Development of a secure framework for an unified authentication mechanism using elliptic curve cryptography*

Anirudh Srivathsan[1*], Ayush Ambastha[2], Sunil I Dodamani[3] & Rajasekar Mohan[4]

PES University, Bengaluru

**Presented in 1st National Conference on Smart Manufacturing & Industry 4.0 (NCSMI4) at Central Manufacturing Technology Institute (CMTI), Bengaluru, during May 30-31, 2019.*

ABSTRACT

Keywords:
Smart Card,
Framework,
Elliptical Curve Cryptography,
Security

Smart Cards are used all over the world to provide personal identification, use services and store information. Since they are usually the single point of authentication that organizations have, extensive research is done to strengthen its security and protect sensitive data such as User credentials from malicious attackers.

This paper primarily deals with the development of a secure framework for availing cloud based services and mitigates the disadvantages of its predecessors by providing a more secure way of transferring data across open channels.

The framework uses a lightweight cryptography algorithm, namely Elliptic Curve Cryptography (ECC) which helps users transfer data securely. ECC provides stronger security when compared to other popular cryptography methods and requires a smaller key size, which is convenient for resource-constrained devices.

1. Introduction

The Internet of Things (IoT) is a contemporary technology, which connects various physical objects that communicate and transfer data among each other over the internet. They can also be remotely monitored and controlled without the need of human-to-human or human-to-computer interaction.

IoT devices are generally resource constrained, so using normal encryption methods will lead to inefficient systems and lead to unnecessary overhead because of the large key size and high processing requirements.

Lightweight cryptography is an encryption method tailored for implementation in resource constrained environments which include components such as RFID tags, sensors, contactless smart cards and so on. In such environments, small key cryptography techniques such as Elliptical Curve Cryptography (ECC) are convenient and is used for end to end data transmission.

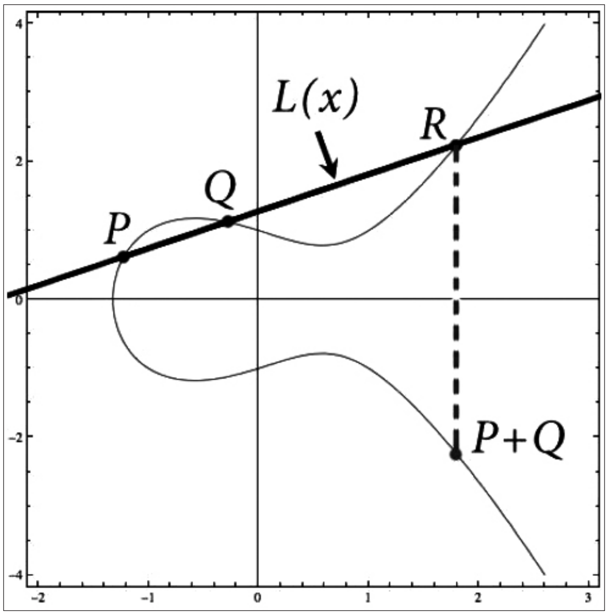


Fig. 1. Basic addition operation on an elliptical curve, used by ECC.

*Corresponding author,
E-mail: poruranirudh@gmail.com(Anirudh Srivathsan)

Using Smart Cards to access cloud based services and for authentication isn't a contemporary technology but started as early as the 1960s. But what has happened because of the onset of this is that all cloud based service providers have their own smart card with their desired level of security. Having so many different cards for different services effectively go against it's purpose, which is User Convenience.

People around the world use Smart Cards for different applications such as mobile communications, banking and financial services, passport, identity cards, transportation, health care services, metro services, electricity payments etc.[2]

The different types of smart cards that are in use today are contact, contactless, dual interface and multi-component cards. The two major types that we'll be referring to in this paper are contact and contactless cards. Contact cards need a physical medium of communication and are widely used for a variety of applications today such as credit cards, metro cards etc. Contactless cards, on the other hand, use a radio interface to make payments and are not that popular due to the vulnerabilities associated with sending data over the open-channel (Internet). Though the contactless cards make the identification process faster, it makes it easier for enterprise hackers to steal information using hidden scanners. Therefore, encrypting the data being sent across is of prime importance and is one of the focal points of this paper.

