# Securing RJSON data between Middleware and Smart phones through Java Script based Cryptographic Algorithms

**Mandeep Singh, Kanwalvir Singh Dhindsa**

***Abstract****—Smartphone has become the most typical and popular mobile device in recent years. It combines the functionality of mobile phone and PDA. Besides, it provides many computers' functionality, Middleware as processing, communication, data storage and etc. It also provides many computers' service, such as web browser, portable media player, video call, GPS, Wi-Fi and etc. This paper provides an effective mechanism for securing the communication between the RJSON data from the middleware and back using various secure cryptographic algorithms implemented in JavaScript. Security plays a vital role in today's mobile world. There are security issues like sniffing of data while accessing information through open channel. Cryptographic techniques play an important role in protecting communication links and data, since access to data can be limited to those who hold the proper key. This paper discusses different cryptographic techniques available in lightweight RJSON format to securely transfer information in a network by an android smartphone. A native android application Sadelok Newspaper is used to securely send data using AES, 3Des and Blowfish and compare them. The paper describes and compares the JavaScript based cryptographic techniques on the JavaScript Object Notation, for faster and efficient encryption of data that is suitable for use in smartphones.*

***Keywords****- 3DES, Blowfish, JSCrypt, Smartphone Security, RJSON, Android, Smartphones.*

## I. INTRODUCTION

Smartphones are usually is a small, portable size computing device, which allows user to input information through touchscreen or small keyboard on the device. Comparing with conventional computer, mobile device is easily carried out but provides much computer functionality, such as processing, communication, data storage. PDA and smartphone are the two most popular mobile devices.

This paper mainly focuses on analysis of security issues on smartphone. Smartphone usually provides many computers' services, such as web browser, video or audio player, video call, GPS, Wireless Network. The top three successful smartphone brands are Google Android, Apple iPhone, and BlackBerry.

### *1.1 Security threats and treads of Smartphone*

Smartphone are increasingly becoming a target of security threats [3]. First, the number of attacker performing browser attack is increasing recent year, whose targets are many different kinds of smartphone's applications. There is one kind of Trojan that can infect users' web searching engine and modify web pages or transactions. Some approaches can be used to protect users from this kind of attack, such as transaction validation, site to client authentication, security code evolution and etc. The providers of smartphone's applications should take more responsibility to protect their users from these attacks. Second, when social network application becomes more and more popular, the importance of security and trust attracts more attention. Security refers to the issue that whether users' private information can be well protected from other illegal accessing. Trust refers to the issue that whether people in the social network provide their real information. To address these issues, a lot of methods, such as strong authentication, account control and protecting application layer attacks, should be added into this kind of applications.

Third, data in form of files are more vulnerable than database records, since files are independent from each other in most of times and hard to be tracked. To perform a high level of accessing control, data safety should be paid more attention.

Fourth, the number of mobile malwares attacking the platform of the device or applications is growing very fast in recent years. Therefore, more sophisticated encryption methods, such as anti-virus software and authentication method should be used. Moreover, application developer should work together to develop more reliable applications.

### 1.2 Middleware used for enhancing communication between web services and smartphones.

Middleware Architecture is mainly used in Distributed Computing system. Distributing Computing Systems "consist of multiple processors that do not share primary memory, but sending messages over network". Mobile clients are distributed computers that connect to the middleware.

Network communication: Hosts who need to communicate with each other involves some transport layer (TCP and UDP) and marshaling, a process of converting data structure to transferable format.

A. **Coordination**: Since distributed systems have multiple points of control, different components need to coordinate and collaborate through synchronization.

B. **Reliability:** Requests maybe lost during the network transmission. The middleware needs to deploy error detection and correction mechanisms to enhance reliability.

C. **Scalability:** Distributed systems deal with client interactions and also interact between distributed components. Changes in the allocation of components could affect the system architecture, which refers as transparency in the reference model of open distributed processing.

**Manuscript received on May, 2013**.

**Mandeep Singh**, Assistant Professor, Depart of Computer Science, Chandigarh Engineering College, Punjab Technical University, Mohali, India.

