Journal of Theoretical and Applied Information Technology
31st August 2019. Vol.97. No 16

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

# COMPARISON OF ECC AND RSA ALGORITHMS IN IOT DEVICES

**ZEINAB VAHDATI, SHARIFAH MD YASIN, ALI GHASEMPOUR, MOHAMMAD SALEHI**

Department of Computer Science
Faculty of Computer Science and Information Technology
University Putra Malaysia

zvahdati70@gmail.com, ifah@upm.edu.my, ali.ghasempor@ieee.org, mohammad.salehi@ieee.org.

**ABSTRACT**

IoT is the evolution of the internet. Concerning tight communication between the individual and business, number of IoT nodes are rapidly increasing. Most of the services in IoT heavily rely on security mechanisms that pose security imperative for embedded devices in IoT. The failures in IoT can have severe results; consequently, the research toward security concerns are of extreme significance in IoT. Preserving the confidentiality and privacy, ensuring the availability of the services that are proposed by IoT ecosystem, assuring the safety of the assets in IoT like devices, data, infrastructures, and users, are the main objectives in IoT security. The significant issue that makes IoT devices vulnerable is the lack of an appropriate security mechanism to preserve data. Attackers can exploit these weaknesses to obtain access to valuable data. Hence, thoughtfully chosen and practically tested encryption algorithm must be performed to enhance the device efficiency and decrease the risk of sensitive data exposure. Understanding and comparing algorithms implemented in IoT devices, regarding performance discussed in this paper. RSA (Rivest Shamir Adleman) and ECC (Elliptic Curve Cryptography) algorithms have been compared for identifying the most lightweight, secure, efficient implementation in IoT. Based on the findings, the ECC algorithm outperforms RSA in a constrained environment in terms of memory requirements, energy consumption, key sizes, signature generation time, key generation and execution time, and decryption time while RSA performs better in verifying the signature and encrypting.

**Keywords:** *Elliptic Curve Cryptography, ECC, RSA, Internet of Things (IoT), Security Services.*

## 1. INTRODUCTION

Internet of Things (IoT) is a worldwide network of some interactive physical and virtual devices (1) that has been introduced by Kevin Ashton for the first time in 1999 within the context of industrial supply chain management (2). The objects that are internet-connected in IoT by the embedded sensors can collect and exchange data (3); objects like readers, RFID tags, actuators, and sensors that enable interactions among the virtual and physical worlds (4). Typical IoT deployment includes various devices with integrated network-based sensors, as illustrated in Figure 1. There are many IoT applications such as process automation, logistics, remote monitoring, smart metering, smart cities, retail, traffic, and health which can be classified into different domains (5).

Since individuals and IoT ecosystem have a direct interaction together, enormous amounts of data can be recorded, processed, stored, consumed and shared; so the collected data can be used to infer or extract sensitive information that is related to the privacy of individuals (6). Confidentiality, availability, integrity, and non-repudiation of connected systems are critical when they are embedded systems in real time (7).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

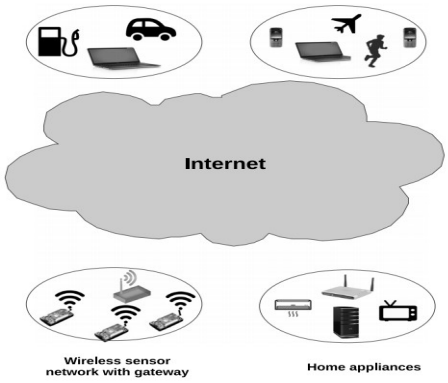


*Figure 1. An overview of IoT elements* (8).

Consequently, it is essential that IoT systems should assure the integrity and confidentiality of the information also the anonymity and privacy of individuals. In IoT-enabled systems, the context in which personally identifiable data is collected must be respected, and end users must be able to monitor sensitive information and decide for themselves to what extent, how and when, information related to them is disclosed to others (6).

Figure 2 presents an authoritative overview of the supportive mechanisms and requirements. Confidentiality can be achieved by encryption (containing encryption of data and VPNs). Digital signatures produce integrity. While the data is transacted between various downstream parties (like e-health claim processing, chain-of-custody, and so on), the recursive digital signatures could be excellent. Availability also can be managed by Intrusion Detection mechanisms (9).

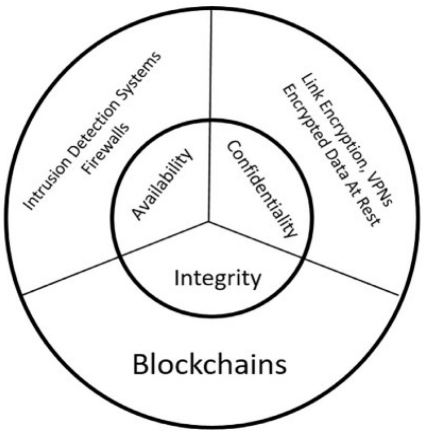


*Figure 2. Security mechanisms for IoT/CPSs* (9)

The security in IoT devices is further significant than a usual system, since the calculations capacities, energy and memory in smart objects are limited (10). Security considers as one of the vital concerns when transferring the data among IoT nodes across the public domain (11). In IoT, an enormous amount of raw data is continuously gathered that needs real-time sensor data streams also techniques to convert these raw data to valuable knowledge. Furthermore, considering privacy and security of data, design criteria of cryptographic algorithms proposed for devices with meager resources are different from that of commonly used ones. This particular field leads to a branch of modern cryptography - lightweight cryptography. Cryptography, as the study of converting normal data into an unreadable form, is performing a vital role in information security (12).

Various cryptographic protocols/methods are used to solve security concerns by encrypting the data (11). Cryptography is a technique used for electronic protection over transmission of valuable data which is mainly science for implementing information security. The primary purpose of cryptography is to preserve data by various authentication scheme. During authenticating the data, it is essential to consider that it should cost less than the value of the original data (13).

Security in IoT networks may be obtained by various encoding algorithms, including Data Encryption Standard (DES), Advanced Encryption Standard (AES), Rivest, Shamir and Adleman (RSA) algorithm, and Elliptic Curve Cryptography (ECC). The most significant and secured algorithms in public key algorithms and also cryptography applications is the RSA algorithm (10).

ECC as another algorithm is developed by Neil Kobiltz and Victor Miller in the 19th century (13), (14). ECC is a public key cryptosystem like RSA (13), (14), (10) which its security strength depends on the difficulty of Elliptic Curve Discrete Logarithm Problem (ECDLP) (13), (14).

ECC is a scalar multiplying which involves point adding and doubling operation (13). ECC has been introduced as an alternative to the appointed methods like the DSA to eliminate the problems of key size, redundancy, and low speed. ECC as the algebraic-curve-based system uses elliptical curve points over a limited field (3).

Recently, various academic research has attained positive progress to address security and privacy concerns in IoT systems (15).