This paper is organized as follows. Section II mentions the related works in this domain, Section III presents our contribution and the improved framework, Section IV gives an example scenario and Section V concludes the paper.

2. Related Works

There has been extensive research on developing frameworks for secure end to end data transmission using smart cards. Some of the noticeable ones are as mentioned below.

'Authentication and Delegation using Smart Cards' by M. Abadi, M. Burrows, C Kaufman, B Lampson in 1993, was one of the first few articles on using smart card plus PIN (Personal Identification Number) to ensure secure access to ATMs and other systems. They discuss about public-key smart-card protocols and analyze their various assumptions and guarantees.

'Efficient password authenticated key agreement using smart cards' by Wen-Shenq Juang in 2004, had a nonce-based user authentication and key agreement scheme with lesser computational complexity and more functionality. It utilized the concept of a session key which the User and server would agree on and also gave the user freedom to choose their own password.

'An efficient biometrics-based remote user authentication scheme using smart cards' by Chun-Ta Li and Min-Shiang Hwang in 2010, used one-way hash functions, bio-metric templates and smart cards in unison to provide mutual authentication between two entities. They also used random numbers instead of time-stamps to resist replay attacks.

"Elliptical Curve Cryptography based Security Framework for Internet of Things (IoT) Enabled Smart Card"[1] is an IEEE publication which this paper intends to work upon. In the paper, the authors suggest a framework which helps to transfer data securely from a device to another using ECC and Bio-metrics, and allows users to access multiple cloud based services in one portal. The framework suggested is an efficient method of providing mutual authentication while keeping the computational complexity low. It also provides an additional layer of security by using OTP and Bio-metric samples together. But, it has a few disadvantages that we are overcoming using the framework suggested in this paper.

The drawbacks of using the previous framework are-

1. If the attacker gets the OTP generated by the GSMS, the entire framework is compromised.
2. Flooding the GSMS is possible. If an attacker gets the smart card and sends requests to the GSMS continuously, it can cause Denial Of Service (DoS).
3. For using multiple services, multiple sessions need to be initiated. Which increases latency between using different applications as the entire authentication mechanism needs to take place all over again.
4. As bio-metrics are being sent, large amounts of latency is involved.
5. The framework doesn't mention where the ECC encryptions and decryptions take place.

3. Our Contributions and Proposed Framework

This section talks about the environment in which the IoT enabled smart card will be incorporated. The proposed framework using smart cards involves ECC based encryption and decryption for secure transfer of date between devices and reduces the vulnerabilities associated with traditional contactless IoT devices.

3.1. Development of the ecosystem for IoT based network

The environment in which the proposed framework is used, consists of a Intelligent Smart Card Reader (ISR), Global Service Management System (GSMS) and the union of Service Providers. These devices are capable of Machine to Machine (M2M) communications, collectively forming an IoT network.

The ISR, present in the point of service, facilitates user authentication by sending the UID to the GSMS.

The GSMS is a cloud storage server which performs two duties. It is responsible for maintaining a database of all UIDs present, which helps in user authentication. Also, it acts as an intermediator between the user and the service providers by forwarding the requests/responses between them.

The Service Providers is a collection of all services that have registered with the GSMS and can be availed by the users using the smart card.

The smart card is designed as per the ISO/IEC 14443/7816 standard and is IoT-enabled [3]. It facilitates authentication as it sends the UID to the GSMS using ECC to verify the user credentials.

The situation in which the Smart Card is deployed is shown in Figures 2-5.