**Kanwalvir Singh Dhindsa,** Associate Professor, BBSBEC, Punjab Technical University, Mohali, India.

D. **Heterogeneity**: Components in a distributed system can be implemented with different languages and deployed on different platforms. Thus, the design needs to consider a heterogeneous environment.

Middleware Architecture is often used to extend functions for thin clients, like smartphones. Uribarren et al. [1] proposed a middleware for adaptation in mobile environments. The proposed middleware hides the complexity of deploying ubiquitous applications. Applications are automatically moved between different platforms. Fig 1 shows the middleware for smartphones over the cloud.

The proposed middleware for Smartphone based devices mostly focus on application and content adaptation. Coordination, scalability, reliability, and heterogeneity are four fundamental requirements for general middleware as well as middleware for mobile device. Scalability can be achieved with distributed middleware [9]. Context can help middleware to adapt to the heterogeneous environment. However, the goal of the paper is to use middleware to improve the interaction between mobile clients and internet services as well as use Cloud platforms to improve the scalability of the middleware.

Middleware act s as proxy that is hosted on the Cloud platforms which provide mobile clients access to Cloud services. The middleware architecture will improves interaction between mobile clients and Cloud Services. The middleware also provides extended functions to mobile clients. In general, the architecture enhances the functionality, reliability and compatibility of the interaction between smartphones and Cloud Services. The middleware overcomes the given below and also enhance the interaction between mobile clients and Web Services.

**A. No Loss of connection**: Client and middleware caching – Copies of service results are stored on both mobile clients and the middleware. When the mobile clients are not able to connect to the middleware, the client-side cache is used. When the middleware to server connection is not available, the middleware returns its cached data to the mobile clients.

**B. Bandwidth/Latency**: Protocol transformation – Protocol transformation reduces the latency as well as bandwidth of the client to service interaction. The middleware transforms Simple Object Access Protocol to a much light- weight format RJSON through RESTful Internet Web Services. Transferring SOAP to light-weight protocols, like RESTful, reduces rocessing time as well as the size of the messages.

**C. Result optimization**: Result optimization reduces the size of the service results, thus reduces the bandwidth used to interact with internet Services. The middleware converts the format of service results from XML to RJSON and removes unnecessary data from the original service result. Less data transferring also reduces network latency.

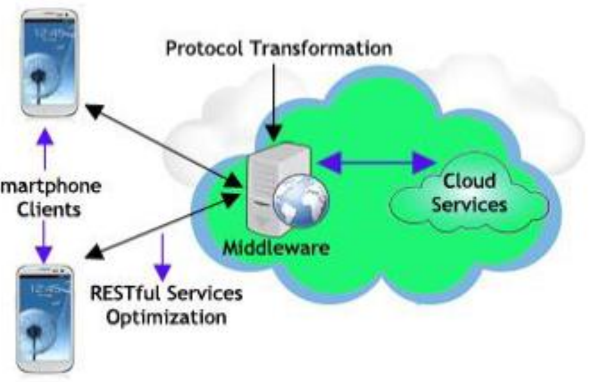

Fig 1: Architecture for Smartphones Middleware

**D. Data Security**: The data will mainly flow from the middleware to the smartphones clients in the form of either XML or RJSON format. The lightweight format for the smartphones[4]. The data which flows from middleware to the smartphones can be either XML in case of SOAP and JSON if we are using the RESTful applications. There is a great need to encrypt the information which flows across the wireless network and which is available in the smartphone applications. Android based smartphones have been taken as an example and its security concerns tackled.

### 1.3 Consuming RJSON services in Android applications

Each time an android application is written chances are your application would need to connect to the outside world to fetch some data, such as live currency exchange rates, weather information, records from databases, etc. One of the easiest ways for your application to connect to the outside world is to use web services.

