Emerald Insight

International Journal of Pervasive Computing and Com...

Collaborative Mutual Identity Establishment (CMIE) for the future internet
Nancy Ambritta P Poonam N. Railkar Parikshit N. Mahalle

Article information:

To cite this document:
Nancy Ambritta P Poonam N. Railkar Parikshit N. Mahalle , (2015),"Collaborative Mutual Identity Establishment (CMIE) for the future internet", International Journal of Pervasive Computing and Communications, Vol. 11 Iss 4 pp. 398 - 417
Permanent link to this document:
http://dx.doi.org/10.1108/IJPCC-04-2015-0024

Downloaded on: 07 November 2016, At: 22:33 (PT)
References: this document contains references to 19 other documents.
To copy this document: permissions@emeraldinsight.com
The fulltext of this document has been downloaded 66 times since 2015*

Users who downloaded this article also downloaded:

(2015),"A password-authenticated secure channel for App to Java Card applet communication", International Journal of Pervasive Computing and Communications, Vol. 11 Iss 4 pp. 374-397 http://dx.doi.org/10.1108/IJPCC-09-2015-0032

(2015),"An analysis of tools for online anonymity", International Journal of Pervasive Computing and Communications, Vol. 11 Iss 4 pp. 436-453 http://dx.doi.org/10.1108/IJPCC-08-2015-0030

Access to this document was granted through an Emerald subscription provided by emerald-srm:563821 []

For Authors

If you would like to write for this, or any other Emerald publication, then please use our Emerald for Authors service information about how to choose which publication to write for and submission guidelines are available for all. Please visit www.emeraldinsight.com/authors for more information.

About Emerald www.emeraldinsight.com

Emerald is a global publisher linking research and practice to the benefit of society. The company manages a portfolio of more than 290 journals and over 2,350 books and book series volumes, as well as providing an extensive range of online products and additional customer resources and services.

Emerald is both COUNTER 4 and TRANSFER compliant. The organization is a partner of the Committee on Publication Ethics (COPE) and also works with Portico and the LOCKSS initiative for digital archive preservation.

*Related content and download information correct at time of download.

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

The current issue and full text archive of this journal is available on Emerald Insight at:
www.emeraldinsight.com/1742-7371.htm

Received 25 April 2015
Revised 25 April 2015
Accepted 7 September 2015

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Collaborative Mutual Identity Establishment (CMIE) for the future internet

Nancy Ambritta P., Poonam N. Railkar and Parikshit N. Mahalle
Department of Computer Engineering,
Smt. Kashibai Navale College of Engineering, Pune, India

Abstract

Purpose – This paper aims at providing a comparative analysis of the existing protocols that address the security issues in the Future Internet (FI) and also to introduce a Collaborative Mutual Identity Establishment (CMIE) scheme which adopts the elliptical curve cryptography (ECC), to address the issues, such as content integrity, mutual authentication, forward secrecy, auditability and resistance to attacks such as denial-of-service (DoS) and replay attack.

Design/methodology/approach – This paper provides a comparative analysis of the existing protocols that address the security issues in the FI and also provides a CMIE scheme, by adopting the ECC and digital signature verification mechanism, to address the issues, such as content integrity, mutual authentication, forward secrecy, auditability and resistance to attacks such as DoS and replay attack. The proposed scheme enables the establishment of secured interactions between devices and entities of the FI. Further, the algorithm is evaluated against Automated Validation of Internet Security Protocols and Application (AVISPA) tool to verify the security solutions that the CMIE scheme has claimed to address to have been effectively achieved in reality.

Findings – The algorithm is evaluated against AVISPA tool to verify the security solutions that the CMIE scheme has claimed to address and proved to have been effectively achieved in reality. The proposed scheme enables the establishment of secured interactions between devices and entities of the FI.

Research limitations/implications – Considering the Internet of Things (IoT) scenario, another important aspect that is the device-to-location (D2L) aspect has not been considered in this protocol. Major focus of the protocol is centered around the device-to-device (D2D) and device-to-server (D2S) scenarios. Also, IoT basically works upon a confluence of hundreds for protocols that support the achievement of various factors in the IoT, for example Data Distribution Service, Message Queue Telemetry Transport, Extensible Messaging and Presence Protocol, Constrained Application Protocol (CoAP) and so on. Interoperability of the proposed CMIE algorithm with the existing protocols has to be considered to establish a complete model that fits the FI. Further, each request for mutual authentication requires a querying of the database and a computation at each of the participating entities side for verification which could take considerable amount of time. However, for applications that require firm authentication for maintaining and ensuring secure interactions between entities prior to access control and initiation of actual transfer of sensitive information, the negligible difference in computation time can be ignored for the greater benefit that comes with stronger security. Other factors such as quality of service (QoS) (i.e. flexibility of data delivery, resource usage and timing), key management and distribution also need to be considered. However, the user still has the responsibility to choose the required protocol that suits one's application and serves the purpose.

Originality/value – The originality of the work lies in adopting the ECC and digital signature verification mechanism to develop a new scheme that ensures mutual authentication between participating entities in the FI based upon certain user information such as identities. ECC provides

International Journal of Pervasive Computing and Communications
Vol. 11 No. 4, 2015
pp. 398-417
© Emerald Group Publishing Limited
1742-7371
DOI 10.1108/IJPCC-04-2015-0024

efficiency in terms of key size generated and security against main-in-middle attack. The proposed scheme provides secured interactions between devices/entities in the FI.

Keywords AVISPA, CMIE, Content integrity, DoS and replay attack, ECC, Mutual authentication

Paper type Research paper

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

1. Introduction

The Internet has its origin dating back to the 1980s that basically involved interconnected computers serving the purpose of dissemination of information and documents. It served the purpose of collaboration between people without any geographical regard. Telecommunication has in fact been a supporting shoulder in making the Internet a success with its basic ideas of using the electric and electronic media for exchange of information over long distances. The wireless communication revolutionized the Internet world with its notable and effective communication with radio waves. The introduction of software (Web browsers) has enabled common users to view and access content easily than ever before which lead to the bloom of the e-commerce category with the involvement of common man. With the advancements in technology, the Internet slowly evolved from being a mere source of information to a means of tracking and identification of entities, thereby taking the world online.

The semiconductor technology (biometric smartcards, radio-frequency identification [RFID] tags, etc.) has paced this evolution enabling the objects to communicate intelligently amongst themselves, thereby taking the world to a new era "The Internet of Things (IoT)". With this new era in progress, there will be an enormous change seen in the way people communicate with the objects and things in the world, thus creating an extraordinary handiness and efficiency to the end-user. The mobile Internet and the intelligent homes are stepping stones to this convenient world of IoT. As mentioned in an article on cyber-physical systems (CPS) by EIT Digital (2015) and SINTEF (2015):

> […] the CPS are a bridge that connect the IoT with the higher level services by providing efficient mechanisms to merge the physical world with the virtual world. Using special sensors and embedded systems monitors, the CPS enables the collection of relevant data regarding a particular process which is made globally available.

The participating semiconductor devices and embedded systems communicate with one another via software applications and perform necessary actions in response to events that occur in the physical world with the help of the globally available data.

With this enormous revolution in the Internet world, comes the dark side of security threat. The Future Internet (FI) being a confluence of various fields, namely, the cloud, mobile, social media, devices and Big data requires security to be addressed with utmost priority due to the risks involved in handling sensitive private data. The vulnerabilities of the various devices in the IoT are due to the inclusion of cloud services and mobile applications to control and access the devices. Following are the security requirements for IoT.

1.1 Privacy

With the innumerable devices collecting personal information including sensitive data, such as credit card numbers and social security numbers, the untrusted cloud and mobile applications that work alongside these devices, exposure of the sensitive data over the public channel in an unencrypted and unprotected manner invite security risks.

