**CHAPTER TWO**

**LITERATURE REVIEW**

This chapter presents a review of the concepts that are important for obtaining basic understanding of the ideas on the research area. The review aims at understanding fundamental points for designing a secure text messaging (SMS) framework. It includes review of the fundamentals of GSM and SMS. It also reviews different encryption techniques. In addition, the review on the state of the art of SMS security will be presented.

**2.1 Second Generation (2G) Networks**

In 1980s and 1990s, wireless communications comes to popularity and also increases the usability in different areas. Since there are a number of drawbacks of analogue 1G system (Stalling, 2005), because of this, second generation (2G) wireless system was developed. 2G network is based on digital technology. Unlike analog signal in 1G, digital signals use binary coding using sequences of 0s and 1s. Digital signals use digital samplers and codecs (coder-decoder) to convert analogue data into digital data. There are three main advantages of 2G over 1G network (Stalling, 2005). The first one is, it introduces data encryption between mobile phones (Stalling, 2005). In the GSM, only the airway traffic between the Mobile Station (MS) and the Base Station Transceiver (BTS) is optionally encrypted with a weak and broken symmetric stream cipher A5 algorithm (Stalling, 2005). The second advantage is, 2G systems were significantly more efficient and simple than 1G (Deitel et al., 2002); and 2G introduced data services for mobile, starting with SMS text messages.

GSM, also known as second generation network or 2G, was first developed in the 1980s by European Telecommunications Standards Institute (ETSI). Original GSM was developed for Europe; the 2G technology is used in more than two hundred countries including Ethiopia in the world today (Elliott and Phillips, 2004). With GSM it was also made possible to send and receive limited amounts of data via the Short Messaging Service (SMS) and mobile Internet browsing via the Wireless Applications Protocol (WAP) (Elliott and Phillips, 2004).

**2.1.1 Overview of 2G Network Architecture**

A GSM network is composed of three main functional components; the mobile phone handset or Mobile Station (MS), the Base Station (BS) and Network Subsystem (NS). The next section describes each component in detail. Figure 2.1 shows an overview of the key elements of a cellular network.

**2.1.1.1 Mobile Station**

This part of GSM network consists of the mobile phone (terminal) and a smart card called the Subscriber Identity Module (SIM). The SIM provides personal mobility, so that the user can have access to subscribed services irrespective of a specific terminal. By inserting the SIM card into another GSM terminal, the user can be able to receive calls at that terminal, make calls from that terminal, and receive other subscribed services.

The mobile equipment is uniquely identified by the International Mobile Equipment Identity (IMEI) (Elliott and Phillips, 2004). The SIM card contains the International Mobile Subscriber Identity (IMSI) used to identify the subscriber to the system, a secret key for authentication, and other information. Customers of monthly paid contracts need to register their personal and other necessary information such as bank details with their service provider; whereas for a pre-paid contract registration is not always necessary (Stalling, 2005) and all registered customer information is identified by its IMSI. The IMEI and the IMSI are independent, thus allowing personal mobility.