This research is comparing the ECC algorithm with RSA algorithm based on analyzing data extracted from literature research and obtained from

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

technical reports. The aim of comparison in this research is to solve some questions such as:

I. What are the performance metrics for comparison of ECC and RSA in IoT devices?

II. Which algorithm outperforms in terms of each parameter?

   a. Which one consumes less energy and requirements (memory and key size)? Which one is more efficient and affordable?

   b. Which algorithm is faster in execution? (generation and verification signature, key generation, and encryption and decryption time).

III. What are the security requirements in IoT devices?

The outcome of this research will be valuable for future work and other researchers. This comparison between ECC and RSA demonstrate performance metrics among these algorithms and elaborate which algorithm outperforms.

The rest of this paper is structured as follows:

Section 2, The methodology regarding preparing the paper has been explained.

Section 3, Briefly explains RSA and ECC algorithms.

Section 4, Some related works that have been done before, have been discussed to show the detail of previous work relating to these algorithms.

Section 5, There is a discussion based on comparing those algorithms.

Finally, Section 6, Conclusion according to the advantages and disadvantages of RSA and ECC in IoT.

## 2. METHODOLOGY

This research will survey a comprehensive review of ECC and RSA algorithms in IoT system by comparing them in terms of some metrics such such as memory requirement, energy consumption, key size, signature generation and verification time, key generation and execution time, encryption and decryption time.

This study is based on previous works on IoT and its security issues including the existing methods to preserve IoT devices. We follow the methodology in Figure 3. As shown in Figure 3, there are five phases in this research, as follows:

Phase I: In this phase, we identified the necessary information about RSA and ECC in IoT by doing a comprehensive literature review.

Phase II: Based on previous researches, measurement parameters have been determined. Parameters such as memory requirement, energy consumption, key size, signature generation and verification time, key generation and execution time, encryption and decryption time.

In this phase, we also identified the security services for IoT and the issues related to them. As presented in Table 1 the mentioned security services are data confidentiality, data integrity, authentication, availability, non-repudiation, and access control.

*Table 1. Security services provided by public-key cryptographic primitives.*

| Security services | Tool |
|---|---|
| Data confidentiality | Encryption/ Decryption |
| Data integrity | Digital signature |
| Authentication | Digital signature |
| Non-repudiation | Digital signature |

Phase III: Afterward, the taxonomy of information about mentioned parameters was formed in separate tables to clarify the comparison. We created the tables for each performance parameters.

Phase IV: In this step, the authors evaluated collected information and compared RSA and ECC. Gathered data has been analyzed based on performance metrics; meaning that which algorithm performs better in terms of consuming less energy, requirements, and time. In this step, the issues of security services in IoT also has been discussed.

Phase V: Lastly, we documented the obtained results, and conclusion has been provided to clear out the research aim.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

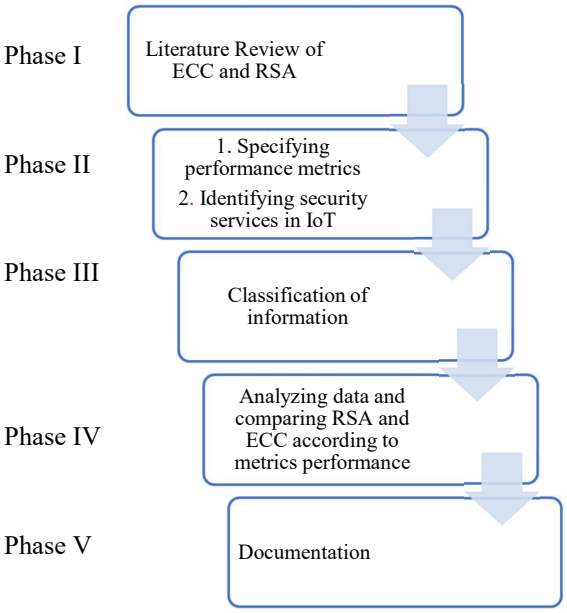


*Figure 3. Research Methodology of paper*

## 3. RSA AND ECC

### 3.1. Rivest, Shamir and Adleman (RSA)

As is mentioned RSA is one of the most generally used and oldest public key cryptography algorithms. In 1977 Ron Rivest, Adi Shamir and Leonard Adleman invented the algorithm (16). Indeed, since RSA algorithm uses a key of at least 1024 bits, and it is a compatible asymmetric cipher and security in this algorithm is assured at the expense of speed (17). This algorithm provides excellent safety in the IoT and MQTT (Message Queuing Telemetry Transport) systems; however, because of some issues such as high energy consumption and complex computing, it isn't compatible with performing at IoT devices (10).

RSA for encrypting and decrypting plaintext makes use of private key and public key. That through the consecutively of this at higher speed, mass encryption-decryption operations can be carried out. RSA is generally applied to secure sensitive data. The protection of RSA relies on the IFP (Integer Factorization Problem) (18).

### 3.2. Elliptic Curve Cryptography (ECC)

The use of ECC has been increased continuously in IoT applications over the last few years (11). ECC was developed in the 19th century as a public key cryptosystem by Neil Kobiltz and Victor Miller (13). In IoT applications, the end nodes require performance optimization of the device concerning improving computing speed and reducing power consumption without any security compromising on the connected devices (11). The difficulty of ECC makes it tough for the attacker to comprehend the ECC and breach the security key. The security level provided by RSA needs1024-bit key but in ECC it can be obtained with a 160-bit key. It is, therefore, appropriate for resource limitation devices such as smart cards, mobile devices and so on. The selection of the suitable elliptical curve is also not simple. Standardization of ECC is essential for effective and practical implementation. National Institute of Standards and Technology (NIST) presents the specifications for ECC that are considered secure to use in the cryptographic application (13).

## 4. RELATED WORK

In 2004, Jansma and Arrendondo, have compared RSA and ECC in terms of the performance characteristics such as key size, run-time (key generation performance, signature generation performance, and signature verification performance). The ascertained results prove that the key generation in RSA is significantly slower than the ECC. It also has been determined that ECC is faster than RSA in creating a digital signature but slower in digital signature verification (especially with the large key length). Therefore, RSA can be the best choice for applications demanding verification of messages more frequently than the signature generation (16).

In (17), there is a review paper proposed by Bafandehkar et al. that compared RSA and ECC based on analyzing the data gathered from technical reports and literature research. According to this paper, ECC has a smaller ration of cost rather than RSA. Furthermore, ECC in comparison with RSA can assure the same level of security with smaller key sizes. Hence, when the computational load isn't increased, ECC has been more suggested to perform further safety and higher speed.