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Securing the interactions between the devices of the IoT and securing the content itself that is exchanged are of prime importance to help maintain privacy of user data.

1.2 Authentication and authorization

An improper authentication and authorization mechanism between devices and cloud allows the attacker to take advantage of the vulnerability and gain access to the sensitive data. Devices should properly identify each other's genuineness by following strong authentication and authorization mechanisms (e.g. strong passwords, certificates exchange and verification) to assure the protected exchange of sensitive data. Other common security threats involving the devices are the denial-of-service (DoS) attacks, replay attacks and collusion attacks. Proper designing of the Web interface and infrastructure and proper session management activities can suitably address these security threats effectively.

The ease and convenience brought through the introduction of the "IoT", an integral part of the FI has given rise to many important challenges in terms of security attacks. To meet the growing demands in the current Internet, new architectures based on new design principles have to be developed which have been discussed by Pan *et al.* (2011) in their article. The Internet has evolved from an end-to-end packet delivery model to a content-centric model. Hence, new architectures should support secure content-oriented, context-aware exchange of data rather than a mere host-to-host delivery of data. Refinements in different aspects such as security, privacy, usability, reliability, mobility and availability have to be addressed separately, and all these ideas should be converged to form a more resilient architecture for the FI. Further, with IoT, a need to maintain secured interactions between devices and also securing the globally exchanged data has to be maintained with utmost importance. Considering the above requirements, many researches in developing suitable architectures are in practice, namely NEBULA of the FI architecture program, and MobilityFirst projects are of noteworthy importance.

2. Motivation

This section explains the necessity for a proper authentication and authorization mechanism in the IoT scenario. Lack of a proper authentication and authorization scenario leads to the following security threats, namely:

- privacy theft;
- difficulty in maintenance of principle of least privilege for authorized users;
- lack of auditability and accountability and delegation of access rights;
- lack of trust between participating devices;
- lack of secured interactions;
- denial of access/service;
- loss/corruption of sensitive data; and
- complete compromise of devices/user accounts.

In Figure 1, the intended User A sends a request along with his credentials to a Provider B. The Intruder I sniffs the data midway and obtains the credentials and poses to be the intended user to the Provider B. Due to the lack of a proper authentication and

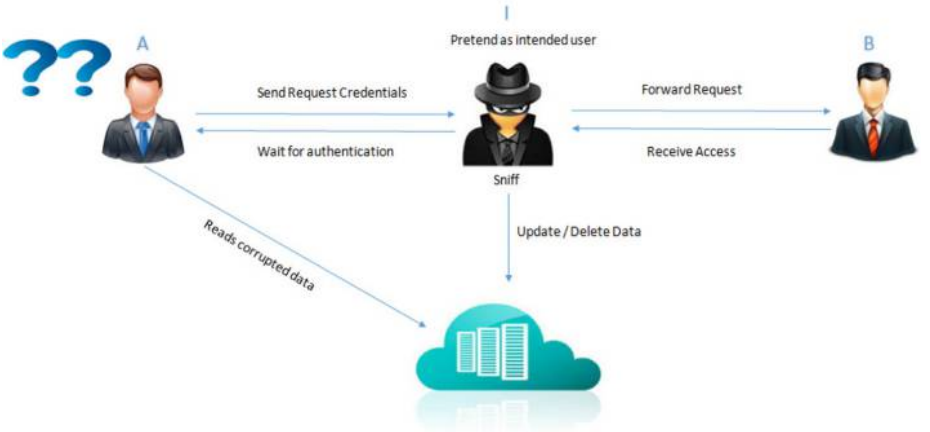


Figure 1.
Motivation

authorization mechanism for validation, the Provider B assumes the attacker to be the intended user and provides the requested information and also carries on further communication with the intruder, thereby breaking the security of the system. This private data theft allows the intruder to manipulate/delete the data in the global storage which when read by an intended user might experience DoS/access or stale/unintended data retrieval. This lack of secured interactions leads to a loss of trust between the participating entities and difficulty in auditing and monitoring the activities of intruders/malicious users.

3. Related works and evaluation

Several schemes have been introduced by researchers to address the security issues that prevail in the various areas that make up the FI. Chan and Zhou (2014) have analyzed authentication in CPS (smart grid) and also addressed substitution attack, a kind of man-in-the-middle attack using two-factor cyber physical device authentication protocol by combining a novel contextual factor (that deals with physical connectivity) with the authentication factor in the challenge-response protocol. This protocol ensures the protection of the digital identities of the devices and its controllability. However, mutual authentication that ensures trust between both participating entities, resistance of the system to privacy theft of user sensitive data/attributes, content integrity of messages exchanged and forward secrecy (an adversary should not be able to trace back to previously communicated data with the current information held by him) have not been spoken about in their work which are essential factors for ensuring secured interactions between entities in FI.

David and Francisco (2014) have presented the suitability of using simple access control lists and capability-based security schemes in Kurento and Nubomedia by using token-based protocols to implement authentication, authorization and auditing. Kurento is an open source software project devoted to building a Web-based Real Time Communication (WebRTC) media infrastructure with Nubomedia as a Platform-as-a-Service (PaaS) written on top of it. However, it still fails to address the important aspects such as mutual authentication, trust between entities, content integrity, privacy of user sensitive data, forward secrecy and resistance against attacks such as DoS and

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

replay. Hong *et al.* (2015) proposed a Shared Authority-based Privacy Preserving Authentication (SAPA) protocol to address the privacy issues using the Attribute-based Access Control (ABAC) and proxy re-encryption. It ensures data anonymity by leveraging the HMAC mechanism and maintains forward secrecy by using session variational parameters to make the communication dynamic. However, mutual authentication and thereby trust establishment is still an issue to be addressed.

Cirani *et al.* (2015) have proposed an architecture that works upon HTTP/ Constrained Application Protocol (CoAP) services in providing an authorization framework that can be integrated upon invoking an external OAuth-based authorization service (OAS). The proposed architecture assumes the usage of existing Extensible Authentication Protocol (EAP) versions to provide content integrity. It also requires that a predefined level of trust exists with the proxy used, and the separate authorization framework is used to provide protection against replay attacks. However, issues regarding privacy theft, DoS attack and forward secrecy have not been addressed in the system. Ning and Yang (2015) have focused on existing U2IoT architecture to develop an Aggregated Proof-based Hierarchical Authorization (APHA) scheme for layered networks. Data integrity is achieved using the one-way hash function and forward secrecy using pseudo-random numbers that provide session freshness and randomness. Mutual authentication is achieved using chebyshev chaotic maps and directed path descriptors. However, auditability and resistance to DoS and replay attacks have not been addressed in the protocol.

Ruj *et al.* (2014) have proposed a new privacy preserving authenticated access control scheme that uses attribute-based encryption and attribute-based structure to maintain anonymity of users. The scheme involves a trustee for token distribution and also safeguards the system from revoked users. The system also protects the privacy of data stored by user on cloud. However, aspects involved in security such as mutual authentication and forward secrecy have not been discussed. Also, the system is assumed to be resistance to replay attack to a certain extent by securing the channel using SSH (Secure Shell). Noureddine and Bashroush (2013) have quoted their previous optimization work on the OAuth 2.0 protocol by introducing an independent authorization server to solve the performance challenges. The usage of OAuth protocol ensures privacy of user sensitive credentials. Further, as an extension to this work, they have introduced the concept of referral tokens to solve identity federation challenges in applications. However, the system requires trust to be provisioned as a prerequisite step to authentication. Hence, here again the requirement of mutual authentication, forward secrecy and resistance to replay attacks has not been addressed. Also, integrity of stored data is achieved via digital signatures while integrity of exchanged user sensitive credentials is not addressed.