For the past few years, XML web services have dominated the arena for web services, as XML was touted as the ubiquitous medium for data exchange. However, using XML as the medium for your data payload suffers from the following problems:

- XML representation is inherently heavy. The use of opening and closing tags add a lot of unnecessary weight to the payload. In the world of mobile applications, shaving a few bytes off the payload will dramatically improve the performance of applications, not to mention the reduction of data transferred over the expensive 3G and LTE wireless networks. This translates into cost savings for both application developers (who need to subscribe to expensive networks for their web servers) and users (who has limited amount of bandwidth to use per subscription).
- XML representation is difficult to parse. While on the desktop, the DOM (Document Object Model) and SAX (Simple APIs for XML) are the two commonly used method for parsing XML Documents; on the mobile platform using DOM and SAX are very expensive, both computationally and in terms of memory requirements.

In recent years, another data interchange format has been gaining in popularity - JSON, or JavaScript Object Notation. Like XML, JSON is a text-based open standard for representing data, and it uses characters such as brackets "[{]}", colon ":" and comma ",", to represent data. Data are represented using simple key/value pairs, and more complex data are represented as associative arrays. JSON is further optimized to be used in communication for both native and browser based applications through RJSON which is known as the more recursive form of JSON the RJSON.

### 1.4 Cryptographic Algorithms

Cryptography is a science of information security. It is the art of protecting the data. It stores and transmits the information safely over the insecure medium like Internet by encoding text data into a form non recognizable format with the help of various encryption algorithms [5] and only the intended user will be able to convert it into original text. The process which converts original data into the unreadable form is called encryption process. The encrypted data is called cipher text. The reverse of data encryption is data decryption which converts the cipher text back into the original text. Original text is also called plain text. Cryptology is a combination of Cryptography (encryption) and cryptanalysis (decryption) [2].Cryptography algorithms are classified as: Symmetric (private key) algorithm and asymmetric (public key) algorithm. In symmetric algorithms uses only one key for encrypt the data and same for decrypt the data. Asymmetric key algorithm uses two keys, one is used to encrypt the data and other is used to decrypt the data. Length of Key has an important place in Symmetric key encryption. For the same algorithm, encryption using longer key is hard to cryptanalyze means more secure as compared to the one using shorter key. Asymmetric encryption techniques are almost one-thousand times slower than symmetric techniques as they require more computational processing power. Many encryption algorithms are widely available and used in information security [10].

They can be categorized into symmetric (private) and asymmetric (public) key encryption [16]. In Symmetric keys encryption or secret key encryption, only one key is used to encrypt and decrypt data. The key should be distributed before transmission between entities. Key plays an important role. If weak key is used in algorithm then everyone may decrypt the data. Strength of Symmetric key encryption depends on the size of key used. For the same algorithm, encryption using longer key is harder to break than the one done using smaller key. Some examples of such algorithms are RC2, DES, 3DES, AES, etc. Asymmetric key encryption or public key encryption is used to solve the problem of key distribution [8]. In Asymmetric keys, two keys are used; private and public keys. Public key is used for encryption and private key is used for decryption (E.g. RSA and Digital Signatures). Because users tend to use two key: public key, which is known to public and private key which is known only to the user. There is no need for distributing them prior to transmission. However, public key encryption is based on mathematical functions, 3.1

Private Key algorithms with high throughput are suitable for data communication, while public key algorithms with much lower throughput are suitable for private key exchange and authentication. Among all available algorithms, RSA, advanced encryption standard (AES), and Elliptic Curve Cryptography (ECC) which are approved by standard organizations are selected for the study [6].

**A. Rivest Shamir Adleman cipher** : The RSA algorithm is based on the presumed difficulty of factoring large integers, the factoring problem. Here, a product of two prime numbers is published along with an auxiliary value, as the public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but only someone with the knowledge of the prime factors can feasibly decode the message. 1.1 RSA vulnerabilities. The system (N,D,E) is likely to be insecure if (p-1), for the p that is one of the factors of N, is a product of small primes. When RSA is implemented with several key pairs, the implementers often choose to use the same N for all key pairs, thus saving computation time. However, since the private and public exponents together always assist in factoring N, every single member of the system will be able to factor N with his key pair and use that result to invert any public exponent to the corresponding private exponent. So it is necessary to generate a new N value for each key pair.