Research on ECC implementation in application embedded iOS has been done by Alam et al. in (19) to compare the performance measures of ECC in a wireless environment, with RSA. From this paper, it determines that RSA in compare with ECC has ten times more computational expenditure than ECC. The size of key pairs and parameters of systems in ECC is smaller than RSA. Since in the same security level, RSA needs the key with a much larger size, ECC can save the bandwidth more considerably than RSA. According to this paper, the ECC key generation are faster than RSA and ECC is much more efficient for small devices compare to RSA.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Dhillon and Kalra also have presented some security challenges in designing secure embedded systems and performed a comparing between RSA and ECC in (7). The comparison result of this paper shows a significant difference between RSA and ECC in terms of execution time. Based on this paper, ECC by applying smaller keys and giving the higher strength of security can save performance costs such as consuming memory, computational costs, and processing power. ECC can be implemented on smaller chips to operate cryptography faster and run quickly with less cost that caused the device to generate insignificant heat and consume lower power. The difference in key size of RSA has made it less appropriate for systems in real time whereas ECC with smaller key sizes has quite complex cryptography and is suitable for constrained embedded systems in real-time. Moreover, it is 10,000 times difficult breaking ECC in compared to an equal 2048-bit RSA. However, the authors also mentioned that ECC has some limitations in terms of reduced battery backup, lesser CPU capacities and small memory that make it difficult to be implemented efficiently.

In (6) a generic implementation of ECC for Smart Parking management systems to optimization of parking spaces within a city has provided a solution that protects the privacy of the users. The author highlights some advantages to ECC:

- ECC outperforms RSA in restrained environments regarding energy consumption, memory requirements, and computation time.
- ECC achieves the same level of security with RSA using smaller parameter sizes.
- ECC uses smaller message sizes that lead to cost less and can be better delivered.

Subsequently, as a point in this protocol that can be discussed, is pre-loading the elliptic curve used including its parameters on the memory of the device. Thus, the tampering and physical attacks can compromise the elliptic curve and its settings (6).

Hasan et al. in (20) have verified ECC-BROSMAP (Broadcast based Secure Mobile Agent Protocol) applying Scyther and then compared it with BROSMAP in terms of computational cost and execution time. According to this paper, execution time in ECC with the key sizes of 224 and 256 is about twice and four times faster than RSA 2048 and 3072 respectively, and the cost of computational also in ECC is more efficient than RSA. ECC-BROSMAP can provide the same level of security requirements as RSA-BROSMAP but with more efficiency and lightweight. ECC-BROSMAP has eliminated the asymmetric encryption, using keys with smaller size, and applied only symmetric encryption to combine with ECC keys.

Mahto and Yadav also have presented a comparative analysis of RSA and ECC in (21). This paper has compared the time lapse of encryption and decryption in RSA and ECC on three samples of input data (8,64 and 256 bits). According to their experimentation, it has been observed that ECC is more applicable and efficient in decryption but in encryption is slow while RSA is more suitable in encryption and slow in decryption. This analysis also recommends that ECC can take less memory. Overall based on their result, it ascertains that ECC surpasses RSA in terms of security and operational efficiency with lesser parameters and is more appropriate for resource constraint devices.

Chhabra and Arora in (22) have proposed a security scheme based on ECC to prevent eavesdropping attacks in Cloud Environments and then compared that with RSA performances. Based on this paper it is determined that the proposed system using ECC excels RSA and is much faster in practical implementation. The proposed ECC based scheme is much faster than RSA for both encryption and decryption and is useful to secure the private data of users. It also is an excellent option for a security mechanism upon eavesdroppers in Cloud storage services. Overall, where users and their storage data are continuously increasing, ECC is excellent for utilization in Cloud storage services.

Another comparison between the algorithms has been presented in (23). The results found in this paper also determine that ECC is more efficient than RSA regarding execution time (encryption and decryption) and key generation time. Memory requirement measure needed in ECC is less than RSA.

In 2018, a researcher has compared Elliptic Curve Digital Signature Algorithm (ECDSA) and RSA regarding key size, energy consumption, average time, when executed on a resource-constrained IoT node. The results obtained from this paper determines that ECDSA has much better results regarding energy efficiency and response time, and it is also suitable to secure resource-constrained IoT devices (24).

In (25), a comparison of ECDSA and RSA as the two most used algorithms in TLS (Transport Layer Security) authentication has been proposed. The results determine enormous differences in the same level of security between RSA and ECC in terms of key size and energy consumption. ECC outperforms RSA regarding values of data

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

throughput and energy consumption. If a higher level of security is needed, the key size in RSA won't be practical but ECDSA could ensure IoT deployments.

Albela et al. in (26) have presented another comparison of RSA and ECC in a fog gateway and two mist end devices. The achieved results ascertained that ECC overcomes RSA in terms of data throughput and energy consumption in all the levels of security. For example, in a security level of 128-bit, RSA consumes the twice of energy than secp256r1 and secp256k1 curves. The secp256r1 and secp256k1 curves are both elliptic curves over a field $z_p$ that p is a 256-bit prime (though different primes for each curve). The "k" in sepc256k1 stands for Koblitz and the "r" in sepc256r1 stands for random. A Koblitz elliptic curve has some unique features that make it feasible to perform the group operation more efficiently (27). It also is concluded that, while the level of security is increased, the key sizes needed for ECC increase more in a linear way than that one's needed for RSA. As mentioned in the previous paper, ECC curves are the curve and platform dependence on the implemented optimizations (secp256r1 curve had better results in compared to the secp224r1) (26).

Recently in 2018 Albalas et al. have introduced a lightweight secure Constrained Application Protocol (CoAP) applying ECC and compared advantages of ECC with RSA. The result obtained of this paper discovered that the proposed secure CoAP with ECC could overcome the CoAP applying RSA regarding saving energy by 47%. Moreover, when applying RSA, the energy consumption of the battery is faster than the CoAP (2).

In (10), an implementation scheme of RSA has been proposed by Kaedi et al. to resist power analysis attacks. In this paper, some disadvantaged of RSA in IoT has been mentioned. Weaknesses such as extra computational load, high power consumption, numerous countermeasure in the side channel that causes the need for more memory, increase the executing time, also need TRNG (True Random Number Generator) that calculate the blinding factors, in conclusion, RSA is not appropriate for IoT approach.

## 5. COMPARATIVE ANALYSIS OF ECC AND RSA

In this part, ECC and RSA has been compared based on performance metrics and security services.

### 5.1. Security Services

ITU-T X.800 has explained the security service as a service that guarantees to preserve systems and/or to transfer data during protocol layers applying suitable security. Moreover, as CNSS declared, security services can accommodate two or more security requirements. Therefore, the security service must satisfy grantees security provisions to transfer data or preserve the systems (2).

Encryption and decryption of Public-key cryptography (asymmetric-key cryptography) are performed by applying two different keys. These two keys in such a key pair are identified as the private and the public key. In public key cryptography, each party is concerned about secure communications during their public key publishing (28). Among the security services which are listed as follows; (2) the public key cryptography can provide some of them:

1. Data confidentiality
2. Data integrity
3. Authentication
4. Availability
5. Non-repudiation
6. Access control (2).

If part A wants to have a confidential communication with the part B should encrypt a message by using B's publicly available key. Such communication can be decipherable only by B because only B has access to the corresponding private key.