Liu *et al.* (2014) have proposed an effective solution that supports multiple users to access and manipulate the stored data. The system ensures anonymity of users by adopting group signatures and dynamic broadcast encryption technique (Access Control Vector – Broadcast Group Key Management [ACV-BGKM]). However, the system has not specifically considered the required security aspects such as mutual authentication and resistance to attacks such as DoS and replay attacks. Also, all of the above protocols except the idea proposed by Ning and Yang (2015) provide only one way authentication, i.e. authentication of device-to-server (D2S) while the reverse is also required to establish a trusted and reliable communication channel. Table I provides a

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Table I. Evaluation of related work

Paper no.	Mutual authentication	Content integrity	Privacy theft	Forward secrecy maintenance	Replay attack resistance	DoS resistance	Audit/ traceability	Trust
Chan and Zhou (2014)	No – one way	Not addressed	Not addressed	Not addressed	Guarded against simple tampering using existing	Yes	Yes	One way (server is assured trust on the requesting)
David and Francisco (2014)	No – one way	Not addressed	Not addressed	Not addressed	Not addressed	Not addressed	Yes	Not addressed
Hong *et al.* (2015	No – one way	Partial – integrity of stored data assured, integrity during exchange of credentials not addressed	Yes – hiding of data using one-way hash function	Yes – session parameters and pseudo-random number	assume security by adopting SSH	Not addressed	Yes	One way (server is assured trust on the requesting device)
Cirani *et al.* (2015)	No – one way	Assumed to be achieved with existing suitable EAP versions	Not addressed	Not addressed	Requires user to adopt ways to assure trust of the proxy used (HTTP/CoAP)	No – requires usage of gateways	Yes	Authorization function delegated to third party, requires an assumed trust level
Ning and Yang (2015)	Yes – Chbyshev chaotic maps	Yes – one-way hash	Yes – aggregated proofs	Yes – session parameters and pseudo-random number	Assumed to be addressed as an effect of mutual authentication	Not addressed	No	Yes
Ruj *et al.*, (2014)	No – one way	Integrity of data shared over channel not addressed	Partial – privacy on data stored not user private information	Not addressed	Assume security by adopting SSH	Not addressed	Yes	Yes – Trusted Third Party
Noureddine and Bashroush (2013)	No – one way	Integrity of data shared over channel not addressed	Yes, OAuth	Not addressed	Not addressed	Yes	Yes	Pre-requisite for authentication
Liu *et al.* (2014)	No – one way	Not addressed	Yes – group signatures	Yes – ACV-BGKM	Not effective	Not addressed	Yes – group manager	One way (server is assured trust on the requesting device)

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

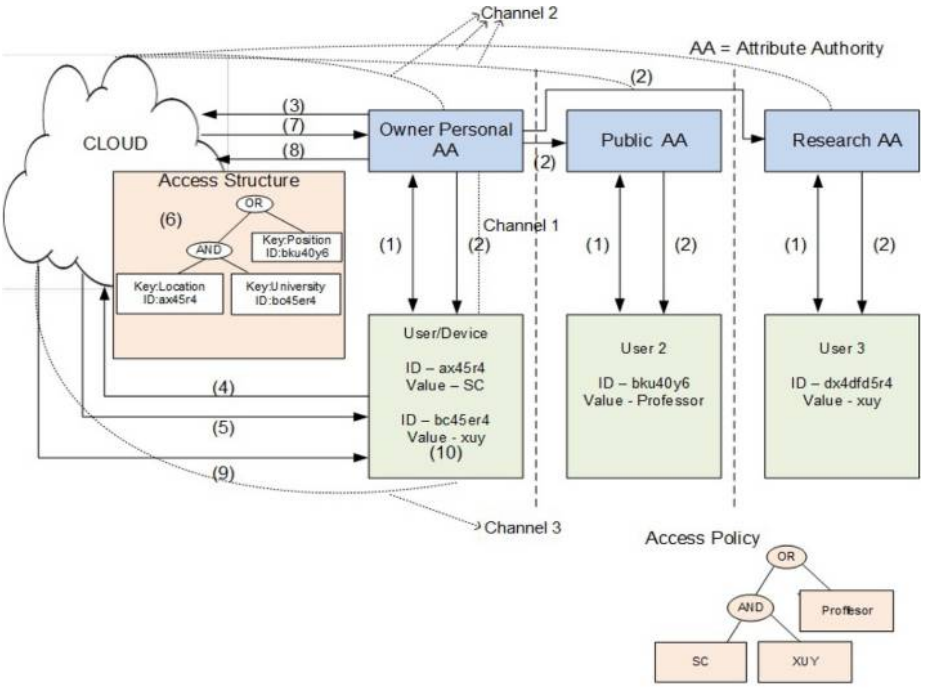
comparative analysis of the security issues addressed in each of the above mentioned systems.

4. Proposed Collaborative Mutual Identity Establishment (CMIE) system

This section introduces us to a system that addresses the security requirements and secured interactions between the entities of the FI, as proposed by Ambritta *et al.* (2014). The algorithm introduced in the system addresses the basic requirements of mutual authentication which enables to ensure trust between the participating entities and also assures that devices security is not compromised.

4.1 System architecture

As shown in Figure 2. the system consists of three entities, namely, the attribute authority (AA) (owner of the data), the public cloud that provides storage and processing power to the miniature devices involved in communication, and the users/devices with their associated sensors. The communication between the entities follows a pattern wherein the access policy lies with the sole owner AA and also the other AAs within the trusted domain of the owner with whom the AA (owner) wishes to share the



Notes: (1) Obtain attributes; (2) provide write and secret keys; (3) outsource encrypted data with access structure; (4) read/write request; (5) access attribute info under user control; (6) map values to access structure; (7) send mapped structure to attribute authority; (8) verify valid users and report to cloud (allow or deny access); (9) authorized users allowed to download; (10) decrypt with secret key

Figure 2.
System architecture

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

access policy, thereby protecting the owners' critical data. The AA requires the cloud and the devices that it is concerned with, to register themselves as an initiation to the participation in the communication depicted as Channel 1 in Figure 2. The AA provides certificates to the entities it is concerned with, which is to be used later to prove their identities to establish trust among the participating entities. The AA uploads the encrypted data and documents tied with the access structure to the public cloud. The cloud only contains the access structure (contains key and ID for the corresponding values in the access policy) so that the cloud knows what attributes to look for in the user/device that requests for access by matching the available IDs mentioned in the structure. The cloud is therefore left unclear about which attribute value is actually the appropriate one to allow access to the data. Its job is to only store the data and collect the required attributes, and forward it to the appropriate AA which takes the decision about allowing/denying access upon comparison with the access policy.

The Collaborative Mutual Identity Establishment (CMIE) scheme that is introduced in the following sections ensures that the interactions between the entities are secured by way of mutual authentication between the entities participating in the communication, thereby ensuring trust, prevention of privacy theft, protection of globally shared data, accountability and prevention of complete compromise of devices/users and access policy. The CMIE algorithm is executed for every communication between the participating entities as described in the following sections.

4.2 CMIE algorithmic details

The various entities involved in the communication in our proposed system, as shown in Figure 2, are:

- AA;
- cloud; and
- user/device.

The communications between these entities are classified into three parts as follows:

(1) *AA (server) –> user/device/cloud (client)* [user/devices register with the AA and obtain keys] represented as Channels 1a and 1b in Figure 2.

(2) *Cloud (server) <–> user/device (client)* [user/devices request access to data on the cloud] represented as Channel 2 in Figure 2.

(3) *AA (server) <–> cloud (client)* [cloud sends the collected and mapped data for verification to the AA] represented as Channel 3 in Figure 2.