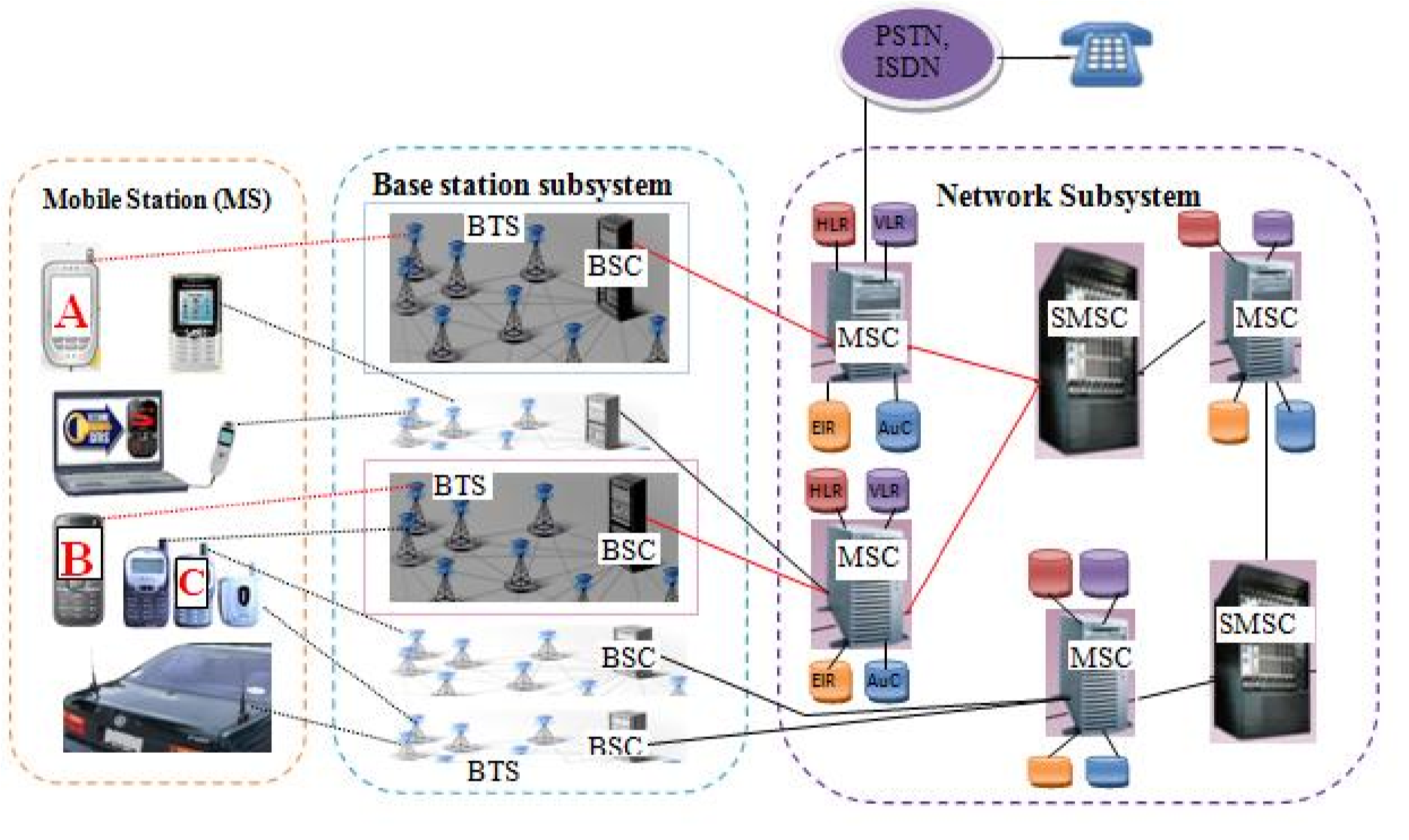


Figure2.1: GSM Network Architecture Overview

**2.1.1.2 Base Station Subsystem (BSS)**

The base station subsystem is composed of two parts, the Base Station Transceiver (BTS) and the Base Station Controller (BSC). The Base Station Subsystem (BSS) is responsible for handling traffic and communications between a Mobile Station (MS) and the Network Subsystem. The BSS consists of two elements, one or more Base Station Transceivers (BTS) and the Base Station Controller (BSC) (Elliott and Phillips, 2004). Each Base Station Transceiver includes a radio antenna which handles the radio-link protocols with the mobile phone and a link to one of the Base Station Controllers. The Base Station Controller manages allocation of radio channels, reservations of radio frequencies and handovers of mobile phone handsets from one cell to another within the BSS. The BSC can either be located with a BTS or can control multiple BTS units, hence serves multiple cells. A group of BSCs is connected to a mobile switching centre (MSC) via microwave links or telephone lines (Hubaux et al., 2000).

**2.1.1.3 Network Subsystem (NS)**

The Network Subsystem (NS) includes Mobile Services Switching Center (MSC) and Sort Message Service Center (SMSC) as subcomponent. Home Location Register (HLR), Visitor Location Register (VLR), Authentication Centre database (AuC), and Equipment Identity Register database (EIR) are databases used in this sub-system (Elliott and Phillips, 2004). Network Subsystem provides a link between the cellular network and fixed networks such as the analogue Public Switched Telephone Network (PSTN) or digital Integrated Services Digital Network (ISDN). The NS controls handovers between cells in different Base Station subsystems, authenticates users and validates their accounts. It also provides functions for worldwide roaming of mobile users. In the next subsection we discuss about both components of NS in detail.