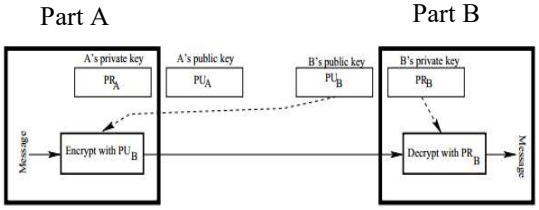


*Figure 4. Confidential communication in public key cryptography* (28).

In Figure 4, if part A wants to transmit an authenticated message to party B, part A should encrypt the message by A's private key. Because this message can be decipherable only by A's public key, which can prove the authenticity of the message (A is indeed the source of the message).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

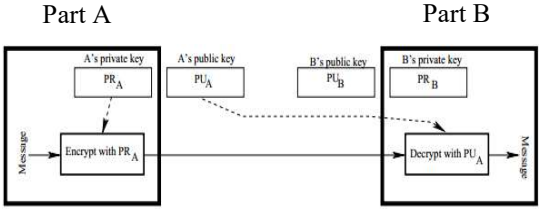


*Figure 5. Authenticated communication in public key cryptography* (28).

The public-key cryptography can accommodate both authentication and confidentiality of messages at the same time.

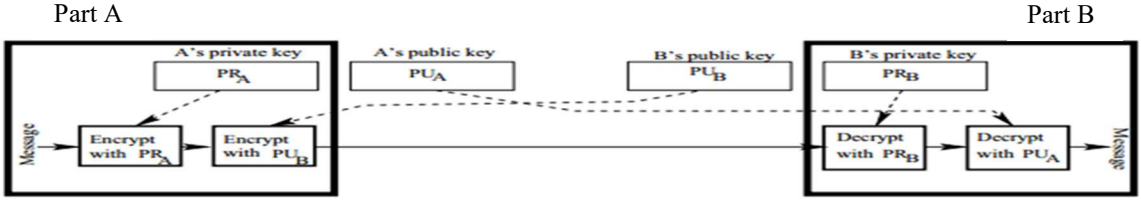


*Figure 6. Confidentiality and authentication* (28).

Figure 6 displays how public-key cryptography can be applied for confidentiality, authentication (digital signatures), and both (28). ECC can provide security services such as Confidentiality, Integrity, Authentication, and Authorization in IoT as follows:

1. Confidentiality: Any unauthorized connection is rejected from access to the data via this security service.
2. Integrity: To ensure that messages received through a destination are not altered.
3. Authentication: It also can be achieved by using the public key, if any anonymous /malicious node wants to interact with network nodes, it requires the public key pair from the authorized node.
4. Authorization: This service provides a unique key pair (private and public) to each node to make the decryption and encryption process (2).

### 5.2. Performance Metrics

Both Elliptical Curve Encryption (ECC) and Adleman algorithm (RSA) are widely used in the IoT environment. Lack of stored energy and computational power is the primary constraint for IoT devices. On the other hand, security challenges are rapidly growing therefore different countermeasures are proposed. Encryption is an excellent technique to secure transmission; however computational unit power is required to perform encryption. This section is focusing on a technical comparison between ECC and RSA algorithms in terms of memory requirement, energy consumption, key size, signature generation and verification generation, key generation and execution time, and encryption and decryption time. Based on the analysis and measurement, memory requirement, energy consumption, key size, signature generation time, key generation and execution time, and decryption time in ECC are less than RSA. Moreover, RSA running faster in signature verification and encryption data.

#### 5.2.1. Memory requirement

The memory required by the both algorithm is presented in this section.

*Table 2. Memory Requirement.*

| Article | Security level | Key size | | Memory requirement ( bytes) | |
|---|---|---|---|---|---|
| | | RSA | ECC | RSA | ECC |
| A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility with Pairing-Based Cryptography (2017) | 80 | 512 | 106 | 157 | 108 |
| | 112 | 768 | 132 | 236 | 117 |
| | 128 | 1024 | 160 | 313 | 125 |
| | 160 | 2048 | 210 | 621 | 140 |

According to Table 2, ECC is better than RSA in terms of memory requirements and need less memory usage. ECC offers the same security level than RSA with using less memory.

#### 5.2.2. Energy consumption

In this section, the energy consumption rates collected for the ECC and RSA in different security level and with various key sizes have been classified.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

*Table 3. Energy consumption.*

| | Article | Security level | Key size (bits) | | Remark | Energy consumption (mWh) | |
|---|---|---|---|---|---|---|---|
| | | | RSA | ECC | | RSA | ECC |
| 1. | A Practical Performance Comparison of ECC and RSA for Resource-Constrained IoT Devices (2018) | 80 | 1024 | 192 | - | 17.86 | 9.05 |
| | | 112 | 2048 | 224 | - | 21.55 | 17.38 |
| | | 128 | 3072 | 256 | - | 56.78 | 15.43 |
| | | 192 | 7680 | 384 | - | - | 22.26 |
| 2. | Clock Frequency Impact on the Performance of High-Security Cryptographic Cipher Suites for Energy-Efficient Resource Constrained IoT Devices (2018) | 80 | 1024 | 192 | Frequency MHz | | |
| | | | | | 80 | ~20 | ~11.58 |
| | | | | | 160 | ~13.41 | ~9.75 |
| | | | | | 240 | ~13.63 | ~8.26 |
| | | 112 | 2048 | 224 | 80 | ~31 | ~17.27 |
| | | | | | 160 | ~20.69 | ~13.06 |
| | | | | | 240 | ~18.18 | ~13.80 |
| | | 128 | 3072 | 256 | 80 | ~67.5 | ~20.83 |
| | | | | | 160 | ~42.5 | ~14.21 |
| | | | | | 240 | ~38.41 | ~12.97 |
| | | 192 | 7680 | 384 | 80 | - | ~28.84 |
| | | | | | 160 | - | ~19.26 |
| | | | | | 240 | - | ~17.35 |

ECC can consume less battery resource and computing power (29). To detail it out, in Table 3, for article no. 1, ECC with 192 bits key size performed same security level however required just over 9 MWh compare to 17.86 MWh in RSA. When security level increased to 128, key size in RSA rose three times from 1024 to 3072 while that of ECC increased by 64 bit and its energy consumption is near four times as much as ECC. Furthermore, as shown in Table 3, for roughly the same power consumption (RSA with approximately 21.55 mWh and ECDSA with 22.26 mWh), ECDSA with ECC algorithm presents greater security level than RSA (192-bit security level in ECDSA compared of RSA with the 112-bit). The considerable finding of ECC is that the level of security implemented by a curve isn't evermore proportional to its performance. As is shown in Table 3, no1, the curve with secp256 has consumed lower energy and performed higher throughput values compared to the secp224r1 curve which is weakest. The reason behind this is that there is a software optimization that has been implemented to speed up the algorithms in terms of mathematical operations as ECC performance are dependent on curve and platform. In Table 3, for article no. 2, in both algorithms by increasing the frequency from 80 MHz to 240 MHz, energy consuming rate reduces. However, ECC in each security level and frequency uses less energy and performs better than RSA.