**B. Advanced Encryption Standard:** Advanced Encryption Standard, also known as Rijndael, is a symmetric block cipher that uses cryptographic keys of 128, 192 and 256 bits to encrypt and decrypt data in blocks of 128 bits.

Table1. Comparison of various Cryptographic Algorithms

| **Algorithm** | **Created by** | **Key Size(bits)** | **Block Size (bits)** |
|---|---|---|---|
| AES | Joan Daemen and Vincent Rijmen, in 1998. | 128<br>192<br>256 | 128 |
| DES | IBM in 1978 | 56 | 64 |
| 3DES | IBM in 1968 | 112 or 168 | 64 |
| Blowfish | Bruce Schbeier in 1993 | 32 - 448 | 64 |

**C. 3DES**: As an enhancement of DES, the3DES (Triple DES) encryption standard was proposed. In this It operates on a 4x4 column major order matrix of bytes and most of the calculations are done in a special finite field. It is Faster than asymmetric key ciphers. • AES 128 bit key usage is faster than ECC 256 bit key usage. standard the encryption method is similar to the one in original DES but applied 3 times to increase the encryption level. But it is a known fact that 3DES is slower than other block cipher methods [12].

**D. BLOWFISH**: It is one of the most common public domain encryption algorithms provided by Bruce Schneier - one of the world's leading cryptologists, and the president of Counterpane Systems, a consulting firm specializing in cryptography and computer security. ZENITH International Journal of Multidisciplinary Research b Blowfish is a variable length key, 64-bit block cipher. The Blowfish algorithm was first introduced in 1993.This algorithm can be optimized in hardware applications though it's mostly used in software applications. No attack is known to be successful against this.

## II. PROBLEM DEFINITION

Android Jelly Bean operating system 4.1.1 is used for implementing various cryptographic algorithms for enhancing security. This section will illustrate the feature and security issue in the Google android smartphone. The first part will introduce the history of the mobile operating system and main features of the smartphone. The second part will analyze security issues of this sort of smartphone.

### 2.1 Features

Android is a famous operating system for mobile device. Its name is from the first developing company, Android Inc. In October 2003, Android Inc. was founded, whose focus is on developing software for mobile devices. After two years, Android Inc. was acquired by Google, and became wholly subsidiary of Google. This was the first signal that Google would expand their services to mobile phone market. Figure 1 shows the images of Android Smartphone devices. Android was revealed on November 5, 2007. On the same day, the news that Open Handset Alliance is founded was announced. This alliance includes many large software, hardware and telecommunication companies, such as Intel, HTC, Motorola, T-Mobile and etc, whose aim is to develop open standards for mobile devices. Table 1 shows the history of Android System.

The most attractive part of Google Android is that Google releases most of source code. Google allows the companies within Open Handset Alliance freely install this operating system. This movement leads to significant growth of the mobile market of Android, from 2.8% market share in 2009 to 48% market share in 2011 in smartphone's market. Google Android has become the bestselling mobile device platform.

Besides that, third-party application developers can use Java, C, or C++ to develop their applications. Google provides online software store whose named is Market. Users can search and download third party applications from this application. Google also provides applications, such as Goolge Voice, Google Goggles. Table 2 shows feature of current Android smartphones

### 2.2 Security Issues

Android also provides application security through "sandbox" which isolates applications from each other. Without permission, one application cannot access to other application's data or private information in the smartphone. Since Android is an open platform operating system, it provides more freedom to the users to install their desire applications. However, it causes the system easier to be attacked at the same time. There are some types of security issues as below.