Figures 3 and 4 provide higher level view of the communication channels and the security goals addressed by the introduction of the algorithm in each of the channels listed above.

Each of the above mentioned pairs require the execution of authentication process in the CMIE algorithm prior to actual exchange of data. The CMIE algorithm uses the elliptical curve cryptography (ECC) presented by Elaine Brow (2010) in his article, which follows the public key cryptography system based on the defined elliptical curve on finite field (GF[P]). Its main advantage is it is resistance to man-in-the-middle attack due to the difficulty involved for an intruder to determine the points from the defined finite field (GF[P]). Also, to get strength equivalent to a 256 bit symmetric key, a standard

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

asymmetric algorithm will have to use a huge key of size 15,360 bits. Lauter (2004); NSA/CSS (2014) have therefore presented that keys of this size are usually not realistic to use due to the amount of processing power that would be required and, hence, the speed requirement of the operations. However, with elliptic curve algorithms, the equivalent key length is 512 bits, which is entirely sensible and reasonable.

The following algorithm uses a variant of the ECC called the Elliptical Curve Digital Signature Algorithm. A Point P on the curve is selected and made available to all the users/entities publicly. Then, each of the participating users/entities follow the algorithm described below to generate their individual public and private key pairs to be used for further communication and exchange of information.

4.2.1 Issuing of certificates. As an initial step, the AA plays the role of assigning temporary identities and certificates to the cloud and users/devices during the registration phase. This provides the AA with the capability to perform *audits* and *tracing activities* in case of any fraudulence. As shown in Figures 5 and 6, the $Entity_1$ (user/device/cloud) chooses a value $L_{d/c}$ (private to $Entity_1$). Similarly, the $Entity_2$ (AA) chooses a value L_{AA} (private key of AA). Then, the $Entity_1$ calculates its public key $Q_{d/c}$ by using the commonly known Point P and the private value $L_{d/c}$. The public key $Q_{d/c}$ is sent to the AA through a secure and authenticated channel. The AA then chooses a unique temporary identity for the device, performs a hash of the identity and a timestamp (certificate expiration time) and signs it using its private key and sends it to the $Entity_1$ over a secure channel. This signed message is the certificate which is then used for further authentication and key generation in the communication between

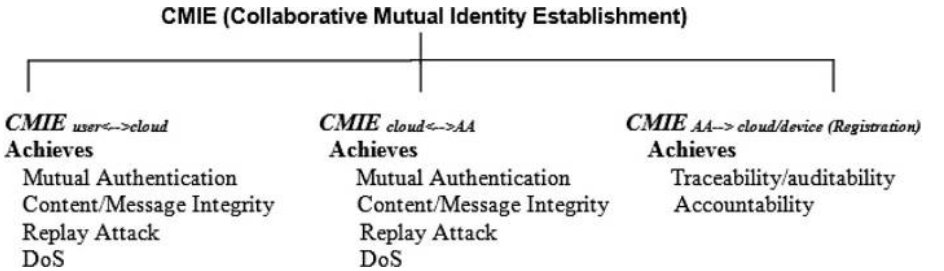


Figure 3.
CMIE algorithm and addressed security goals

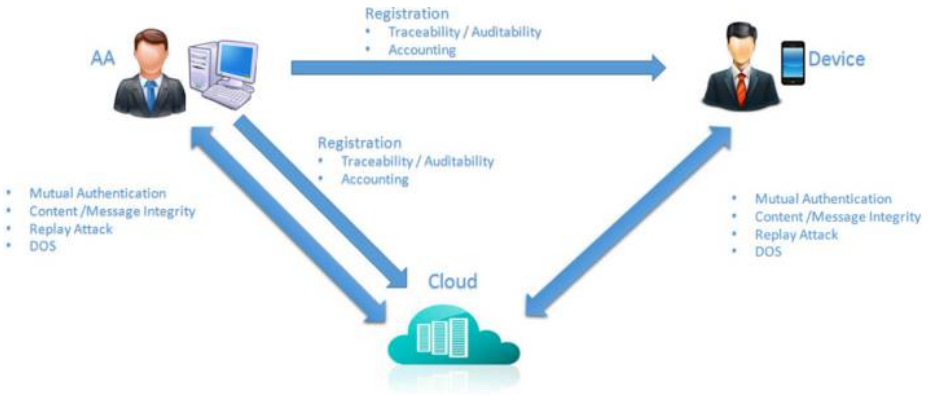


Figure 4.
Communication channels between entities and security goals achieved

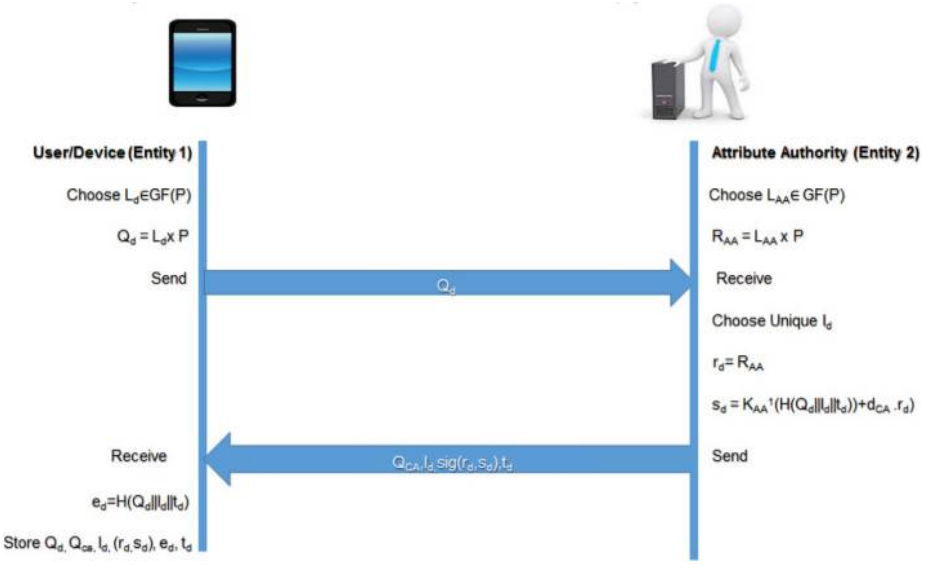


Figure 5.
Certificate and identity distribution to user/devices

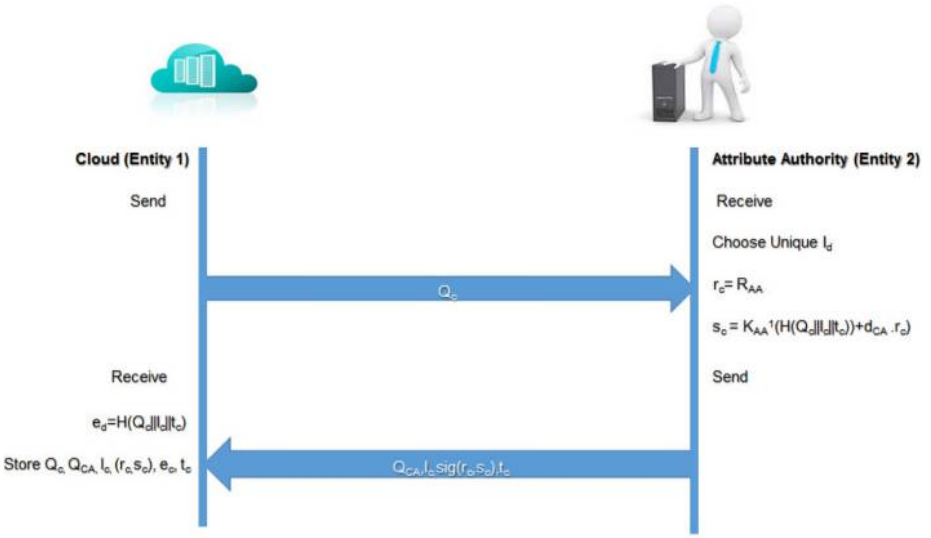


Figure 6.
Certificate and identity distribution to cloud

participating entities. The certificate consists of integers, namely, $r_{d/c}$ and $s_{d/c}$ which as a pair form the signature of the concerned entity. The subscript *d* concerns the users/devices and the subscript c concerns the cloud.

