main components which are described as follows. The Fog manager nodes play the role of the access gateways and connect with the computing devices on the vehicles which are called the fog worker nodes. A fog manager node is responsible for managing worker nodes in a given geographical area. Fog workers perform the computation jobs which they are assigned to by the fog managers. Data and results are relayed back and forth between the manager and the worker nodes. A dynamic authorization system (DAS) is adopted to provide temporary access permissions. An Infrastructure Node with Common Service Entity (CSE), called an IN-CSE node working as a system server, is in charge of storing the data gathered and relayed by the fog managers. The IN-CSE node must also guarantee that only authenticated and authorized fog workers can interact with fog managers. In general, compute service

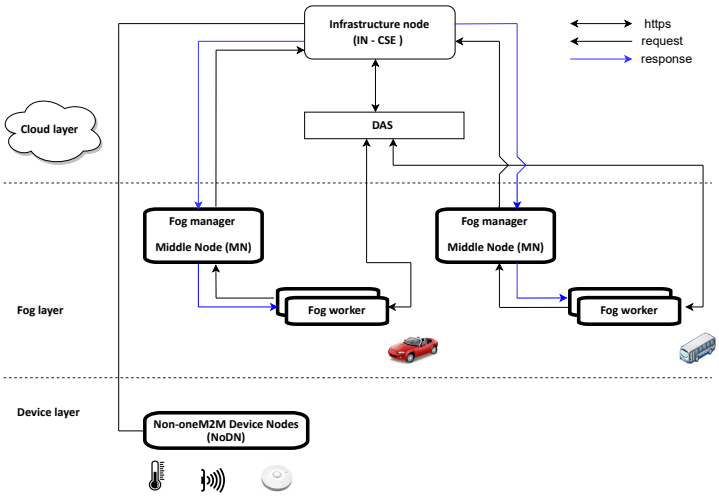

Fig. 2. oneVFC secure platform

requests can come from a number of fog worker nodes and the manager node will return the aggregated results to the on-road vehicle users who requested the service. The AI-based services are deployed to process huge amounts of data generated by Closed-circuit television (CCTV) cameras installed along the roads in combination with data from sensors, cameras on the vehicles. In [3], the authors proposed the oneVFC platform which supports a federated learning framework and AI-based services. In a FL framework, the fog manager is developed to carry out machine learning tasks from data distributed at the fog worker nodes, while the raw data is kept at the workers and never uploaded to the central server, which is known as the IN-CSE server. We also assume that secure channels exist between the IN-CSE server and the DAS, and between the DAS and the fog worker. These channels are fundamental for the exchange of sensitive authorization information and the privacy of AI processing data, respectively. According to the design of the manager node in [3], the Federated Learning AI model is performed and exchanged between the fog manager and the fog worker. To summarize, a securely processing procedure consists of five main steps, given as follows:

- A service requester who can be a fog worker requests a service to the manager. This service includes applications in distributed learning or Federated Learning (FL) approach for the AI-model training. The requester should first register to the DAS to obtain a ticket access before being able to communicate with the manager. The DAS will send the registrations of the fog workers to the storage of the IN-CSE server, and then the IN-CSE server and the DAS share a symmetric key that will be used to verify the fog worker's authenticity.
- The fog manager processes the service request by assigning subtasks to the worker nodes and retrieves the fog worker's authenticity from the IN-CSE server. Consequently, the IN-CSE server will find the list of suitable

computing nodes for doing the requested job and send notification to the managers.
- The fog manager assigns a computing process to the worker nodes including feature sets, embedding parameters and the execution program.
- The fog workers download the data model, execute the computation task, and send back the results to the manager node. The data can be text files representing the AI model parameters, which are all considered input parameters for the model aggregation.
- The fog manager aggregates the local models and updates the global model. The process of exchanging local models and updating the global model are repeated till the required precision is reached, and then the fog manager transmits the final model to the service request node.

In the next section, we describe in more detail how the authentication and key agreement protocol of [4] can be integrated in the oneVFC platform, according to the architecture and processes described above.