**2.1.1.3.1 Mobile Services Switching Center (MSC)**

MSC is the main part of the Network Subsystem (NS) (Elliott and Phillips, 2004). The MSC performs the switching of calls between the mobile and other fixed or mobile network users, as well as the management of mobile services such as registration, authentication, location updating, handovers, and call routing to a roaming subscriber. It also performs such functions as toll ticketing, network interfacing, common channel signaling, and others (Elliott and Phillips, 2004). Every MSC is identified by a unique ID. MSC uses four databases in order to perform its functionality.

**2.1.1.3.1.2 Home Location Register Database (HLR)**

This contains an entry for every SIM card issued, including details such as telephone number, mobile equipment number, equipment type and subscription type (Elliott and Phillips, 2004). Basically, GSM networks have only one HLR. In addition, dynamic information about the mobile subscriber is stored, for instance the current Location Area (LA). A Location Area consists of one or a number of cells. As soon as a mobile phone user leaves his/her current LA, this temporary data, stored in the HLR, is immediately updated when the user changes the current location (Steele, Lee, and Gould, 2001).

**2.1.1.3.1.3 Visitor Location Register Database (VLR)**

This is a temporary database that maintains information about subscribers that are currently physically in the region covered by the Mobile Switching Centre (MSC) (Steele, Lee, and Gould, 2001). Entries are added when users enter the VLR domain and deleted when users leave the VLR‟s domain. The VLR stores information transmitted by the HLR, such as authentication data, telephone number, agreed services, allowing the MSC to make a connection. It temporarily stores a user’s last known Location Area and records whether or not the subscriber is active and other parameters associated with the subscriber. The VLR also contains information that enables the network to find a particular subscriber in the event of an incoming call.

**2.1.1.3.1.4 Authentication Centre Database (AuC)**

The Authentication Centre database (AuC) handles authentication and encryption keys for all subscribers in the home and visitor location registers (Steele, Lee, and Gould, 2001). It stores data needed to authenticate a call and to encrypt both voice and data traffic. This encryption is only used for wireless communication which is from Mobile Station subsystem up to Base Station Transceiver (BTS). Since, GSM network is composed of wireless and wired network, the encryption does not include wired part of the network such as data stored on MSC.

**2.1.1.3.1.5 Equipment Identity Register Database (EIR)**

The Equipment Identity Register database (EIR) lists stolen phones, stores subscriber and equipment numbers (IMEI) of phones that are to be excluded from the network (Steele, Lee, and Gould, 2001). It can block calls from stolen mobile stations and prevent network use by handsets that have not been approved (Walke, Seidenberg, and Althoff, 2003).