4.2.2 Mutual authentication. Mutual authentication and exchange of messages over established secure communication link is achieved by adopting the algorithm in Figure 7 (communication between user/device [Client/$Entity_1$] and the cloud (Server/$Entity_2$]) and Figure 8 (communication between cloud [Client/$Entity_1$] and the AA

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

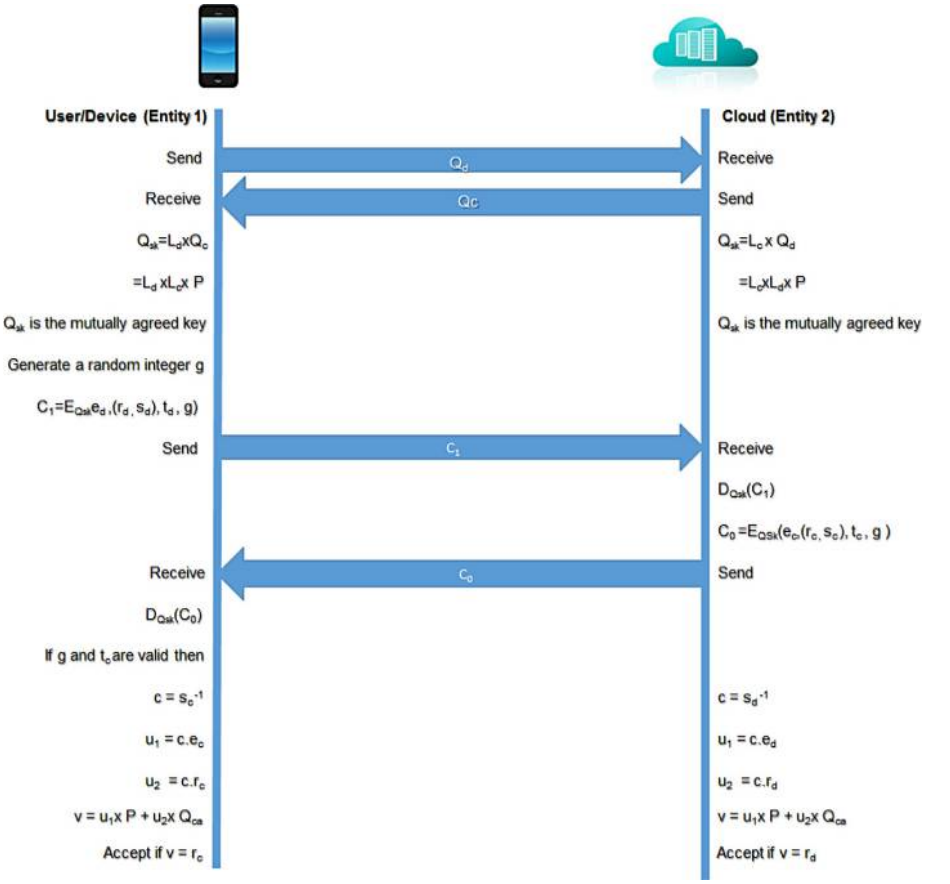


Figure 7.
Mutual authentication (user/device [Client/Entity$_1$] and the cloud [Server/Entity$_2$])

[server/Entity$_2$]) in reference to the work done by Mahalle *et al.* (2013). The algorithm follows that whenever there is a request initiated by one of the entities in the system (AA, devices/users and the cloud), they immediately exchange keys following the ECCDH (elliptical curve cryptography Diffie Hellman key exchange algorithm).

Initially, the entities exchange their public keys (Q_d, Q_c) (in case of a communication between the user/devices[client] and the cloud [server]) or (Q_c, Q_{AA}) (in case of a communication between the cloud [client] and the AA [server]) generated by multiplying a chosen random integer (L_d and L_c) or (L_c and L_{AA}), respectively, from the finite field with the publicly distributed value P from the finite field GF(P). Upon receiving each other's private keys, the entities work upon generating a shared secret key (Q_{sk}) by multiplying the other party's public key with its own private key.

The entities participating in the communication then exchange their certificates for verification. Therefore, considering the case where communication takes place between the user/device (client referred to as Entity$_1$) and the cloud (server referred to as Entity$_2$), as in Figure 7, Entity$_1$ encrypts the concatenation of its certificate ($e_d \parallel [r_d, s_d]$), the certificate expiration time t_d and a random number g (to maintain the freshness of the

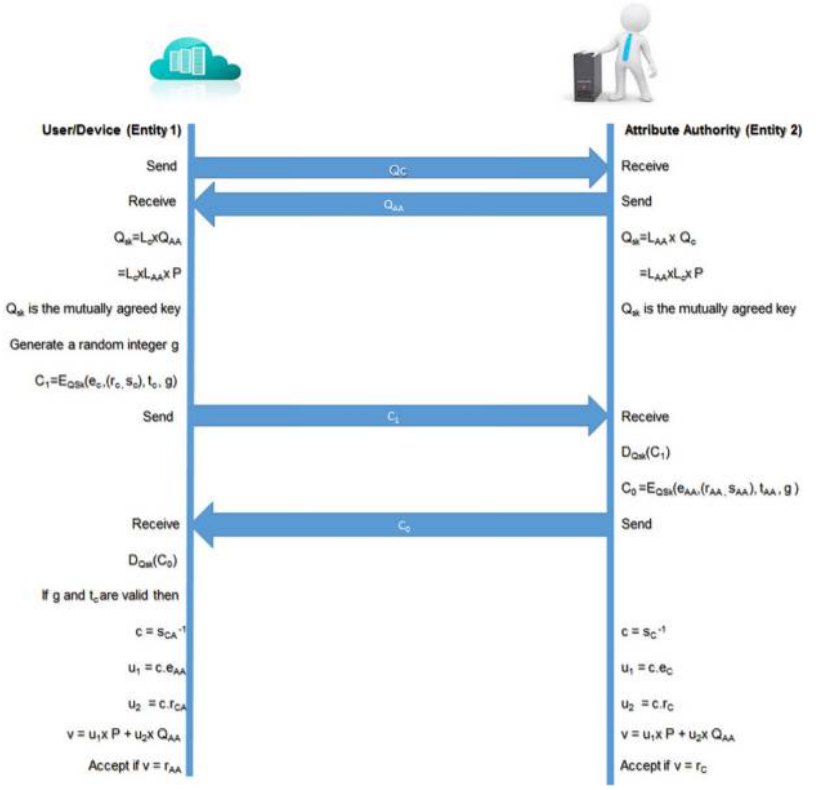


Figure 8.
Mutual authentication (cloud [Client/Entity$_1$]) and the attribute authority (Server/Entity$_2$)

transmitted data) with the shared secret key Q_{sk}, labeled as C_1 and sends it to Entity$_2$. Upon receiving the certificate, Entity$_2$ decrypts the certificate and checks for the validity of the certificate.

If the read values of g and the certificate expiration dates t_d and t_c are valid then it proceeds further toward verifying the signature of the certificate received from Entity$_1$. If the values are invalid, the connection is aborted. If verified to be valid, Entity$_2$ then similarly encrypts the concatenation of its certificate ($e_c \| (r_c, s_c)$), the certification expiration date t_c and the same random number *g* received from Entity$_1$ with the shared secret key Q_{sk}, labeled as C_0 and sends it to Entity$_1$ for verification. Entity$_1$ follows the same verification procedure and determines the validity of the certificate.