3.2. Proposed framework

In this framework, Both the users as well as the service providers initially register with the Global Secure Management System (GSMS). The users provide the GSMS with credentials such as name, date of birth and mobile number along with the users' mobile device credentials. The GSMS verifies the information provided by the user and stores the details in its database following which the users receive a Unique Identification (UID) placed on a Smart Card. The service providers register with the GSMS and similarly, receive a Service Identification (SID) number (Fig.2).

The user places the Smart card on the ISR which carries out mutual authentication. The ISR reads the UID number from the card and sends it to the GSMS using Elliptical Curve Cryptography (ECC)[5]. If the UID sent matches the one previously stored on the GSMS database during the registration process, A 'success message' is displayed indicating that a limited-time session has been activated for the user. If not, No Operation takes place and an error message is sent to the user (Fig.3).

On the registered mobile device, The user then opens the Application to access the services and waits for the it to check whether a session is active or not. Once that is done the App requests

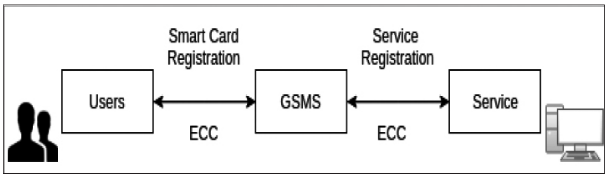


Fig. 2. Registration of the users and service providers

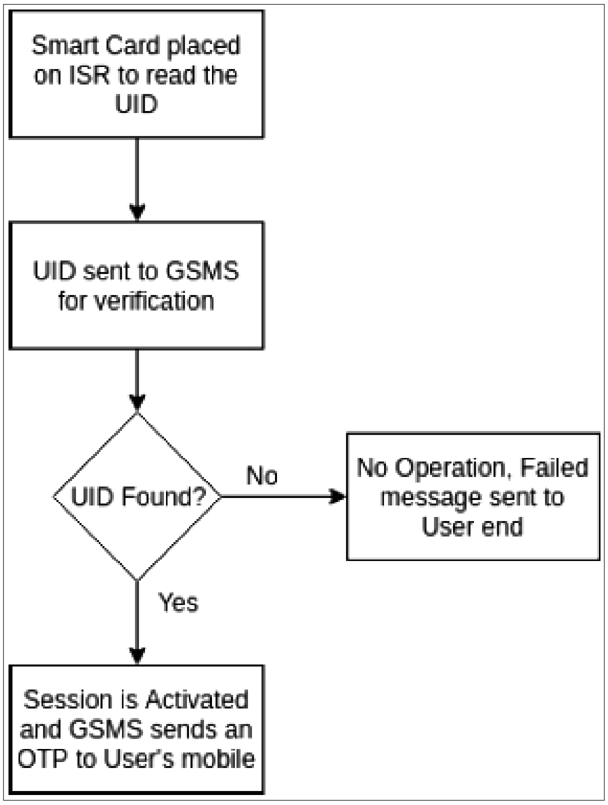

Smart Card placed on ISR to read the UID
UID sent to GSMS for verification
UID Found?
No
No Operation, Failed message sent to User end
Yes
Session is Activated and GSMS sends an OTP to User's mobile

Fig. 3. Session activation.

for the OTP which is sent to the user's mobile. The OTP is then encrypted and sent to the GSMS for verification. If the OTP is correct, a list of registered services are displayed on the UI of the application. The User can then use the required services as long as the session is active (Fig.4).

The GSMS now acts as an intermediator between the service provider and the user, helping the secure transfer of data between the two entities.

There is no direct communication between the user and the service provider which provides abstraction to the cloud-based service providers.