**2.1.1.3.2 Short Message Service Centre (SMSC)**

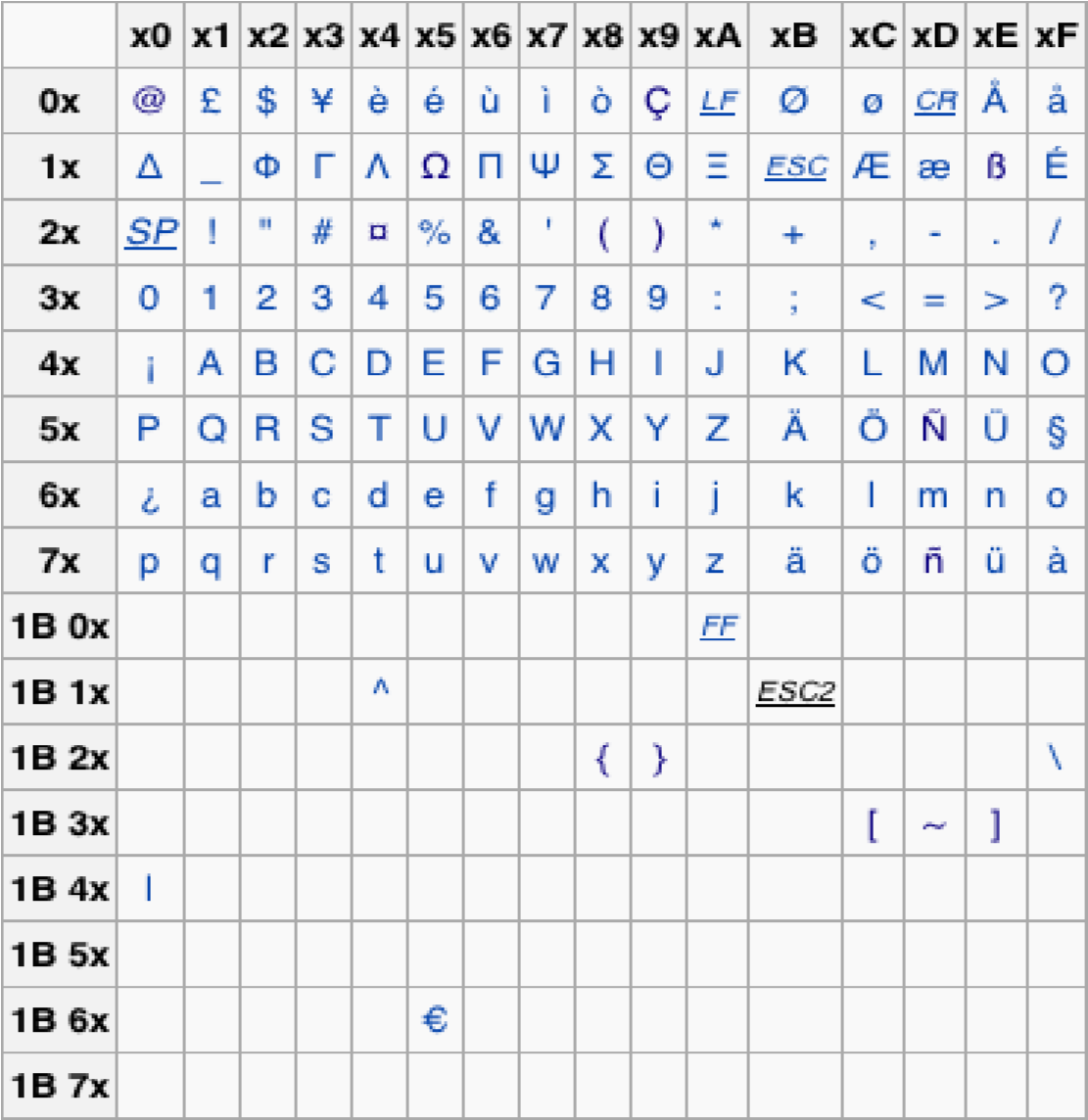
Short Message Service is a mechanism of delivery of short messages over the mobile networks.