Similarly, considering the case where communication takes place between the cloud (client referred to as Entity$_1$) and the AA (server referred to as Entity$_2$), as in Figure 8, the same sequence of operations are followed as mentioned in the previous paragraph with e_c, (r_c,s_c) representing the certificate of Entity$_1$ and t_c representing the certificate expiration time of Entity$_1$. Similarly, e_{AA}, (r_{AA}, s_{AA}) representing the certificate of Entity$_2$ and t_{AA} representing the certificate expiration time of Entity$_2$. Q_{sk} represents the shared secret key, C_0 represents AA's (Entity$_2$'s) own certificate and C_1 represents entity1's certificate. K_{sk} again represents the session shared secret key used for further communication.

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

4.2.3 Content integrity and replay attack resistance. Once the signatures have been verified for *mutual authenticity*, we add another key exchange step to generate a shared secret key K_{sk} that is to be used for each session. The shared secret key K_{sk} is generated by adding the random number *g* to the shared secret key (Q_{sk}) generated that were used during the mutual authentication phase. This idea of session establishment using session keys helps to handle the DoS attack by regulating access to the stored data/resources to one request associated with one ID (i.e. one device) per session, thereby preventing any malicious device/user flooding the system with requests leading to DoS to intended users/devices.

After the shared session key has been created, any member/entity that needs to send a message to the other entity follows the "encrypt then MAC" mechanism, as specified by Colin Percival (2009) to ensure *integrity*, *authenticity* and *security of the content/message* that is exchanged. Considering the communication between the user/device (Client referred to as $Entity_1$) and the cloud (server referred to as $Entity_2$) in Figure 9, $Entity_1$ first encrypts the message (concatenation of device identity [I_d] and attributes) to be sent using Q_{sk} and further performs a keyed hash-based MAC of the encrypted message using the shared session key K_{sk}, which is sent to $Entity_2$. On the receiver side ($Entity_2$), the HMAC of the received encrypted message is recalculated to check for *integrity* and *authenticity*. If the same HMAC is generated as the one that was received, then the message is accepted. Similarly, considering the case where communication takes place between the cloud (Client referred to as $Entity_1$) and the AA (server referred to as $Entity_2$), as in Figure 10, the same sequence of operations as mentioned above are performed following which the response (concatenation of device identity [I_d] and access response [grant/deny access]) from $Entity_2$ is encrypted using Q_{sk} and further a keyed hash-based MAC of the encrypted message is computed using the shared session key K_{sk}, which is sent

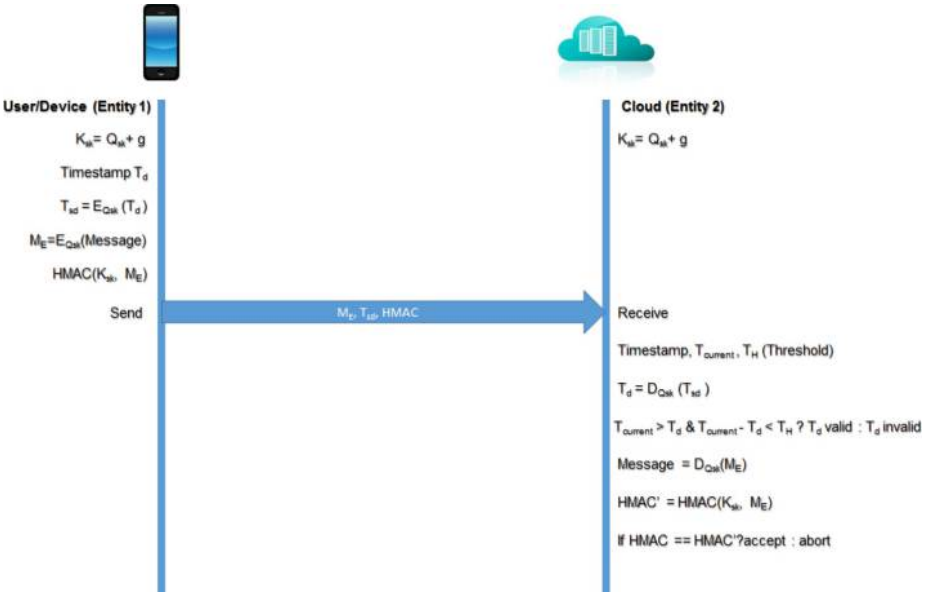


Figure 9.
Content integrity and reply attack resistance (user/device [Client/$Entity_1$] and the cloud [Server/$Entity_2$])

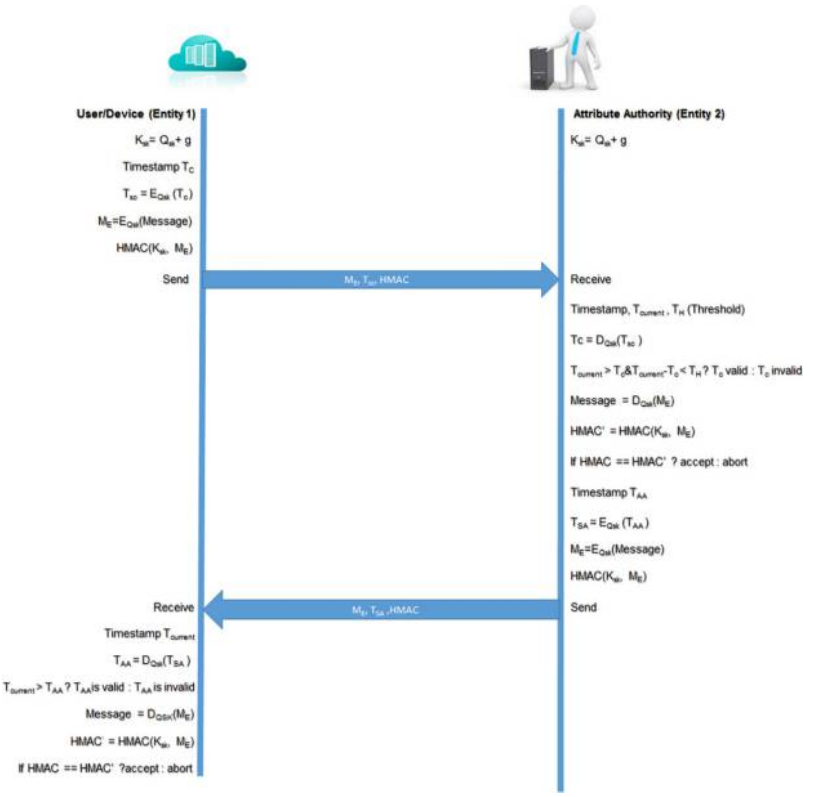


Figure 10.
Content integrity and reply attack resistance (cloud [Client/Entity$_1$] and the AA [Server/Entity$_2$])

to Entity$_1$. On the receiver side (Entity$_1$), the HMAC of the received encrypted message is recalculated to check for integrity and authenticity. If the same HMAC is generated as the one that was received, then the message is accepted. The inclusion of timestamp T_{AA} (for AA), T_c (for cloud) and T_d (for user/device) helps to prevent replay attacks by verification against the $T_{current}$ (current time) on the receiver side. The message is accepted for further processing only if the difference in time does not exceed the T_H (threshold time), i.e.

$$T_{current} - T_d < T_H.$$

Additionally, considering the case where the AA (owner of the data) wants to put his data over the cloud, the data are kept safe by performing the "encrypt then MAC" mechanism in which the keys used for encryption and hashing is private to the AA and not made available to the cloud. The counter decryption key is made available only to the legitimate users by the AA itself during the initial registration phase or at the time of downloading the requested data. Thus, the algorithm on the whole addresses the mutual authentication to establish a secured communication channel, man-in-the-middle attack (replay attack), DoS and content/message integrity. A man-in-the-middle attack can succeed only when the attacker can impersonate each endpoint to the

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

satisfaction of the other, which in this algorithm cannot happen because the certificate validity has to be verified on both sides prior to exchange of data. The proposed algorithm also maintains *forward secrecy* by maintaining session variables (session key generated newly for every communication) along with random numbers to maintain freshness and randomization of sessions.