### 5.2.3. Key size

To implement an algorithm, the first and major parameter is selecting the size of the keys. It is essential to mention that large key sizes provide better security but also are more expensive, hence selecting the keys should be done carefully to have the smallest key size and greater security (30). The recommended keys size in RSA keeps rising (from 1024 bit to 15360 bit) for maintaining adequate strength in cryptography. However, ECC can provide the same level of security and cryptographic stability with shorter key sizes. ECC improves safety by reducing computational requirements. At the following table, the difference between key sizes is gathered. Since ECC uses the small key size is more attractive for devices in IoT that have restricted storage or processing strength. Additionally, the keys with smaller size can offer faster SSL (Secure Sockets Layer) handshakes (that is speedier in translating the page load) and more powerful security (31).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

*Table 4. Key Size of RSA and ECC.*

| Security level | Key size | | Security-aware CoAP Application Layer Protocol for the Internet of Things using Elliptic-Curve Cryptography | A Practical Evaluation on RSA and ECC-Based Cipher Suites for IoT High-Security Energy-Efficient Fog and Mist Computing Devices | A privacy-preserving smart parking system using an IoT elliptic curve based security platform | Elliptic Curve Cryptography for Real Time Embedded Systems in IoT Networks | Comparison of ECC and RSA Algorithm in Resource Constrained Devices |
|---|---|---|---|---|---|---|---|
| | RSA | ECC | | | | | |
| 80 | 1024 | 160-233 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 112 | 2048 | 224-255 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 128 | 3072 | 256-383 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 192 | 7680 | 384-541 | ✓ | ✓ | ✓ | ✓ | ✓ |
| 256 | 15360 | 512+ | ✓ | ✓ | ✓ | ✓ | ✓ |

According to Table 4, with same security level, ECC has less key size in compare to RSA.

### 5.2.4. Signature generation and signature verification time

A DSA (Digital Signature Algorithm) contains digital signature generation and signature verification processes. A signatory applies the generation process to create a digital signature on data, and a verifier applies a method to verify the authenticity of the signature (32), (33). In this section time of signature generation and verification has been compared.

*Table 5. Signature generation and signature verification time.*

| Article | Security level | Key size | | Signature Generation Time | | Signature Verification Time | |
|---|---|---|---|---|---|---|---|
| | | RSA | ECC | RSA | ECC | RSA | ECC |
| A Comparative Study of RSA and ECC and Implementation of ECC on Embedded Systems (2016) | 80 | 1024 | 163 | 0.01 | 0.15 | 0.01 | 0.23 |
| A Review: Security of Data in Cloud Storage using ECC Algorithm (2016) | 112 | 2240 | 233 | 0.15 | 0.34 | 0.01 | 0.51 |
| Performance Based Comparison Study of RSA and Elliptic Curve Cryptography (2013) (34) | 128 | 3072 | 283 | 0.21 | 0.59 | 0.01 | 0.86 |
| Performance Comparison of Elliptic Curve and RSA Digital Signatures (2004) | 192 | 7680 | 409 | 1.53 | 1.18 | 0.01 | 1.80 |
| | 256 | 15360 | 571 | 9.20 | 3.07 | 0.03 | 4.53 |

In Table 5, the time of signature generation and signature verification in RSA is comparable and quicker than ECC respectively. As shown, the time of signature generation in RSA and ECC isn't too different in the security level between 80 and 192, while at the larger key sizes (with the 15360 and 571 in RSA and ECC respectively), RSA uses about three times more than ECC.

RSA in larger key size consumes much more time (16). However, RSA outperforms ECC in terms of signature verification. In RSA, the time for verifying a signed message is insignificant for the used key length, while ECC falls behind to perform in each key range and show an approximately linear increase with rising the key sizes. Hence, RSA can be applied for applications demanding verification of messages more than the generating signature (16).

### 5.2.5. Key generation and execution time

Key generation is the first and essential section of an algorithm. It is applied to generate the public and private keys that take different times in each algorithm. RSA and ECC also vary in terms of time at creating and executing phase. Table 6 shows the differences between them (30).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

*Table 6. Key generation and execution time.*

| | Article | Security level | Key size | | Key Generation Time (ms) | | Execution time (ms) | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | RSA | ECC | RSA | ECC | Remark | RSA | ECC |
| 1. | A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility with Pairing-Based Cryptography (2017) | 80 | 512 | 106 | 383 | 57 | Public key operation | 430 | 810 |
| | | | | | | | Private key operation | 10990 | 810 |
| | | 112 | 768 | 132 | 889 | 98 | Public key operation | 1940 | 2190 |
| | | | | | | | Private key operation | 83260 | 2190 |
| | | 128 | 1024 | 160 | 2609 | 108 | - | - | - |
| | | 160 | 2048 | 210 | 18399 | 121 | - | - | - |
| 2. | Secure Lightweight ECC-Based Protocol for Multi Agent IoT Systems (2017) | 80 | 2048 | 224 | - | - | Creating request | 78.77 | 32.66 |
| | | | | | - | - | Decryption result | 34.89 | 33.40 |
| | | | | | - | - | Preparing result | 16.49 | 0.81 |
| | | 112 | 3072 | 256 | - | - | Creating request | 181.80 | 32.66 |
| | | | | | - | - | Decryption result | 34.93 | 33.40 |
| | | | | | - | - | Preparing result | 48.51 | 0.81 |
| 3. | Elliptic Curve Cryptography for Real Time Embedded Systems in IoT Networks (2016) | 80 | 1024 | 160-233 | ~2000 | ~476.19 | - | - | - |
| | | 112 | 2048 | 224-255 | ~15333.33 | ~857.14 | - | - | - |
| 4. | A Comparative Study of RSA and ECC and Implementation of ECC on Embedded Systems (2016) | 80 | 1024 | 163 | 160 | 80 | - | - | - |
| | | 112 | 2240 | 233 | 7470 | 180 | - | - | - |
| 5. | A Review: Security of Data in Cloud Storage using ECC Algorithm (2016) | 128 | 3072 | 283 | 9800 | 270 | - | - | - |
| | | 192 | 7680 | 409 | 133900 | 640 | - | - | - |
| 6. | Performance Comparison of Elliptic Curve and RSA Digital Signatures (2004) | 256 | 15360 | 571 | 679060 | 1440 | - | - | - |