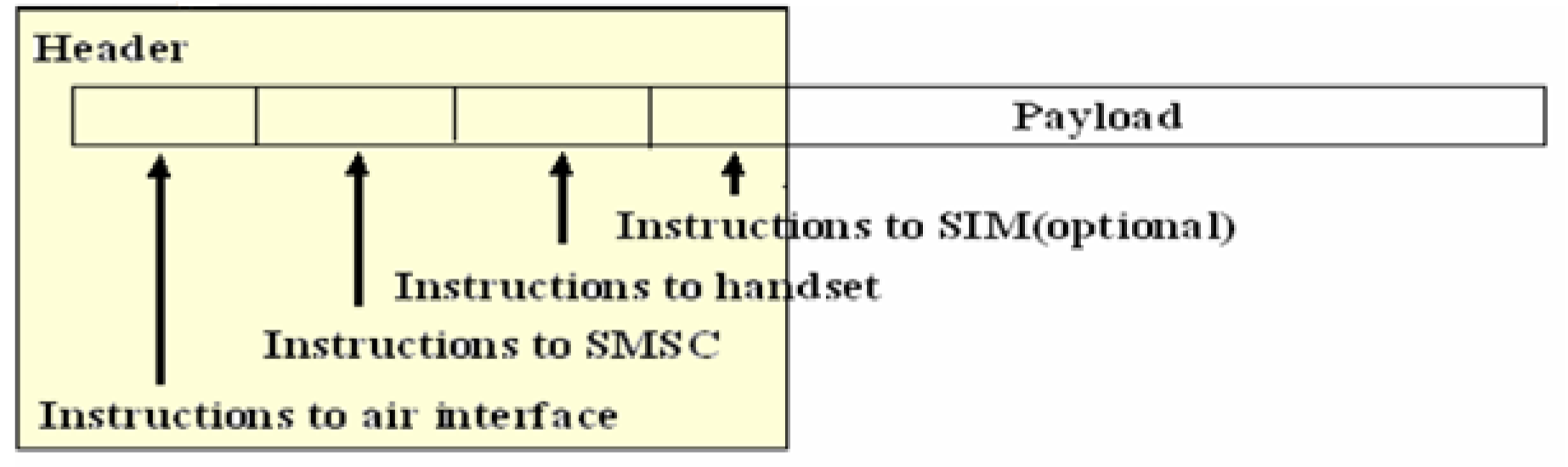
It is a store and forward way of transmitting messages to and from mobiles. The Short Message Service Center (SMSC) is part of GSM network which is responsible for the replaying, storing and forwarding of a short message (Steele, Lee, and Gould, 2001). A Short Message Service Centre (SMSC) is usually owned and run by a telecommunication operator. An SMS message may pass through a number of SMSC or other SMS gateways (which act as bridges between two or more SMSCs running different SMSC protocols) before reaching the recipient‟s device. An SMSC helps route SMS messages and manage the process. If the intended SMS recipient is not online, the SMSC will keep the stored SMS message for a “validity period” before deleting it from storage as plain text (Steele, Lee, and Gould, 2001).

An SMS can contain up to 140 bytes. The GSM character set is encoded using 7-bits, rather than the usual 8-bits that make a byte. This means there can be 160 characters in an SMS. This 7-bit encoding limitation means only 128 standard characters can be encoded. The GSM standard gets round this by also having the extended GSM character set. There are another 10 characters which are actually sent by sending two 7-bit characters, an escape (ESC) character followed by another character. It is also possible to send characters (such as Amharic characters) that are not in the GSM character set, but each character represented by two bytes, it means only 70 characters can be sent per SMS. Table 2.1 shows a full list of standard and extended GSM characters (www.clockworksms.com/blog/the-gsm-character-set).

In practical use, SMS messages are sent over the GSM network; however, this SMS can be altered, intercepted or spoofed in between the network (Otto and Virtanen, 2004). This threatens the confidentiality, integrity and availability which are the key elements for providing a secure service for any type of communication over any network that uses either wired or wireless medium. For those applications that require secure transmission of a message, such as SMS-Banking and SMS-Voting, end-to-end encryption is advisable between the sender and the recipient. Figure 2.2, shows different SMS headers and the payload (which is the content of the message). The header provides four bytes to specify metadata and the size of the payload (Nokia Forum, 2013). The maximum length of the payload is usually 160 characters at 7-bit per character. The encryption, which is the effort of this research, is conducted on the payload part of the SMS.

**Table 2.1:** List of Standard and Extended GSM Characters (www.clockworksms.com/blog/the-gsm-character-set)





**Figure 2.2:** SMS Message Structure (Clements, 2018)

**2.1.2 Steps for SMS Communication**

To demonstrate how the cellular architecture works, below is what happens when two mobile phones controlled by the same mobile switching center (MSC) communicate using SMS (Walters, and Kritzinger, 2016).

**Step 1:** The Mobile Station (MS) is powered on and registered with the network, which is used to identify the mobile phone user and to register its location.

**Step 2:** The Mobile Station transfers the short message to the MSC.

**Step 3:** The MSC accesses the Visitor Location Register (VLR) to verify that the message fulfills the requirements and it does not violate restrictions.

**Step 4:** The MSC sends the short message to the SMSC using the „forward and store‟ operation.

**Step 5:** After completing its internal processing, the SMSC accesses the Home Location Register (HLR) to know the current location of the receiver mobile station.

**Step 6:** If the required phone is available, the SMSC sends the short message to the MSC using the „forward short message‟ operation. If the receiver phone is not available, the SMSC stores the message until the phone is available or the sated time expires.