5. Results and discussion

This section provides an evaluation of the CMIE algorithm described in Section 4.2 against the Automated Validation of Internet Security Protocols and Applications (AVISPA, 2015) tool, details of which have been explained by A. Armando *et al.* (2005) based upon the Dolev–Yao intruder model to verify that the security solutions that are claimed to be addressed are effectively achieved in reality. The AVISPA tool, which is also available as a Web interface, is a push-button tool that provides a modular and simple formal language for specifying protocols and their security properties. It also associates itself with various back ends that provide a platform for automatic protocol analysis.

5.1 CMIE verification

The algorithm described in Section 4.2 has been represented as a sequence of actions in Figure 11 to help the reader understand the evaluation procedure that follows in this section. The algorithm has been evaluated in two phases. The first phase includes the verification procedure for mutual authentication and the second phase for the verification of resistance against replay attack.

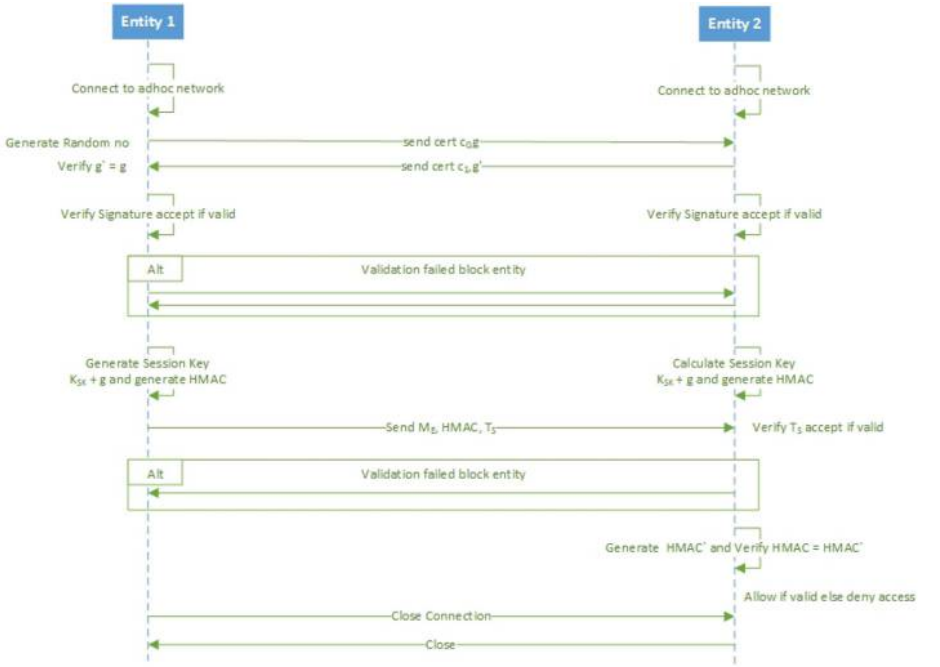


Figure 11.
CMIE algorithm sequence diagram

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

5.1.1 Evaluation procedure. Assumption: The system has exchanged shared secret keys using ECCDH key exchange algorithm. Also, according to the Dolev–Yao model, the intruder is assumed to have exposure to the device ID and the hash function that carries information that provides access control [i.e. ID ‖ Attributes].

5.1.1.1 Mutual authentication.

$$E_1\, E_2 \colon C_1; [sig_{E1},\ t_{E1},\ g]\_Q_{SK}$$

$$E_2\, E_1 \colon C_2; [sig_{E2},\ t_{E2},\ g]\_Q_{SK}$$

where,

E_1 = Entity 1 (Cloud | AA);
E_2 = Entity 2 (Cloud | device);
sig_{E1}, sig_{E2} = signature of E1 and E2, respectively;
T_{E1}, t_{E2} = timestamp for E1 and E2, respectively;
G = random number;
Q_{SK} = shared secret key; and
_ (underscore) = encryption.

5.1.1.2 Exchange of information (message/content integrity and prevention of replay attack). Generate a session key $K_{SK} = Q_{SK} + g$:

$$E_1\, E_2 \colon [M_E, T_{sd}, HMAC]; [Msg]\_Q_{SK},\ [T_u]\_Q_{SK},\ HMAC\,[K_{SK} \| ID \| \text{ Attributes}]$$

$$E_2\, E_1 \colon [M_E, T_{sd}, HMAC]; [Msg]\_Q_{SK},\ [T_u]\_Q_{SK},\ HMAC\,[K_{SK} \| Response]$$

where,

E_1 = Entity 1 (Cloud | AA);
E_2 = Entity 2 (Cloud | device);
Msg = attributes and ID from devices and response from cloud/AA; and
T_u = timestamp of E1 and E2, respectively.

- *Mutual authentication*: Achieved by securely sharing secret key Q_{SK} and the random number *g* helps to prevent any forgery, thereby maintaining freshness. The certificates (C_0 and C_1) in Figure 11 that are exchanged for verification are encrypted using the shared secret key Q_{SK}, known only to the participating entities. Upon successful verification and mutual authentication is achieved. The verification of the generated random number during the exchange of the certificates helps in preventing any forgery, thereby providing a means for mutual authentication. Verification results show that secure mutual authentication is achieved.
- AVISPA evaluation – mutual authentication:

```
% OFMC
% Version of 2006/02/13
SUMMARY
  UNSAFE
DETAILS
  ATTACK_FOUND
```

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

```
PROTOCOL
  /home/avispa/web-interface-computation/zmd/tempdir/workfileHXnsdn.if
GOAL
  authentication_on_sk2
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.22s
  visitedNodes: 43 nodes
  depth: 3 plies
ATTACK TRACE
i → (a, 6): start |
(a, 6) → i: SA1 (1).exp(g, DHX (1)).Ni (1)
i → (b, 3): SA1 (1).exp (g, DHX (1)).Ni (1)
(b, 3) → i: SA1 (1).exp(g, DHY(2)).Nr (2)
i → (a, 6): SA1 (1).exp (g, DHY(2)).Nr(2)
(a, 6) → i: {a.{SA1 (1).exp(g, DHX (1)).Ni (1).Nr(2)}_inv(ka).SA2(3)}_(f(Ni (1).Nr(2).SA1
(1).exp(exp(g, DHY(2)), DHX (1))))
i → (b, 3): {a.{SA1 (1).exp(g, DHX (1)).Ni (1).Nr(2)}_inv(ka).SA2(3)}_(f(Ni (1).Nr(2).SA1
(1).exp(exp(g, DHX (1)), DHY (2))))
(b, 3) → i: {b. {SA1 (1).exp(g, DHY(2)).Nr (2).Ni (1)}_inv(kb).SA2(3)}_(f(Ni (1).Nr(2).SA1
(1).exp(exp(g, DHX (1)), DHY (2))))
```

The attack trace, as mentioned above, shows the intervention of an intruder gaining access to the information that is exchanged. However, as we use ECC domain parameters for the exchange of information and establishment of shared secret keys which is resistant to man-in-middle attack by its very nature, the attack trace can be neglected:

- *Man-in-middle attack*: Considering the exchange of access control information, i.e. M_E, T_{sd}, HMAC, the information is not revealed to the attacker despite his attempts to eavesdrop the channel due to the encrypted exchange. If an attacker manages to obtain the attributes that gives access to devices, the masquerade attack is prevented by using an ID to validate the correct device, i.e. HMAC [K_{SK}‖ ID‖ Attributes]. The ID of the authentic device is prevented from being stolen by the intruder by applying public key cryptography to ID during the exchange of information. In this case, according to the algorithm, we ensure that the authentication process has been done before access control.
- *Replay attack*: In an unprotected communication, an adversary can intercept the message sent out from an entity ($Entity_1$). However, it is not possible in our system because the algorithm is designed to detect any intrusion by verifying timestamp $T_{U/D}$. If the $T_{current} - T_{D/U}$ is older than the predefined threshold value T_H, the message is considered invalid and has been held back by an unintended user or tampered. If $T_{U/D}$ is invalid, HMAC is not valid and consistent, thereby helping to maintain the integrity of the content that is shared over the channel. Figure 12 shows that the output window that proves that the algorithm is resistant (safe) toward replay attacks when validated against AVISPA.