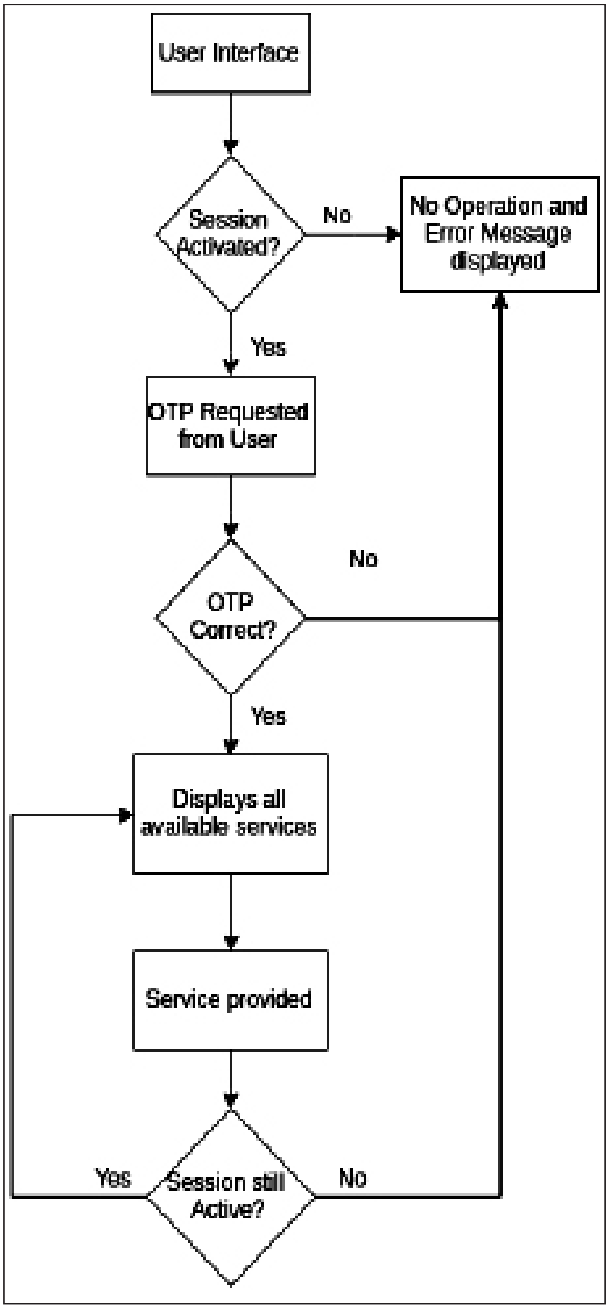


Fig. 4. Availing services at the user interface.

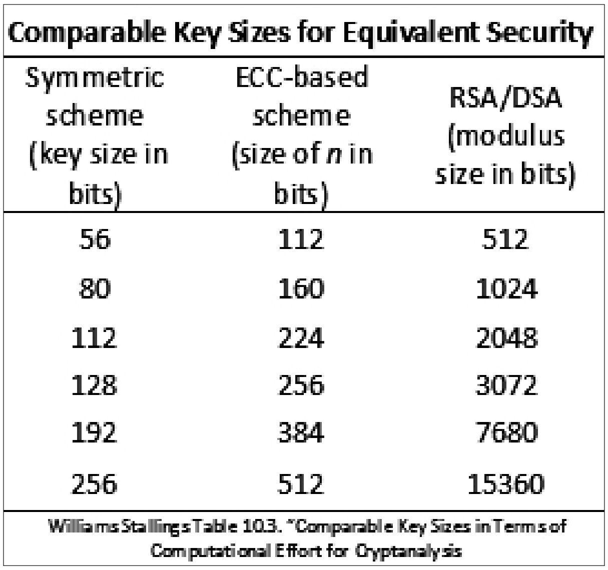
3.3. Elliptic curve cryptography (ECC)

ECC is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. It uses predefined constants known as 'Domain Parameters' which are known by all the devices taking part in the communication.

It is a convenient encryption method for resource constrained devices, typically used in the IoT environments. The performance of this algorithm is mainly intertwined with the efficiency of its scalar multiplication algorithm [6].