**Step 7:** If the receiver’s phone is available to identify receiver’s current location, MSC retrieves the subscriber information from the VLR. This operation may include an authentication procedure.

**Step 8:** The MSC transfers the short message to the appropriate Mobile Station.

**Step 9:** The MSC returns to the SMSC the outcome of the “forward short message” operation used for delivery report.

**Step 10:** The SMSC acknowledges to the MSC the successful outcome of the “forward short message” operation.

**2.2 Message Cryptography (Encryption/Decryption)**

Encryption is the process of converting messages, information, or data into a form which is unreadable by anyone except the intended recipient. Encrypted data must be decrypted, before it can be read by the receiver. In this section, two basic schemes of cryptography will be discussed; symmetric (secret) key cryptography and asymmetric (public) key cryptography.

**2.2.1 Symmetric Cryptography**

Symmetric key cryptography, also known as secret key cryptography, uses a single key for both encryption and decryption. There are various symmetric key algorithms such as DES, Triple DES, AES, RC4, RC6, and BLOWFISH (Diaa et al., 2008). Since, the security of the symmetric encryption algorithm depends on the secrecy of the key, the secret key must be transferred using secure channel. If unauthorized user access the secret key the encryption becomes useless. Figure 2.3 shows the encryption and decryption process of symmetric key cryptography.

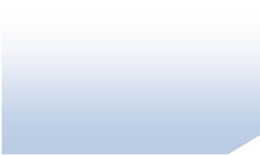
A number of researches were conducted for different secret key algorithms such as DES, 3DES, AES, and Blowfish (Monika, 2012). The results of the study show that, Blowfish had a very better performance (Anjali and Rajeshwari, 2013) compared to other algorithms in terms of power consumption, processing time, and throughput in case of encryption and decryption. Blowfish has no any known security weak points so far, this makes it an excellent candidate to be considered as a standard encryption algorithm.



**Encryption**



**Decryption**



Plain Text

“Abc123”



Plain Text

“Abc123”



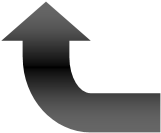
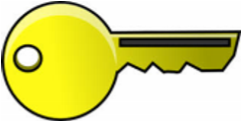
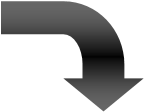
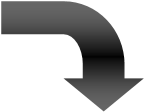
Cipher

Text

“

?h%mqj‟e

”



Shared Secret Key

Figure2.3: Encryption/Decryption with Symmetric or Secret key Cryptography Algorithms (Diaa et al., 2008)

**2.2.1.1 Blowfish Encryption**

Blowfish was designed in 1993 by Bruce Schneier, and it was mainly designed to be used in embedded systems (Diaa et al., 2008). It was designed as a fast, free alternative to existing encryption algorithms. It is a symmetric block cipher that supports 64 bits block size and a variable key size from 32 bits up to 448 bits long. As shown in Figure 2.5, Blowfish algorithm is a Feistel Network, iterating a simple encryption function 16 times. It uses simple operations that are efficient on microprocessors such as exclusive-or, addition, table lookup, modular- multiplication. Blowfish supports variable number of iterations. For applications with a small key size, we can improve the security by increasing the number of iterations. Blowfish consists of two main parts which are key-expansion and data encryption. During the key expansion stage, the actual key is converted into several sub keys and data encryption performs the actual encryption by using 16-round Feistel network. The next section describes both key-expansion and data encryption part of the algorithm.

A Feistel network is a general method of transforming a function called F-function to a permutation. It was invented by Horst Feistel and has been used in many block cipher designs. As shown in Figure 2.5, Feistel Network works as follow (Brent, 2004):

* Split each input block into halves, for example in Blowfish 64 bit input is divided into two 32 bits;
* Right half becomes new left half;
* The new right half is produced by XOR left half with the result of applying F- function the right half and the key;
* Outputs are added modulo 232 and XORed to produce the final 32-bit output.

As shown in Figure 2.4 (Brent, 2004), Blowfish's F-function splits the 32-bits input into four 8-bits quarters, and uses each 8-bits as input to the respective S-boxes for table lookup as shown in Figure 2.4. S-boxes accept 8-bits input and Produce 32-bits output.