Figure 12.
AVISPA evaluation – replay attack

- *DoS attack*: DoS attack happens when an attacker accesses a particular resource (data in cloud/device) enormously by using different identities/devices, thereby flooding the system with pending requests which leads to the negligence of legitimate requests by the server/device. However, this is controlled in our system by introducing the idea of session establishment and using session keys (K_{SK}) to handle the DoS attack by regulating access to the stored data/resources to one request associated with one ID (i.e. one device) per session. This helps in preventing any malicious device/user flooding the system with requests leading to DoS to intended users/devices.

Considering the IoT scenario, another important aspect that is the device–location (D2L) aspect has not been considered in this protocol. Major focus of the protocol is centered around the device-to-device (D2D) and device-to-server (D-2-S) scenarios. Also, IoT basically works upon a confluence of hundreds for protocols that supports the achievement of various factors in the IoT, for example Data Distribution Service, Message Queue Telemetry Transport, Extensible Messaging and Presence Protocol, CoAP and so on. Interoperability of the proposed CMIE algorithm with the existing protocols has to be considered to establish a complete model that fits the FI. Further, each request for mutual authentication requires a querying of the database and a computation at each of the participating entities side for verification which could take considerable amount of time. However, for applications that require firm authentication for maintaining and ensuring secure interactions between entities prior to access control and initiation of actual transfer of sensitive information, the negligible difference in computation time can be ignored for the greater benefit that comes with stronger security. Other factors such as quality of service (QoS) (i.e. flexibility of data delivery, resource usage and timing), key management and distribution also need to be considered. However, the user still has the responsibility to choose the required protocol that suits one's application and serves the purpose.

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

6. Conclusions and future work

An attack-resilient protocol that manages the communication between devices in the IoT is mandatory to address the security issues that prevail. This paper presents an effective CMIE protocol that provides mutual authentication, secured interactions between participating devices, forward secrecy by way of session operators (session keys) and pseudo-random numbers, and also address DoS and replay attack and message/content integrity. The algorithm helps maintain the entity's potential to perform audits and accounting and ensures trust between participating entities. The protocol is also evaluated against AVISPA to ensure that the algorithm is resistant toward DoS, replay attack and man-in-the-middle attacks. The future plan is to extend this protocol to suite the auto-delegation and revocation of access rights in the system with proper security evaluations. Also, interoperability of the devised protocol with existing protocols in the IoT is an area that needs focus in future.

References

Ambritta, N.P., Railkar, P.N. and Mahalle, P.N. (2014), "Proposed identity and access management in future internet (IAMFI): a behavioral modeling approach", *Journal of ICT*, Vol. 2 No. 1, pp. 1-36.

Armando, A., Basin, A., Boichut, Y., Chevalier, Y., Compagna, L., Cuellar, J., Hankes Drielsma, P., He'am, P.C., Kouchnarenko, O., Mantovani, J., Modersheim, S., von Oheimb, D., Rusinowitch, M., Santiago, J., Turuani, M., Vigan'o, L. and Vigneron, L. (2005), *Springer-Verlag Berlin Heidelberg*. The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications, pp. 281-285.

AVISPA Web Interface (2015), "The Avispa Project", available at: www.avispa-project.org/web-interface/basic.php (accessed 13 September 2015).

Chan, A.C.-F. and Zhou, J. (2014), "Cyber–physical device authentication for the smart grid electric vehicle ecosystem", *IEEE Journal on Selected Areas in Communications*, Vol. 3 No. 7, pp. 1509-1517.

Cirani, S., Picone, M., Gonizzi, P., Veltri, L. and Ferrari, G. (2015), "IoT-OAS: an OAuth-based authorization service architecture for secure services in IoT scenarios", *IEEE Sensors Journal*, Vol. 15 No. 2., pp. 1224-1234.

Colin Percival (2009), "Daemonic dispatches", available at: www.daemonology.net/blog/2009-06-24-encrypt-then-mac.html (accessed 13 September 2015).

David, F.-L. and Francisco, J.L. (2014), "Authentication, authorization, and accounting in WebRTC PaaS infrastructures", *IEEE Internet Computing*, Vol. 18 No. 6 pp. 34-40.

EIT Digital (2015), "Innovation and entrepreneurship", available at: www.eitictlabs.eu/innovation-entrepreneurship/cyber-physical-systems/ (accessed 13 September 2015).

Elaine Brow (2010), *Algebraic Geometry*, Elliptical Curve Cryptography, Math 189A.

Hong, L., Huansheng, N., Quigxu, X. and Yang, L.T. (2015), "Shared authority based privacy-preserving authentication protocol in cloud computing", *IEEE Transactions On Parallel and Distributed Systems*, Vol. 26 No. 1, pp. 241-251.

Pan, J., Paul, S. and Jain, R. (2011), "A survey of the research on future internet architectures", *IEEE Communications Magazine*, Washington University, 0163-6804/11.

Lauter, K. (2004), "The advantages of elliptical curve cryptography for wireless security", *IEEE Wireless Communications*.

Liu, X., Zhang, Y., Wang, B. and Yan, J. (2014), "Mona: secure multi- owner data sharing for dynamic groups in the cloud", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 24 No. 6, pp. 1182-1191.

Noureddine, M. and Bashroush, R. (2013), "An authentication model towards cloud federation in the enterprise", *The Journal of Systems and Software*, Vol. 86 No. 9, pp. 2269-2275.

Ning, H. and Yang, L.T. (2015), "Aggregated-proof based hierarchical authentication scheme for internet of things", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26 No. 3, pp. 657-667.

NSA/CSS (2014), available at: www.nsa.gov/business/programs/ellipticcurve.shtml (accessed 13 September 2015).

Mahalle, P.N., Anggorojati, B., Prasad, N.R. and Prasad, R. (2013), "Identity authentication and capability based access control (IACAC) for the internet of things", *Journal of Cyber Security and Mobility*, Vol. 1 No. 1 pp. 309-348.

Ruj, S., Stojmenovic, M. and Nayak, A. (2014), "Decentralized access control with anonymous authentication of data stored in clouds", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25 No. 2, pp. 384-394.

SINTEF (2015), "Internet of things and cyber physical system", available at: www.sintef.no/home/Information-and-Communication-Technology-ICT/Departments/Communication-systems/InternetOfThings/ (accessed 13 September 2015).

Corresponding author
Nancy Ambritta P. can be contacted at: nancy.ambritta@yahoo.com

Downloaded by TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES At 22:33 07 November 2016 (PT)

For instructions on how to order reprints of this article, please visit our website:
www.emeraldgrouppublishing.com/licensing/reprints.htm
Or contact us for further details: **permissions@emeraldinsight.com**