Sometimes a flaw of software can cause significant matter. The common method to solve the problem is that the developers recognize the flaws and provide update version of software. For example, Skype once was truly careless to store username, contacts and some other private information. Also in late 2010, a research found that there was a flaw in Android that allowed attacker to download files on Secure Digital (SD) card through JavaScript or HTML. The most recent one was that researchers found out that nearly all Android devices had a security hole in their authentication token. It was possible to man-in-the-middle attack to the Android devices. Now Google has already fixed this flaw.

| Version | Codename | API | Distribution |
|---|---|---|---|
| 1.6 | Donut | 4 | 0.1% |
| 2.1 | Eclair | 7 | 1.7% |
| 2.2 | Froyo | 8 | 4.0% |
| 2.3 - 2.3.2 | Gingerbread | 9 | 0.1% |
| 2.3.3 - 2.3.7 | | 10 | 39.7% |
| 3.2 | Honeycomb | 13 | 0.2% |
| 4.0.3 - 4.0.4 | Ice Cream Sandwich | 15 | 29.3% |
| 4.1.x | Jelly Bean | 16 | 23.0% |
| 4.2.x | | 17 | 2.0% |

Table 1: History of Android operating System

The second type of security issue is that malicious applications can steal users' private data. Because some of applications may need user's Permission to access to SD card, send messages, or access contacts, some malicious applications pretend to be innocent to access and steal data. This may cause a serious damage to users, especially when the device stores lots of confidential information. Although these sorts of malwares show up occasionally, Google removes them quickly.

The third type of security issue is Root Trojans. Android default setup is to disable of access root, but many users like to root their mobile device. This increases the potential of being attack. Some malwares can steal users' confidential information or even remotely control the users' device.

Whenever Google finds out these bad applications, it will remove them quickly. But still, this kind of Trojan does not stop, so it is better for users to ensure root safety by themselves.

For users to improve the level of security for their Android phones, first, the simplest and the most effective method is to set password of the device. Before inputting correct the password, attackers cannot access stored information. Fingerprint lock is the most secure method. Since some user may concern about leakage of their biometric information, setting up a password still can well protect the device.

Second, user should not change root Android device. Some users want to download applications from unofficial third-party application store, so they choose to root their Android device. This is a dangerous decision, because it will remove many restrictions and security protections from default setting. Root Android phone also means to open their system-level access, which allows many malware applications attack the device. So unless you are a master of Android, it is a bad choice to root Android phone.

Third, although Google Android Market does not ensure that all their applications are free of malwares, Google will remove that application from Market and remotely remove them from devices if many users report a same malicious application. So downloading applications from official Android Market ensures higher degree of security.

Fourth, there are some anti-virus applications available on the Market. Installing one of popular anti-virus can help users scan bad applications and enhance the security. Fifth, user should ensure the wireless connection is secured and turn off Wi-Fi when they do not use it. Only connecting to familiar wireless network is also a good method to protect the security of device.

Sadelok Application has the ability for users to turn on HTTPS to encrypt all communications with the site. However, some apps require users to switch to a regular HTTP connection to use the app, but don't warn users that the switch then becomes permanent. Security can be provided by using the Secure Sockets Layer (used in HTTPS) on mobile platforms there should be an way for our mobile users to begin encrypting traffic using encrypted Wi-Fi networks. For this it is high time to compare JavaScript based different wireless communication to encrypt the RJSON data. JSCrypt crypto library of the JavaScript is used in the experiments for comparison.

## III. EXPERIMENTAL SETUP

The various symmetric encryption algorithms [11] have been implemented suing the JavaScript crypto library used to encrypt and decrypt the lightweight RJSON and the experiment has been carried out on a middleware machine hosted on cloud. In this experiment software encrypts & decrypts the JSON file of different sizes.

As we use the lighter weight JSON for to transmit the information from the middleware to the smartphones and also within the smartphones. We can use the JavaScript based cryptographic techniques to secure the information with the use of many encryption algorithms.

To secure our Android app, “Sadelok”, A Punjabi newspaper Application all data connections and data transfers from and to the client application, to and from the server has to be authenticated, encrypted when sent and decrypted when received using different algorithms that takes care of the key exchange problem. The app will help in blocking calls and messages from untrusted contacts stored as blacklists in the client application which are synchronized to the user’s account in the server. For different algorithms, the length of the input block, the output block and the state is 128 bits to 256 bits. This is represented by Nb = 4, which reflects the number of 32-bit words. The length of the cipher key, K, is 128 bits which is represented by Nk = 4. The number of rounds to be performed during the execution of the algorithm is dependent on the key size which is represented by Nr = 10.