* + - 1. **Blowfish Encryption Algorithm**

Data encryption part of Blowfish algorithm begins with a 64 bits block of plain text and that will be converted into a 64 bits cipher text.

Blowfish decryption is exactly the same as encryption, except that P1, P2... P18 are used in the reverse order. As described in Blowfish Key Expansion part below, P1, P2... P18 are sub keys produced from the secret key.

* + - 1. **Blowfish Key Expansion**

Blowfish can support a variable key length from 32 bits up to 448 bits. This key is not directly used for encryption and decryption but, used in the creation of a number of sub keys. All necessary sub keys are produced as follow (Brent, 2004):

1. Initialize the P-array and then the four S-boxes; this process uses the hexadecimal fraction of pi. The hexadecimal fraction of pi was selected because it produced a random number for the initialization. After initialization, the values of the P-array and S-boxes are modified using the key. The key, which can consist up to 448-bits, is segmented into 32-bit values. If the key is less than 448-bits, the key is repeated. For example, if the key was “1234” and each number represented 8 bytes, the first 32-bits value would be “1234” and the next 32- bits value would be “1234.” This process would be repeated to obtain the required 448-bits key. For a 448-bit key, there would be 14 32-bits values, which means the key-array (K) has 4 32bits values less than the P-array.

Figure

2

**.**

4

**:**

Overview of F

-

Function (Brent, 2004).

**S**

**-**

**Box 1**

**S**

**-**

**Box 2**

**S**

**-**

**Box 4**

**S**

**-**

**Box 3**

**32**

**bits**

***∑***



***∑***

=

SUM



=

XOR

***∑***

**bits**

**8**

**8**

**bits**

**8**

**bits**

**8**

**bits**

**bits**

**32**

**32**

**bits**

**32**

**bits**

**32**

**bits**

**32**

**bits**

**bits**

**32**

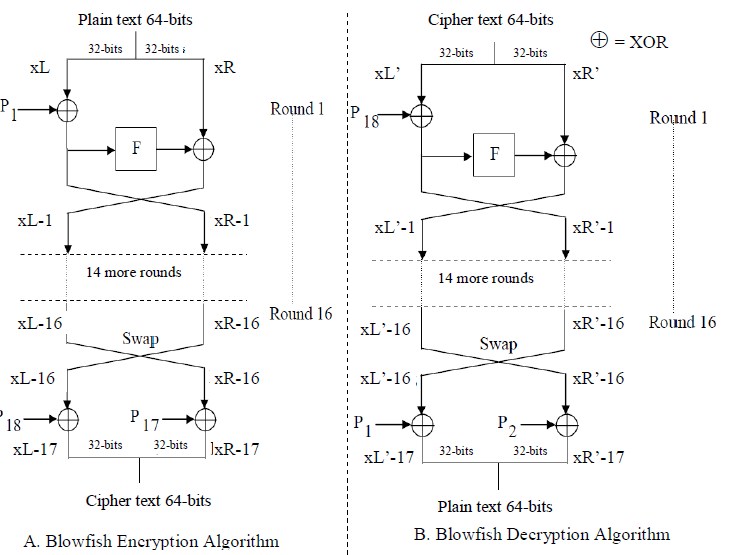


Figure 2.5: Feistel Network for Blowfish Encryption and Decryption (Brent, 2004)

1. XOR P1 with the first 32 bits of the key (K1), XOR P2 with the second 32-bits of the Key (K2), and so on for all bits of the key. This process is continued until P15 because the keyarray has only 14 32-bits. According to [14], an exclusive-or operation would be conducted with P15 and the first 32-bit key value (K1), which is incorrect. The key value is actually the next value of the key repetition, which would essentially be K15. For example, if the original key was “123”, where each number represents 8-bits, and K14 equaled “1231”, the next 32bit key value for the exclusive-or operation with P15 would be “2312” (K15) and not “1231” (K1) as mentioned by Stalling, 2005.
2. Encrypt the all-zero string with the Blowfish algorithm, using the sub keys described in steps

(1) and (2).

1. Replace P1 and P2 with the output of step (3).
2. Encrypt the output of step (3) using the Blowfish algorithm with the modified sub keys.
3. Replace P3 and P4 with the output of step (5).
4. Continue the process, replacing all entries of the P array, and then all four S-boxes in order, with the output of the continuously changing Blowfish algorithm.