Table 6, no. 1, points out that the public key operation in RSA is faster while ECC in the private key operation is so much faster than RSA. For private key operations in the security level of 80-bit and 112-bit, RSA is about 14-times and 40-times slower than ECC respectively. While in public key operation, ECC is slower than RSA (23). Moreover, based on Table 6, no. 2, since ECC-BROSMAP applies symmetric encryption; therefore, ECC execution time is less than RSA besides performance optimized. On the other hand, the time of creating request (by the mobile car rental application) and preparing result (by the server) in ECC is lower compared to RSA. Because of the number of asymmetric operations applied in BROSMAP based RSA; which make ECC quicker than RSA while creating a request. While RSA's decryption time is slightly faster than ECC. During decrypting results, RSA-BROSMAP needs one asymmetric process, one symmetric process, and three hashes, while ECC-BROSMAP needs equal hash numbers and two symmetric operations. The

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

compiler in JAVA runs the second and third encryptions in ECC quicker than the first symmetric encryption because of the code optimization. That presents comparable results with BROSMAP based RSA even though the text is encrypted twice in ECC which should reduce the performance (20). In Table 6, no 4,5 and 6, the results ascertain that key generation process is much faster and get a better result in ECC (RSA is about 15 times slower than ECC). Following that ECC algorithm doesn't have to allocate resources to create prime numbers which are a computationally intense act, can generate the public and private key pair with higher speed in comparison to RSA. In conclusion, key generation time in ECC increases linearly with the key size; however, RSA grows exponentially (16). RSA guarantee can handle 450 requests per second with 150 milliseconds average response time wherever ECC requires only 75 milliseconds to respond to the same amount of requests per second. ECC has excellent response time when it communicates for the server to desktop (29).

### 5.2.6. Encryption and decryption time

Encryption is the method to encode information or message so that just authorized and allowed parties can access and it's not reachable by unauthorized parties (35). Decryption is the method that takes encrypted or encoded information or messages and turns them back into plaintext which is readable for user or computer (36). These processes take some time to be done which is different in each algorithm. In this section in Table7 results of comparing the encryption and decryption time between RSA and ECC using different key sizes have been gathered and presented.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

*Table 7. Encryption and decryption time.*

| No | Article | Security level | Key size | | | Encrypt /Decrypt Time (ms) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | RSA | ECC | | Total Time | | Remark | RSA | ECC |
| | | | | | | RSA | ECC | | | |
| 1. | RSA and ECC: A Comparative Analysis (2017) | 80 | 1024 | 160 | 8 bits | 785 | 1815.2 | Encryption | 30.7 | 488.5 |
| | | | | | | | | Decryption | 754.3 | 1326.7 |
| | | | | | 64 bits | 5673.8 | 8078.4 | Encryption | 136.6 | 2168.5 |
| | | | | | | | | Decryption | 5537.2 | 5909.9 |
| | | | | | 256 bits | 19877.2 | 30809.1 | Encryption | 559.6 | 7924 |
| | | | | | | | | Decryption | 19317.7 | 22885.1 |
| | | 112 | 2048 | 224 | 8 bits | 2737.5 | 3789.3 | Encryption | 29.9 | 2203 |
| | | | | | | | | Decryption | 2707.5 | 1586.3 |
| | | | | | 64 bits | 20574.3 | 16918.8 | Encryption | 163.5 | 9985.5 |
| | | | | | | | | Decryption | 20410.8 | 6933.3 |
| | | | | | 256 bits | 102615.3 | 66033.9 | Encryption | 581.5 | 39700.8 |
| | | | | | | | | Decryption | 102033.7 | 26333.1 |
| | | 128 | 3072 | 256 | 8 bits | 6971.4 | 5645.3 | Encryption | 30.5 | 3876.3 |
| | | | | | | | | Decryption | 6940.9 | 1769 |
| | | | | | 64 bits | 46645.4 | 22446.6 | Encryption | 167.2 | 15088.2 |
| | | | | | | | | Decryption | 46478.2 | 7358.4 |
| | | | | | 256 bits | 210169.7 | 85844.6 | Encryption | 561.1 | 58438.6 |
| | | | | | | | | Decryption | 209608.6 | 27406 |
| | | 144 | - | - | 8 bits | 13696.2 | 6728.8 | Encryption | 48.9 | 4726.6 |
| | | | | | | | | Decryption | 13647.2 | 2002.2 |
| | | | | | 64 bits | 77902.7 | 28709.3 | Encryption | 138.5 | 20230.8 |
| | | | | | | | | Decryption | 77764.2 | 8478.5 |
| | | | | | 256 bits | 311636.8 | 109655.6 | Encryption | 571.8 | 77503.4 |
| | | | | | | | | Decryption | 311064.9 | 32152.2 |
| 2. | A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility with Pairing-Based Cryptography (2017) | 80 | 512 | 106 | - | 77 | 11 | - | - | - |
| | | 112 | 768 | 132 | - | 160 | 17 | - | - | - |
| | | 128 | 1024 | 160 | - | 338 | 16 | - | - | - |
| | | 160 | 2048 | 210 | - | 1867 | 15 | - | - | - |

In Table 7, the overall time of encryption and decryption operations in ECC is less than RSA; however, it is noticeable to mention that based on Table 7, no 1, ECC takes longer time than RSA in encryption, but it is very efficient than RSA in decryption whereas RSA is very efficient in encryption but slow than ECC in decryption. But in

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

overall ECC is more effective than RSA as based on the results in Table 7.

## 6. RESEARCH FINDINGS

### 6.1. What are the performance metrics for comparison of ECC and RSA in IoT devices?

Performance of the ECC and RSA algorithms is evaluated based on the various metrics which are best suited for the cryptographic algorithms. The metrics that are selected for the evaluation are memory requirement, energy consumption, encryption and decryption time (37), key generation and execution time, key size, signature generation, and verification time.

6.1.1. Memory requirement: Different encryption techniques need different memory size for implementation. The memory requirement depends on the number of operations that are done by the algorithm, the key size, initialization vectors used, and type of services. The memory utilized influences the cost of the system. It is desirable that the memory required should be as small as possible (38).

6.1.2. Energy consumption: It refers to the entire energy needed by the encryption/decryption algorithm. It is appraised according to the throughput of the encryption/decryption algorithms.

6.1.3. Key size: In encryption methodologies, key management is an essential aspect that determines how the data is encrypted. The image loss the encryption ratio is based on this crucial length. The symmetric algorithm utilizes a variable key length, which is longer. Each algorithm employs a particular number of key length, which is used as a seed in the process block (38).

6.1.4. Signature generation time: Time of generation signature that uses a private key to generate a digital signature (39).

6.1.5. Signature verification time: Signature verification is a technique to compare signatures and validate the identity of an individual (is used by banks, intelligence agencies and high-profile institutions) (40). The time of verification refers to time of verifying the signature when a user accesses the system (41).