The block diagram of the cipher is described in the Fig. 1. The algorithms take the cipher key, K, and perform a key expansion routine to generate a key schedule. The cipher transformations can be inverted and then implemented in reverse order to produce a straight forward inverse cipher for different algorithms. The individual transformations used in the inverse cipher – InvShiftRows ( ), InvSubBytes ( ), InvMixColumns ( ) andAddRoundKey ( ). Fig. 3 describes the data flow diagram for “Sadelok”. The user data includes the blacklisted contacts which includes a name and a number.

## IV. IMPLEMENTATION

### 1.5 Compared Algorithms

This section intends to give the readers the necessary background to understand the key differences between the compared algorithms.

**a) DES:** (Data Encryption Standard), was the first encryption standard to be recommended by NIST (National Institute of Standards and Technology). It is based on the IBM proposed algorithm called Lucifer. DES became a standard in 1975. Since that time, many attacks and methods recorded that exploit the weaknesses of DES, which made it an insecure block cipher.

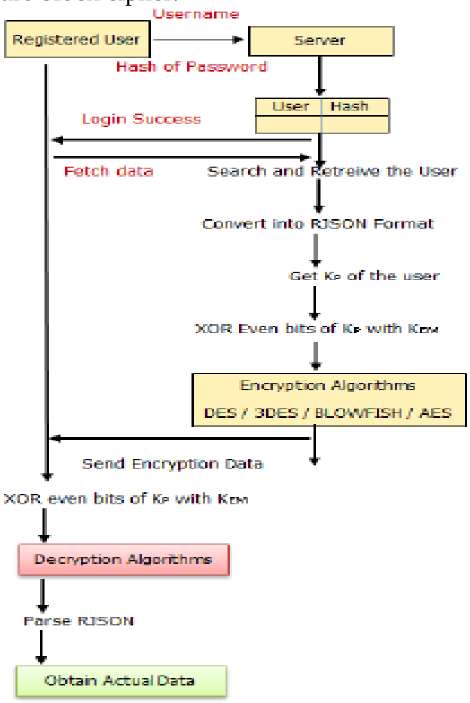

Figure 3: Flow Diagram of Sadelok Application.

**b) 3DES**: As an enhancement of DES, the3DES (Triple DES) encryption standard was proposed. In this standard the encryption method is similar to the one in original DES but applied 3 times to increase the encryption level. But it is a known fact that 3DES is slower than other block cipher methods.

**c) AES**: (Advanced Encryption Standard), is the new encryption standard recommended by NIST to replace DES. Rijndael (pronounced Rain Doll) algorithm was selected in 1998 after a competition to select the best encryption standard. Brute force attack is the only effective attack known against it, in which the attacker tries to test all the characters combinations to unlock the encryption. Both AES and DES are block ciphers.

**d) BLOWFISH**: It is one of the most common public domain encryption algorithms provided by Bruce Schneier - one of the world's leading cryptologists, and the president of Counterpane Systems, a consulting firm specializing in cryptography and computer security. Blowfish is a variable length key, 64-bit block cipher [7]. The Blowfish algorithm was first introduced in 1993.This algorithm can be optimized in hardware applications though it's mostly used in software applications. No attack is known to be successful against this

### 1.6 Key sharing algorithm

The mobile app and the server will already have a key, KEM, is 128 bits which is hardcoded. For the Authentication of the user, MD5 hashing technique is used which is of length 128 bits. The hashed password of the user (application) is represented as KP. In order to generate round keys for the AES technique, an initial key, KI, has to be obtained. This is key can be formed by XOR-ing the even bits of the hashed key, KP,with the embedded key, KEM. Equation (1) represents the same.