Encryption strength is related to the difficulty in discovering the key and is often described in terms of the size of the key (bits) used to perform the encryption. In general, longer keys provide stronger encryption. Different ciphers and encryption algorithms may require different key lengths to achieve the same level of encryption strength. Since it is tough to manage keys with large key sizes, smaller keys which provide the same level of encryption strength are preferred. As shown in Fig. 5.

ECC is an efficient method to deal with both Smaller and Larger Key sizes when compared to RSA and DSA (Digital Signature Algorithm) for the same level of security. [7]



Comparable Key Sizes for Equivalent Security

Symmetric scheme (key size in bits)	ECC-based scheme (size of *n* in bits)	RSA/DSA (modulus size in bits)
56	112	512
80	160	1024
112	224	2048
128	256	3072
192	384	7680
256	512	15360

Williams Stallings Table 10.3. "Comparable Key Sizes in Terms of Computational Effort for Cryptanalysis

Fig. 5. Comparison of key sizes for popular encryption standards.

The mathematical operations of ECC is defined over the elliptic curve E defined by the equation -

$y2 = x3 + ax + b$ ....(1)

Each value of the ‘a’ and ‘b’ gives a different elliptic curve.

Assume the communication is done between two nodes, node A and node B with private keys D and E respectively and G is the Generator point on the curve.

The public keys of nodes A and B are P and Q respectively, given by -

$P = D * G$ ....(2)

$Q = E * G$ ....(3)

Let node A be the transmitting node, it encrypts the data with the Symmetric key given by the product of its private key, D, and node B's public key, Q, such that

$Sa = D * Q$ ....(4)

On the receiver side, node B calculates the same Symmetric key by multiplying its private key to the public key of node A, which it then uses to decrypt the message. The arithmetic behind this is explained in equations (5) and (6).

$Sb = E * P = E * (D * G)$ ....(5)

$Sb = D * (E * G) = D * Q = Sa$ ....(6)

To make encryption and decryption more feasible both sender and the receiver should know and agree upon the table defined with the chosen Elliptic Curve and a generator point G. Thus, this serves as a secure method of information transfer taking advantage of the Discrete Logarithm Problem.

4. Advantages of Proposed Framework

1) Compared to existing database systems, GSMS has on a structured database which acts as a platform for authentication as well.
2) For retail services, this approach is cost effective as there is less maintenance.
3) This framework is easily scalable using concepts such as mobile database and fog servers.
4) Investments made by service providers for their own local servers and authentication mechanism are minimized.
5) Flooding the GSMS cannot harm the system or be used for malicious activities.
6) Multiple services can be availed in one session through the GSMS.