6.1.6. Key generation and execution time: This time refers to the time required by key generation function to create keys. All these functions produce different times based on the size of text files and key length in any algorithm (42).

6.1.7. Encryption time: It refers to the entire time needed to generate a cipher-text from plain-text. This time is used to calculate the throughput of the encrypted algorithm (provides the encryption rate).

6.1.8. Decryption time: It refers to the total time needed to generate the plain-text from Cipher-text. This time is used to calculate the throughput of the decrypted algorithm (provides the decrypted rate) (37).

### 6.2. Which algorithm outperforms in terms of each parameter?

The result determines that ECC outperforms in terms of some parameters like memory requirements, energy consumption, key sizes, signature generation time, key generation and execution time, and encryption time.

RSA is more successful regarding signature verification time and decryption time.

6.2.1. Which one consumes less energy and requirements (memory, energy consumption, and key size)? Which one is more efficient and affordable?

In the same security level, ECC needs less memory usage than RSA (23). ECC in each level of security and frequency utilizes less energy (25), (24), (2) battery resource, and power (29). Furthermore, while applying RSA, the energy of the battery consumed faster than ECC (2). In the same security level, ECC has less key size in compare to RSA (2), (26), (6), (7), (17).

ECC with using the less key size and energy consumption overcomes RSA. Since ECC saves the bandwidth more than RSA (19) and outperforms RSA by 47% in terms of saving energy (2), is more efficient.

6.2.2. Which algorithm is faster in execution? (generation and verification signature, key generation and execution, and encryption and decryption).

ECC takes less and more time than RSA during the signature generation and signature verification respectively (19), (30), (34), (16). At the larger key

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

sizes, RSA uses about three times more than ECC to generate signature; since some of the time takes to calculate the SHA-1 hash for the message, that RAS in larger key size consumes much more time. In terms of signature verification, RSA outperforms ECC. In RSA, the time of verifying a signed message is insignificant for the used key length, while ECC lags to perform in each key range (16).

Regarding to key generation and execution time, it's concluded that key generation and execution time in ECC is less than RSA (23), (20), (7), (19), (30), (16); and increases linearly with the key size; however, RSA grows exponentially (16).

In terms of encryption and decryption time, the total time to encryption and decryption in ECC is less than RSA (23) (22) (21). Based on the results in (21) it is remarked that RSA is more useful in encryption data while ECC is more efficient in decryption.

Overall, ECC is more secure and effective than RSA (21).

### 6.3. What are the security requirements in IoT devices?

The security requirements in IoT are confidentiality, integrity, authentication, and authorization (2). ECC and RSA can provide security services for IoT, however, based on the findings from I and II, ECC is more suitable for providing the security requirements in IoT.

## 7. CONCLUSION

As can be seen, the number of IoT devices besides the amount of information that will be generated is increasing significantly, therefore main objectives are assuring the safety of IoT devices, data, and users. Choosing an algorithm that provides all confidentiality, privacy and availability plays a vital role in the protection of users and data. In this paper from different aspects both ECC and RSA, algorithms are reviewed comprehensively, and in all parameters, ECC could excel RSA

The comparison has been made in terms of some metrics such as memory requirement, energy consumption, key size, signature generation and verification time, key generation and execution time, encryption and decryption time.

The result shows that ECC is more successful in terms of some parameters like memory requirements, energy consumption, key sizes, signature generation time, key generation and execution time, and decryption time.

Regarding to memory requirement, ECC surpasses RSA in terms of security and operational efficiency in small devices and constrained resource. Since energy consumption of the battery is faster in RSA, ECC saves 47% energy and overcomes RSA. ECC also saves the bandwidth more than RSA. It has been estimated that in the future, the key size in RSA won't be practical for a higher security level and ECC will able to preserve IoT deployments. As RSA applies the various countermeasure and extra computational load, hence, requires more memory and additionally, is slower than ECC to generate the signature, generate the key, and execute.

However, on restrained embedded devices, ECC has some limitations regarding reduced battery backup, minor CPU capacities and small memory that make it difficult to implement efficiently. These problems should be solved to produce an extremely optimized implementation in embedded devices. Consequently, ECC has been more suggested that offers more security and higher speed. While RSA performs better in verifying the signature and encrypting.

Therefore we can conclude that since ECC use smaller key sizes (its computational cost is about ten times less than RSA) and performs faster and more efficient, is more potentially offered for devices in IoT. Especially in devices like embedded systems or smart cards that need cryptography to transfer data securely and so on. Moreover, RSA can be more appropriate to employ in some application that requires to verify messages more than generating a signature.

## ACKNOWLEDGMENT

We would like to thank the Ministry of Higher Education, for supporting directly in this research work under FRGS Grant no. 5524822.

## REFERENCES:

[1]. 1. Porion S. Reassessing a Turbulent Decade: the Historiography of 1970s Britain in Crisis. Études Anglaises [Internet]. 2016;69(3):301–20. Available from: https://www-cairn-info.ezp-prod1.hul.harvard.edu/revue-etudes-anglaises-2016-3-page-301.htm

[2]. 2. Albalas F, Al-Soud M, Almomani O, Almomani A. Security-aware CoAP application layer protocol for the internet of things using elliptic-curve cryptography. Int Arab J Inf Technol. 2018;15(3A Special Issue).

[3]. 3. Tiwari HD, Kim JH. Novel Method for DNA-Based Elliptic Curve Cryptography for IoT Devices. ETRI J. 2018;40(3):396–409.

[4]. 4. Riahi Sfar A, Natalizio E, Challal Y,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Chtourou Z. A roadmap for security challenges in the Internet of Things. Digit Commun Networks [Internet]. 2018;4(2):118–37. Available from: https://doi.org/10.1016/j.dcan.2017.04.003

[5]. 5. Čolaković A, Hadžialić M. Internet of Things (IoT): A review of enabling technologies, challenges, and open research issues. Comput Networks. 2018;144:17–39.

[6]. 6. Chatzigiannakis I, Vitaletti A, Pyrgelis A. A privacy-preserving smart parking system using an IoT elliptic curve based security platform. Comput Commun [Internet]. 2016;89–90:165–77. Available from: http://dx.doi.org/10.1016/j.comcom.2016.03.014

[7]. 7. Kalra PDK and S. Elliptic Curve Cryptography for Real Time. Ieee. 2016;1–6.