III. Proposed Data Authentication and Integrity Scheme for oneVFC secure platform

Throughout this section, the notations are denoted as follows: G is a base point in Elliptic Curve Qu-Vanstone (ECQV), $H(.)$ is a hash operation SHA256, $cert_u$ is the user's implicit certificate and $||$ is a concatenating operation.

The goal is to strengthen the security of the VFC data, i.e., FL AI training model feature sets and parameters. Therefore, an indirect dynamic authorization system (DAS) is required. The intended fog worker has to request access tokens to the DAS before being able to communicate with the fog managers. This allows to protect the FL AI training model from the fog manager to malicious fog workers. The overall process in the oneVFC infrastructure can be segregated into the following five phases: 1) System Initialization Phase, 2) Registration and Release Ticket Phase, 3) Mutual Authentication and Key Exchange Phase, and 4) Service-Delivery Training Model Phase. The detailed description of these phases is presented below:

A. System Initialization

During this phase, the DAS prepares a pair of keys including a private key k and a public key P_{DAS} following the approach of the Elliptic Curve Qu Vanstone (ECQV) implicit certificate mechanism. The DAS creates the database of accessing tickets with a referring table named UserAccess. This table contains the features of each ticket such as an User's identifier ID_u, $holderName$, $permission$. Two functionalities are the derivation of the worker's key pair and the worker's subscription to a particular access ticket. Table. I shows the certified scheme from the DAS. The fog worker derives a pair of keys and requests a certification to the DAS. A random value u is generated by the fog worker, the point $U = uG$ is computed by an EC multiplication. The fog worker sends its identifier ID_u and the computed point U to the DAS. The latter also picks a random value a, generates the point $A = aG$ as an EC multiplicative operation, and then the fog worker's implicit certificate $cert_u = U + A$. Next, the DAS computes the implicit signature as:

$$q_u = H(cert_u||ID_u)a + k$$

and the fog worker's public key is computed as:

$$P_u = H(cert_u||ID_u)cert_u + P_{DAS}$$

Upon reception of the message $(cert_u, q_u, P_{DAS}, P_{IN})$, the worker node derives its private and public key d_u, P_u as:

$$d_u = H(cert||ID_u)u + q_u$$

$$P_u = d_uG$$

To verify the authenticity of the DAS, the fog worker computes its own public key by using the public key of the DAS: $P_u^* = H(cert_u||ID_u)cert_u + P_{DAS}$ If P_u equals P_u^*, the worker is sure about the authenticity of the certificate received and can trust the key pair computed using the DAS's implicit signature q_u. The IN-CSE server receives and stores (ID_u, P_u) in its storage.

B. Registration and Release Tickets

The fog worker registers to an AI service and receives a temporary access right. The ticket includes the specific User's identifier corresponding to the AI service the fog worker (data owner) is interested in, the type of the AI services and the expiration date. The registration phase is started by the fog worker to receive an access token. The detailed elaboration of this phase is given in Table II.

1) The fog worker generates two random numbers c, z and timestamp T_R, and then computes the EC point $Z = zG$, and two symmetric keys, denoted $K_r = H(d_uP_{DAS}||T_R$ and $K_z = H(zP_{DAS}||T_R)$, respectively.
2) The fog worker requests a subscription Sub which includes the AI services denoted AIS_n, the type of data sets $Type$, the random number c, the fog worker's identity ID_u, and the key K_r for guaranteeing the authenticated process. The subscription request $Sub = E_{K_z}(AIS_n||Type||c||ID_u||K_r)$ is sent to the DAS together with the timestamp T_R and the point Z.
3) Upon reception of the request, the DAS computes the key K_z using Z and the received timestamp T_R, and decrypts Sub in order to obtain the name of the AI services AIS_n the client is interested in, the type of dataset $Type$, the random number c, ID_u and K_r.
4) Next, it checks if the worker's identifier and corresponding public key P_u are already present in the UserAccess table. Then, it computes the same disposable symmetric key K_r using the received timestamp T_R and checks if it corresponds with the received value.
5) The DAS generates a new symmetric key K_t, based on the usage of the masked identity $Q_u = H(ID_u||c)$ and the IN-CSE's public key P_{IN}. The common shared key is defined as $K_t = H(Q_u||P_{IN})$.