### 1.7 Performance metrics

The performance metrics are encryption time (milliseconds), decryption time (milliseconds) and throughput (Mb/sec.). Fig. 2: Performance Metrics [14] of Encryption Algorithms. The performance metrics analyzed and discussed by the researchers regarding encryption algorithms are discussed below:

**Encryption Time:** It is the time that an encryption algorithm takes to produce a cipher text from a plain text. Encryption time is used to calculate the throughput of an encryption process. In other words, it indicates the speed of the encryption process. The encryption time is generally calculated in milliseconds. It is the time taken by an encryption algorithm to encrypt the data. Less is the encryption time; more will be performance of that algorithm [12].

**Decryption Time:** It is the time that an encryption algorithm takes to produce a plain text from a cipher text. Decryption time is used to calculate the throughput of a decryption process. In other words, it indicates the speed of the decryption process. The decryption time is generally calculated in milliseconds. It is the time taken by an encryption algorithm to decrypt the data. Less is the decryption time; more will be performance of that algorithm.

**Throughput:** The throughput of the encryption scheme is calculated as the total plain text in encrypted in Kbytes divided by the encryption time in milliseconds. The unit of throughput is MB/Sec. More is the throughput; more will be the performance [15]. The throughput of the encryption scheme is calculated as the ratio of total plain text by encryption time [13].

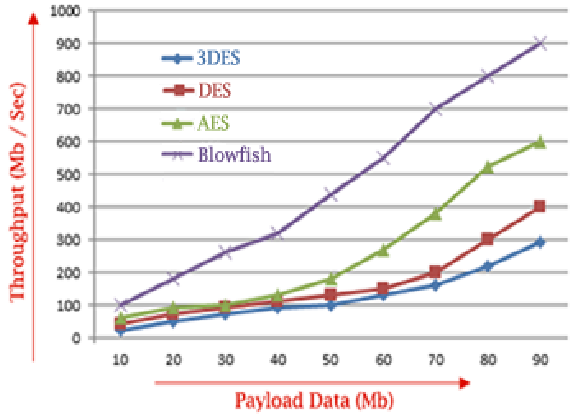

Figure 4: Throughput of the Encryption process on RJSON

$$\textit{Throughput} = \frac{Tp\ (Kbytes)}{Et\ (Milliseconds)} \quad - \quad \textit{Equation: 1}$$

Where;

*Tp* = Total Plain Text (Kbytes)

*Et* = Encryption Time (Milliseconds)

*Equation:1* represents the formulae to represent the Throughput of the Encryption as well as the decryption process. Figur;4 shows the comparison of the throughput of the encryption proves of all algorithms and Figures shows the throughput of the decryption process of all the algorithms. The results establish the superiority of the Blowfish algorithms for encrypting and decrypting the RJSON data over wireless networks. All the above parameters are taken into account while calculating the Throughput of the Encryption as well as Decryption Process.

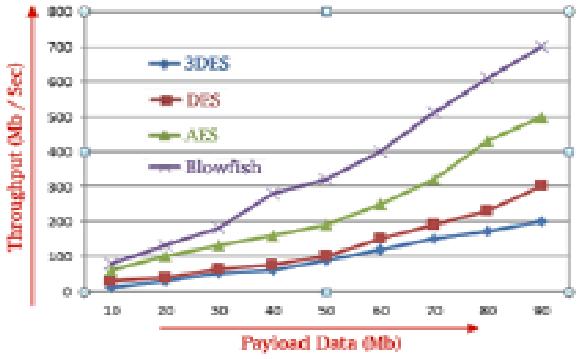

Figure 5: Throughput of the Decryption process on RJSON.

## V. RESULTS

Four algorithms AES, DES, Blowfish and 3DES were taken to encrypt and decrypt the RJSON data. The encryption time and decryption time were calculated and finally the throughput of encryption time and throughput of the decryption time were calculated for all four algorithms. Results conclude that the Blowfish is the best of all the four algorithms studied as it has minimum encryption time and maximum throughput due to its better performance
Followed by AES, DES and 3DES.