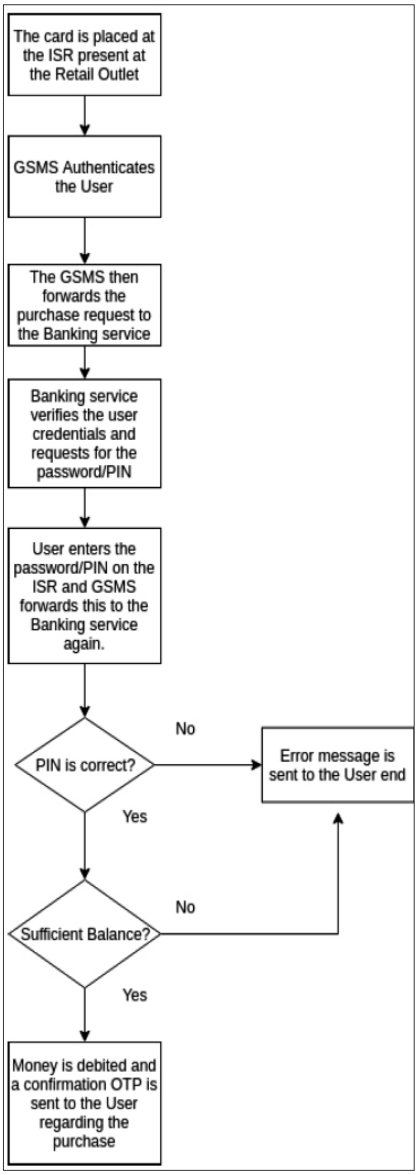

The card is placed at the ISR present at the Retail Outlet
GSMS Authenticates the User
The GSMS then forwards the purchase request to the Banking service
Banking service verifies the user credentials and requests for the password/PIN
User enters the password/PIN on the ISR and GSMS forwards this to the Banking service again.
PIN is correct?
No
Error message is sent to the User end
Yes
Sufficient Balance?
No
Yes
Money is debited and a confirmation OTP is sent to the User regarding the purchase

Fig. 6. Example scenario in which smart card is deployed.

5. Example Scenario

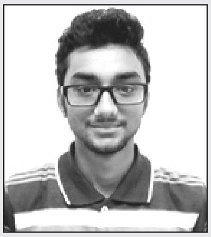
An example presented in this section is when the Smart card is used at a retail outlet to purchase items. As seen in the diagram (Fig.6), First, The smart card is authenticated by the GSMS and the purchase request is forwarded to the banking service. Once the credentials are verified by the service provider, the money is debited from the account of the user and is sent to the retail outlet's account. All communication between the User and the Banking service is done through the GSMS in an encrypted manner. A pre-requisite for this is that both, the banking service and the retail outlet have been registered with the GSMS and have been provided unique SIDs (Service Identification Numbers).

6. Conclusion

The framework proposed in this paper provides the user with cloud-based services using the unique UID provided by the GSMS. It focuses on User convenience and mitigating the security based vulnerabilities of using the open channel in contactless smart cards.

References

1. Daisy Premila Bai, T; Michael Raj, K; Albert Rabara, S: Elliptical Curve Cryptography based Security Framework for Internet of Things (IoT) Enabled Smart Card, World Congress on Computing and Communication Technologies (WCCTT), 2017.
2. Moncef, A; Amar, S: Elliptic Curve Cryptography and its Applications, Proceedings IEEE International Workshop on Systems Signal Processing and their Applications (WOSSPA), 247-250, 2011.
3. Ansari B; Hasan, MA: High-Performance Architecture of Elliptic Curve Scalar Multiplication, Computers, IEEE Transactions, vol. 57, no. 11, 1443, 1453, 2008 ◘



Anirudh Srivathsan is a graduate in Electronics and Communication Engineering from PES University, Bengaluru, India. His areas of interests include IoT, Computer Networks and Cryptography. He is currently working on developing security framework for IoT devices and comparative study of cryptographic algorithms for enhanced security.

Ayush Ambastha is a graduate in Electronics and Communication Engineering from PES University, Bengaluru, India. He has worked in Red Hat on the Web Administration tools for SDS (Software Defined Storage) solutions and containerization software's such as Openshift 4.0 as a full stack developer. His areas of interests are Cyber Security, IoT and Robotics. He is a Red Hat Certified Specialist in Security: Linux, Red Hat Certified Engineer (RHCE) and a Red Hat Certified System Administrator (RHCSA).

Sunil I Dodamani is a graduate in Electronics and Communication Engineering from PES University, Bengaluru, India. His areas of interests are Communication, Cryptography, Network Security. He is currently working on Cryptographic methods for security in IoT devices.

Rajasekar Mohan is presently working as an Associate Professor, Dept of ECE, PES University, Bengaluru. He is a graduate in Electrical and Electronics Engineering from College of Engg., Guindy, Chennai; M.Tech(CSE)-IIT Madras and MBA (International Business) from Annamalai University. He has 25 years of Industry experience in operating, maintaining, planning national level high-availability terrestrial communication systems for mission-critical operations. The systems include Microwave (LOS), Troposcatter, Mobile & Transportable communication systems, OFC, VSATs, WAN/LAN etc. He has 9 Conference papers published in various National/International conferences. His areas of interest and research include Wireless systems, IoT & Robotics and Embedded Systems.