TABLE I
FOG WORKER DERIVES ITS KEY PAIR AND RECEIVES THE IMPLICIT CERTIFICATE FROM THE DAS

fog worker (ID_u)		**DAS** (k, P_{DAS})		**IN-CSE server** (k_{IN}, P_{IN})
Generate random u, $U = uG$	$\xrightarrow{ID_u, U}$		$\xleftarrow{P_{IN}}$	
		Generate random a $A = aG$ $cert_u = U + A$ $q_u = H(cert_u \|\| ID_u)a + k$ $P_u = H(cert_u \|\| ID_u)cert_u + P_{DAS}$ Store $(ID_u, cert_u, P_u)$		
			$\xrightarrow{(ID_u, P_u)}$	
				Receive (ID_u, P_u)
	$\xleftarrow{cert_u, q_u, P_{DAS}, P_{IN}}$			
$d_u = H(cert_u \|\| ID_u)u + q_u$ $P_u = d_uG$ Verify $P_u = H(cert_u \|\| ID_u)cert_u + P_{DAS}$				

6) Finally, the DAS sends the ticket, encrypted with the key K_r, back to the fog worker in response to the subscription request.

C. Key agreement phase and shared training model

The fog worker requires to deploy the FL AI training model services. The fog manager reevaluates the access request after the fog worker's authenticity has been verified. If the ticket access is granted, the fog worker and fog manager end up sharing a common session key. Table III shows the key agreement scheme, a common session key SK is computed and shared by the IN-CSE server. Beyond this, the algorithm allows that the fog manager can verify the fog worker's authenticity while the DAS checks if the presented ticket contains a valid identifier for the requesting access. Firstly, the fog worker generates the masked identity Q_u using the secret random number c and sends a message containing the masked identity Q_u and the ticket received during the registration phase to the DAS. As soon as the fog manager receives the authentication request, it computes the decryption key K_t, using the masked identity Q_u and the IN-CSE's public key P_{IN}, to allow the decryption of the ticket. If the result contains a valid AI service type AIS_n, a valid expiration time and a user identifier, then the fog worker is authenticated. Next, the fog manager generates a timestamp T_s, retrieves the AE identifier containing the measurements corresponding to the requested AI service type and sends a message which consists of the AE identifier, T_s and the user identifier to the IN-CSE server over the channel on the oneVFC platform. The IN-CSE computes the session key SK using the received timestamp T_s and delivers it to the fog manager through the same channel. Upon reception of this message, the fog manager protects the FL AI training model using the session key SK and sends the encrypted FL AI training model to the fog worker. Finally, the fog worker receives the encrypted FL AI training model and the timestamp T_s, and computes the same session key SK using its secret key d_u as $SK = H(d_uP_{IN}||T_s$. This shared session key can now be used by the fog worker and fog manager to create a secure channel for the coming session.

IV. Experimental results and Security Analysis

A. Time cost evaluation

We evaluate the time required to perform the proposed system which is described previously in Section II and the possibility to use constrained WSN device as fog worker. In addition, the time required by our IAM scheme to identify, authenticate, authorize and establish a secure channel is the total time cost, involving the EC point multiplication and EC point addition, the symmetric encryption/decryption and the hash operation. In the IAM system, we execute two authenticated encryption operation modes that are the AES-128 with CCM and GCM modes. To investigate the overall latency of the proposed scheme, we perform the benchmarking analysis in three steps. We first measured the time cost required by each cryptographic operation involved in the algorithm. The times for EC point multiplication (ECmult), EC point addition (ECadd), AES-128 CCM/GCM (Encryption/Decryption) and SHA256 on a Raspberry Pi are denoted as T_m, T_a,T_s and T_h, respectively. For the symmetric encryption algorithm and the hash function we used a payload size of 128 bytes. The average over 20000 samples are shown in Table IV.