[8]. 8. Khan MA, Salah K. IoT security: Review, blockchain solutions, and open challenges. Futur Gener Comput Syst [Internet]. 2018;82:395–411. Available from: https://doi.org/10.1016/j.future.2017.11.022

[9]. 9. Minoli D, Occhiogrosso B. Blockchain mechanisms for IoT security. Internet of Things [Internet]. 2018;1–2:1–13. Available from: https://linkinghub.elsevier.com/retrieve/pii/S2542660518300167

[10]. 10. Kaedi S, Doostari MA, Ghaznavi-Ghoushchi MB. Low-complexity and differential power analysis (DPA)-resistant two-folded power-aware Rivest–Shamir–Adleman (RSA) security schema implementation for IoT-connected devices. IET Comput Digit Tech [Internet]. 2018;12(6):279–88. Available from: http://digital-library.theiet.org/content/journals/10.1049/iet-cdt.2018.5098

[11]. 11. Sakthivel TKR. High - performance ECC processor architecture design for IoT security applications. J Supercomput [Internet]. 2019;(0123456789). Available from: https://doi.org/10.1007/s11227-018-02740-2

[12]. 12. Surendran S, Nassef A, Beheshti BD. A survey of cryptographic algorithms for IoT devices. 2018 IEEE Long Isl Syst Appl Technol Conf LISAT 2018. 2018;1–8.

[13]. 13. Shruti P, Chandraleka R. Elliptic Curve Cryptography Security in the Context of Internet of Things. 2017;8(5):90–3.

[14]. 14. Daisy Premila Bai T, Albert Rabara S VJA. Elliptic Curve Cryptography based Securing Framework for Internet of Things and Cloud Computing. Conf Recent Adv Comput Eng by WSEAS. 2015;65–74.

[15]. 15. Noor MBM, Hassan WH. Current research on Internet of Things (IoT) security: A survey. Comput Networks [Internet]. 2018;(xxxx). Available from: https://www.sciencedirect.com/science/article/pii/S1389128618307035?dgcid=rss_sd_all

[16]. 16. Xiang S, Lai S, Meng Y. Performance Comparison of Elliptic Curve and RSA Digital Signatures. Zhongguo Zhong xi yi jie he za zhi Zhongguo Zhongxiyi jiehe zazhi = Chinese J Integr Tradit West Med [Internet]. 2004 Nov;29(11):979–81. Available from: http://www.ncbi.nlm.nih.gov/pubmed/20329605

[17]. 17. Bafandehkar M, Yasin S, Mahmod R, Hanapi ZM. Comparison of ECC and RSA Algorithm in Resource Constrained Devices. 2013;0–2.

[18]. 18. Mamathashree AM, Remya K, Santhosh Kumar BJ. Fault analysis detection in public key cryptosystems (RSA). Proc 2017 IEEE Int Conf Commun Signal Process ICCSP 2017. 2018;2018-Janua:505–8.

[19]. 19. Alam M. A Comparative Study of RSA and ECC and Implementation of ECC on Embedded Systems. 2016;3(03):86–93.

[20]. 20. Hasan H, Salah T, Shehada D, Zemerly MJ, Yeun CY, Al-qutayri M, et al. Secure Lightweight ECC-Based Protocol for Multi- Agent IoT Systems. 2017;

[21]. 21. Mahto D. RSA and ECC : A Comparative Analysis. 2017;12(19):9053–61.

[22]. 22. Chhabra A, Arora S. An Elliptic Curve Cryptography Based Encryption Scheme for Securing the Cloud against Eavesdropping Attacks. Proc - 2017 IEEE 3rd Int Conf Collab Internet Comput CIC 2017. 2017;2017-Janua:243–6.

[23]. 23. Al Hamid HA, Rahman SMM, Shamim Hossain M, Almogren A, Alamri A. A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility with Pairing-Based Cryptography. IEEE Access. 2017;5(XX):22313–28.

[24]. 24. Suárez-Albela M, Fraga-Lamas P, Castedo L, Fernández-Caramés T. Clock Frequency Impact on the Performance of High-Security Cryptographic Cipher Suites for Energy-Efficient Resource-Constrained IoT Devices. Sensors [Internet].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2018;19(1):15. Available from: http://www.mdpi.com/1424-8220/19/1/15

[25]. 25. Su M, Fern TM. A Practical Performance Comparison of ECC and RSA for Resource-Constrained IoT Devices. 2018;

[26]. 26. Suárez-Albela M, Fraga-Lamas P, Fernández-Caramés T. A Practical Evaluation on RSA and ECC-Based Cipher Suites for IoT High-Security Energy-Efficient Fog and Mist Computing Devices. Sensors [Internet]. 2018;18(11):3868. Available from: http://www.mdpi.com/1424-8220/18/11/3868

[27]. 27. Cook JD. A tale of two elliptic curves [Internet]. Available from: https://www.johndcook.com/blog/2018/08/21/a-tale-of-two-elliptic-curves/

[28]. 28. Kak A. Lecture 12 : Public-Key Cryptography and the RSA Algorithm Lecture Notes on " Computer and Network Security " by Avi Kak ( kak@purdue.edu ) Goals : Comput Netw Secur. 2018;1–94.

[29]. 29. RSA vs ECC – Which is Better Algorithm for Security? [Internet]. Available from: https://www.ssl2buy.com/wiki/rsa-vs-ecc-which-is-better-algorithm-for-security

[30]. 30. Harsha A, Patil B. A Review: Security of Data in Cloud Storage using ECC Algorithm. Bonfring Int J Softw Eng Soft Comput. 2017;6(Special Issue):143–6.

[31]. 31. Julie Olenski. ECC 101: What is ECC and why would I want to use it? [Internet]. p. 1. Available from: https://www.globalsign.com/en/blog/elliptic-curve-cryptography/

[32]. 32. Message I, Signatures D, Algorithm DS. Elliptic Curve Digital Signature. 2004;2004(April):1–7.

[33]. 33. Digital Signature Generation and Verification. Available from: https://www.vocal.com/cryptography/dsa-digital-signature-algorithm/

[34]. 34. Sinha R, Srivastava HK, Gupta S. Performance Based Comparison Study of RSA and Elliptic Curve Cryptography. Int J Sci Eng. 2013;4(5):720–5.

[35]. 35. Encryption [Internet]. Available from: https://en.wikipedia.org/wiki/Encryption

[36]. 36. Decryption [Internet]. Available from: https://www.computerhope.com/jargon/d/decrypti.htm

[37]. 37. B.Bharathi, G.Manivasagam MAK. Metrics for Performance Evaluation of Encryption Algorithms. 2008;23(4):291–8.

[38]. 38. Awotunde JB, Ameen AO, Oladipo ID, Tomori AR, Abdulraheem M. 152847-400438-1-Sm. 2016;13(2):74–82.

[39]. 39. Johnson L. Statutory and Regulatory GRC. Secur Control Eval Testing, Assess Handb. 2015;11–33.

[40]. 40. Signature Verification [Internet]. Available from: https://www.techopedia.com/definition/30499/signature-verification

[41]. 41. Signature Verification in Real Time [Internet]. Available from: https://www.xyzmo.com/e-signature-products/signature-verification

[42]. 42. Maqsood F, Ahmed M, Mumtaz M, Ali M. Cryptography: A Comparative Analysis for Modern Techniques. Int J Adv Comput Sci Appl. 2017;8(6):442–8.