## VI. CONCLUSION

The comparison between AES, DES, 3DES and Blowfish in terms of Encryption time, Decryption time and Throughput. The presented simulations showed that Blowfish has better performance in terms of Encryption time, Decryption time and Throughput. Second point can be noticed here that AES has advantage over the other 3DES and DES in terms of throughput & decryption time except Blowfish. In future the work may be extended by including the schemes and techniques by studying the different data packet sizes over different types of data such as image, sound and video and developing a stronger encryption algorithm with high speed and minimum energy consumption.

## REFERENCES

[1] A. Uribarren, J. Parra, J.P. Uribe, M. Zamalloa, and K. Makibar, “Middleware for Distributed Services and Mobile Applications,” InterSense '06: Proceedings of the first international conference on Integrated internet ad hoc and sensor networks, New York, NY, USA: ACM, 2006.

[2] William Stalling, “Cryptography and Network Security Principles and Practice 5th Edition”, Pearson. 30th International Conference on, 2008, pp. 31–40.

[3] Hardjono, “Security in Wireless LANs and MANs”, Architect House Publishers, 2005.

[4] E. Oliver, “A survey of platforms for mobile networks research,” SIGMOBILE Mob. Comput. Commun. Rev., vol. 12, 2008, pp. 56–63.

[5] SimarPreet Singh, and Raman Maini (2011), “Comparison of Data Encryption Algorithms”, International Journal of Computer Science and Communication Vol. 2, No. 1, January-June 2011, pp. 125-127.

[6] Mingyan Wang, Yanwen Que (2009),“The Design and Implementation of Passwords Management System Based on Blowfish Cryptographic Algorithm”,

[7] Challa Narasimham , Jayaram Pradhan (2008), “Evaluation of Performance Characteristics of Cryptosystem Using Text Files”, Journal of Theoretical and Applied Information Technology, pp 254-259.

[8] Tingyuan Nie, Chuanwang Songa and Xulong Zhi (2010), “Performance Evaluation of DES and Blowfish Algorithms”, Proceedings of 2010 IEEE International Conference on Biomedical Engineering and Computer Science (ICBECS- 2010), 23-25 Apr 2010. pp 1-4.

[9] H.E. Bal, J.G. Steiner, and A.S. Tanenbaum, “Programming languages for distributed computing systems,” ACM Comput. Surv., vol. 21, 1989, pp. 261–322.

[10] A.Rathika, Parvathy Nair and Parvathy Nair (2011), “A High Throughput Algorithm for Data Encryption” International Journal of Computer Applications (0975 – 8887) Volume 13, No.5, January 2011 pp 13-16.

[11] M.Umaparvathi, Dr.Dharmishtan and K Varughese (2010), “Evaluation of Symmetric Encryption Algorithms for MANETs”, Proceedings of 2010 IEEE International conference on Computational Intelligence and Computing Research, 28-29 Dec. 2010, pp 1-3.

[12] Allam Mousa and Ahmad Hamad (2006), “Evaluation of the 3DES Algorithm for Data Encryption”, International Journal of Computer Science & Applications Vol. 3, No.2, June 2006, pp 44-56.

[13] Diaa Salama Abdul. Elminaam, Hatem Mohamed Abdul Kader and Mohie Mohamed Hadhoud (2008), “Performance Evaluation of Symmetric Encryption Algorithms”, IJCSNS, VOL.8 No.12, December 2008, pp 280-286.

[14] Allam Mousa and Ahmad Hamad (2006), “Evaluation of the RC4 Algorithm for Data Encryption”, International Journal of Computer Science & Applications Vol. 3, No.2, June 2006, pp 44-56.

[15] Lavanya P and M Rajashekhara Babu (2011), “Performance Analysis of Montgomery Multiplication Algorithm for Multi-core Systems Using Concurrent International Journal of Computer Applications (0975 – 8887) Volume 44– No11, April 2012.

[16] Y. Kumar, R. Munjal and H. Sharma, Comparison of Symmetric and Asymmetric Cryptography with existing vulnerabilities and counter measures, International Journal of Computer Science and Management Studies, 11( 03), Oct 2011.