To evaluate the performance of our scheme, we computed the time required by the fog worker to establish a shared FL AI model training with the Fog manager. Table V indicates the computational time required during the ECQV Initialization phase, the fog worker registration phase and the key exchange phase. We distinguish between the usage of CCM and GCM mode in each of these phases. In this experiment, the fog worker device is considered to be a Raspberry PI 4 (RAM 512MiB CPU 1.5GHz). Finally, to verify the proposed authenticated encryption scheme, a testbed was built as a scale model of the reality. The testbed includes a fog manager node, a fog worker node, the IN-CSE server and the DAS. The underlying network supporting the communication among nodes is Wifi-based. The hardware and software configurations of the fog manager are a personal computer (RAM 8G CPU 2.2GHz), Network switch CISCO, Linksys EA2700 Wireless-N 2.4Ghz - 5GHz. The measurements of the total time cost with the 95% confidence level over 30 samples are represented

TABLE II
REGISTRATION AND RELEASE TICKET SCHEME

fog worker $(ID_u, d_u, P_u, P_{DAS})$		**DAS** (k, P_{DAS}, P_{IN})
Generate random c, z $Z = zG$		
Generate timestamp T_R		
$K_r = H(d_u P_{DAS} \| \| T_R)$		
$K_z = H(zP_{DAS} \| \| T_R)$		
$Sub = E_{K_z}(AIS_n \| \| Type \| \| c \| \| ID_u \| \| K_r)$		
	$\xrightarrow{T_R, Sub, Z}$	
		$K_z = H(kZ \| \| T_R)$
		$D_{K_z} -> AIS_n, Type, c, ID_u, K_r$
		Check if $K_r^* = H(kP_u \| \| T_R) = K_r$
		$Q_u = H(ID_u \| \| c)$
		$K_t = H(Q_u \| \| P_{IN})$
		$Ticket = E_{K_t}(ID_u \| \| AIS_n \| \| T_{exp})$
		$ET = E_{K_r}(Ticket \| \| T_{exp})$
	$\xleftarrow{ET}$	
$D_{K_r}(ET) -> Ticket, T_{exp}$		

TABLE III
KEY AGREEMENT ALGORITHM

fog worker (ID_u, d_u, P_u, P_{IN})		**Fog manager**		**IN-CSE server** (k_{IN}, P_{IN})
$Q_u = H(ID_u \| \| c)$	$\xrightarrow{Q_u, Ticket}$		$\xleftarrow{P_{IN}}$	
		$K_t = H(Q_u \| \| P_{IN})$		
		$D_{K_t}(Ticket) -> ID_u, R_u, T_{exp}$		
		Generate timestamp T_s		
		Retrieve $AE - ID$ using R_n		
			$\xrightarrow{AE-ID, ID_u, T_s}$	$SK = H(k_{IN} P_u \| \| T_s)$
			$\xleftarrow{SK}$	
		$EU = E_{SK}(AI - TrainingModel)$		
	$\xleftarrow{T_s, EU}$			
$SK = H(d_u P_{IN} \| \| T_s)$				
$D_{SK} -> AI - TrainingModel$				

TABLE IV
THE AVERAGE TIME OF THE EC POINT MULTIPLICATION, EC POINT ADDITION, AES-128 CCM/GCM AND SHA256 HASH FUNCTION IMPLEMENTED ON A RASPBERRY PI 4 (RAM 512MIB CPU 1.5GHZ)

Measurement points	Operations	Latency (μs)
T_m	ECmult (input 32 bytes)	673.46 ± 7,030
T_m	ECmult (input 16 bytes)	859.493 ± 5,720
T_a	ECadd (input 32 bytes)	13.514 ± 0,109
T_a	ECadd (input 16 bytes)	14.696 ± 0,077
T_s	AES 128 - CCM	20,230 ± 0,051
T_s	AES 128 - GCM	13,530 ± 0,047
T_h	SHA256	3.740 ± 0,037

TABLE V
THE TOTAL COMPUTATIONAL TIME USING AES 128-CCM AND AES 128-GCM ($milisecond$)

Phases	Operations	AES-128	
		CCM	GCM
Initialization	$3T_m + T_a + 2T_h$	9.161	7.979
Registration	$3T_m + 3T_s + 6T_h$	7.865	7.225
Key exchange	$T_m + T_s + 4T_h$	7.239	7.300
Total	$7T_m + T_a + 3T_s + 6T_h$	24.265	22.504

in Figure 3. The computational time is measured by deploying the protocols on the local devices such as Raspberry Pi or PC, not connecting to the internet. The testbed latency is including the response time of the Wifi network. In any case, the authenticated encryption scheme with AES 128 - GCM still gives in both cases better execution time.

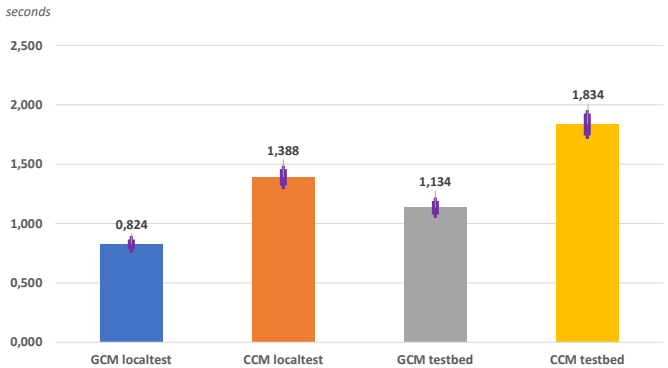

Fig. 3. The computation time and the testbed latency (response time) with AES 128-CCM and AES 128-GCM

B. Security Analysis

For the security evaluation of the protocol in the presence of an active and passive attacker, we can rely on the analysis

provided in [4]. To further investigate the strength of the protocol with respect to for instance side channel attacks and fault attacks, we perform two security tests, measuring the Ciphertext Difference Rate (CDR) and the Key sensitivity. In this context, the attacker tries to infer any relation between the encrypted and original data. Therefore, different statistical tests have been applied to obtain the change rate of the number of digits between encrypted messages when one digit of the plaintext is changed. The different statistical properties are evaluated by the CSR. We apply it with a set of 100 data instances of size 1470 bytes each. Figure 4 shows the values of the CDR from the GCM and CCM modes in the oneVFC platform to verify the influence of a single byte change in the plain data to the corresponding ciphered data. The CDR of the authenticated encryption with GCM is slightly better than the CCM mode and always greater than 87%, which is sufficient to resist differential attacks.

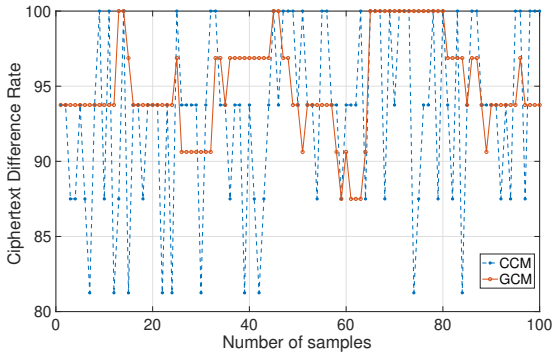

Fig. 4. Ciphertext Difference Rate corresponding with a single byte change in the plaintext

The key sensitivity of the cryptosystem should be as high as possible. The key sensitivity determines the impact of a small difference in the keys on the encryption and decryption processes. The Message Authentication Code (MAC) difference rate (MDR) is the MAC changing rate by adapting the secret key with a small difference. Here, the MDR caused by the key difference is defined as follows:

$$MDR = \frac{diff(Ti, Ti_1) + diff(Ti, Ti_2)}{2 \times sizeof(MAC)} \times 100\% \quad (1)$$

where Ti, Ti_1 and Ti_2 are MACs using the secret key K, $K + \Delta K$ and $K - \Delta K$, respectively. The small difference in secret keys is ΔK and $diff(Ti, Ti_1)$ is the number of different values of bits between the MACs Ti and Ti_1. The average key sensitivity in Figure 5 is for both modes higher than 90%, with a small advantage for GCM. As a consequence, a small difference in the key causes a high amount of changes in the cipher images.

V. Conclusion

In this paper, we have designed and implemented an effective security mechanism for the oneVFC plaform, allowing fog managers and workers to establish a trust relationship

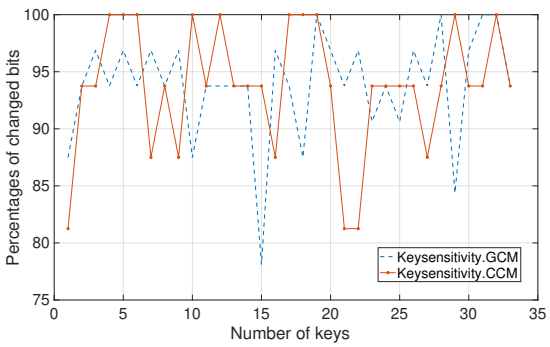

Fig. 5. The difference between MACs (% of changed bits)

to share the AI training model in the Federated Learning approach on secure channels. All the required authenticated operations on our testbed requires for the fog worker 1.13 and 1.83 seconds with AES 128 GCM and CCM, respectively. As a consequence, a high throughput with reasonable hardware resources can be obtained. The encryption process based on AES with GCM mode exhibits a slightly higher performance efficiency and security protection compared to the CCM mode.

Acknowledgment

The research work is supported by Vingroup Innovation Foundation (VINIF) in project code VINIF.2019.DA16.

References

[1] Hassan Noura, Ola Salman, Ali Chehab, Raphael Couturier, “Preserving data security in distributed fog computing,” Ad Hoc Networks Volume 94, November 2019, 101937

[2] Jiawen Kang; Rong Yu; Xumin Huang; Yan Zhang, ”Privacy-Preserved Pseudonym Scheme for Fog Computing Supported Internet of Vehicles”, IEEE Transactions on Intelligent Transportation Systems (Volume: 19, Issue: 8, Aug. 2018

[3] Phung Kieu-Ha, Tran Hieu, Nguyen Thang, Dao Hung V., Tran-Quang Vinh, Truong Thu-Huong, Braeken An and Steenhaut Kris., ”oneVFC—A Vehicular Fog Computation Platform for Artificial Intelligence in Internet of Vehicles,” in IEEE Access, vol. 9, pp. 117456-117470, 2021.

[4] Simone Patonico, Thanh-Long Nguyen, Placide Shabisha, An Braeken, Kris Steenhaut, “Toward the inclusion of end-to-end security in the OM2M platform,” The Journal of Supercomputing volume 77, pages 4056–4080 (2021)

[5] Qinglei Kong, Rongxing Lu , Maode Ma and Haiyong Bao, ”A Privacy-Preserving and Verifiable Querying Scheme in Vehicular Fog Data Dissemination”, IEEE Transaction on Vehicular Technology, VOL. 68, NO. 2, FEBRUARY 2019

[6] Abebe Diro, Haftu Reda, Naveen Chilamkurti, Abdun Mahmood, Noor Zaman, Yunyoung Nam, “Lightweight Authenticated-Encryption Scheme for Internet of Things Based on Publish-Subscribe Communication,” IEEE Access, vol. 8, pp. 60539-60551, 2020.

[7] Saurav Prakash and et al.,”Coded Computing for Low-Latency Federated Learning Over Wireless Edge Networks”, IEEE Journal on Selected Areas in Communications, Volume 39, Issue 1, Jan. 2021 pp 233–250

[8] Haftay Gebreslasie Abreha, Mohammad Hayajneh and Mohamed Adel Serhani,”Review Federated Learning in Edge Computing: A Systematic Survey”, Sensors 2022, 22, 450.

[9] An Braeken, ”Public key versus symmetric key cryptography in client–server authentication protocols”, International Journal of Information Security volume 21, pages 103–114 (2022)