**2.2.2 Public Key Cryptography**

As mentioned in symmetric key encryption, there is problem in the distribution of symmetric keys due to the nature of symmetric key encryption, i.e., if someone accesses the secret key he/she can encrypt/decrypt messages, so if the key is stolen through some means the encryption becomes useless. Also, symmetric key encryption needs to manage large amount of keys, i.e., n (n-1)/2 when n is number of users (Ranbir, 2009). Public key encryption was designed to solve this problem by having a key-pair for each user, a public key which is used to encrypt the message at the sender side and a private key used to decrypt the encrypted message at the receiver side as shown in Figure 2.6 (Sonali, and Malik, 2013).

From public key it is computationally infeasible to compute the private key, in this way the private key and public key should be related in such a way that it should not be easy to derive the private key from the public key; this usually by using some mathematical functions such as the factorization of large numbers or the discrete functions. Some of public key algorithms are ECC, RSA, Diffie-Hellman key exchange and DSA. In the later section we will see the comparison of different public key algorithms in order to select the efficient one in terms of encryption/decryption time, and space throughput requirements.

The study in Lokesh and Sonali, 2013 shows that, Elliptic Curve Cryptography (ECC) is the most efficient public key encryption scheme based on elliptic curve concepts that can be used to create faster, smaller, and efficient cryptographic keys. ECC can provide various security services in the form of key exchange, communication privacy through encryption, authentication of sender and digital signatures to ensure message integrity (Al-Bakri et al., 2011). ECC helps to establish equivalent security with lower computing power and battery resource usage, this makes ECC suitable for handheld devices (i.e., cellular phones and PDAs) (Alpesh and Abhilash, 2011). As mentioned in many literatures that a considerably smaller key size can be used for ECC compared to RSA. (Lokesh and Sonali, 2013). Also, mathematical calculations required by Elliptic Curve cryptosystem are easier, hence, require a low calculation power. Therefore ECC is a more appropriate asymmetric cryptosystem to be used on small devices like mobile phones.

The result in Rounak, Hemant, and Sumita, 2013 shows, ECC has better performance on different aspects like key generation time, memory space and encrypt/decrypt time comparing with RSA and they conclude that Elliptic Curve Cryptography is better than RSA on mobile phones.

The security level which is given by RSA can be provided by smaller keys of Elliptic Curve Cryptography, thus, this small key size in ECC reduces processing overhead. As compared to RSA, which offers 1024 bits security strength, ECC offers the same in 160 bits key length (Sonali, and Malik, 2013). Table 2.3 shows Equivalent key sizes of ECC and RSA (Alese, Philemon, and Falaki, 2012). The research conducted in Pardeep, 2011, concludes that, ECC provides higher speeds, smaller power consumption, bandwidth, and storage efficiencies compared with other asymmetric cryptography such as RSA.

**2.2.2.1 Elliptic Curve Cryptography (ECC)**

Elliptic Curve Cryptography (ECC) is a public key encryption technique based on Elliptic Curve theory that can be used to create faster, smaller, and more efficient cryptographic keys. Elliptic curves have been extensively studied in different researches as mentioned in previous section and almost all researches shows that, ECC has better performance in different aspects such as key size, encryption/decryption time requirement, key generation time, and memory requirement. Since, ECC is suitable encryption algorithm to handheld device (phone, PDA) than other public key cryptography (Daniel. 2005), in this study, we propose ECC to make secure channel to transfer one time password (secrete key) and in the next subsection we will see ECC algorithm in details.



**Encryption**



**Decryption**



Plain Text

“

Secure SMS

”



Cipher

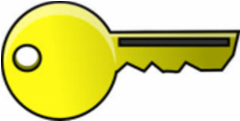
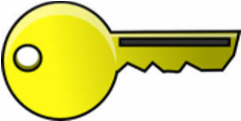
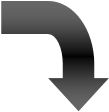
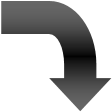
Text

“

1

/o?h%my@t2qj‟e

”



Recei

ver‟s Public

Key

Receiver‟s Private

Key



Plain Text

“

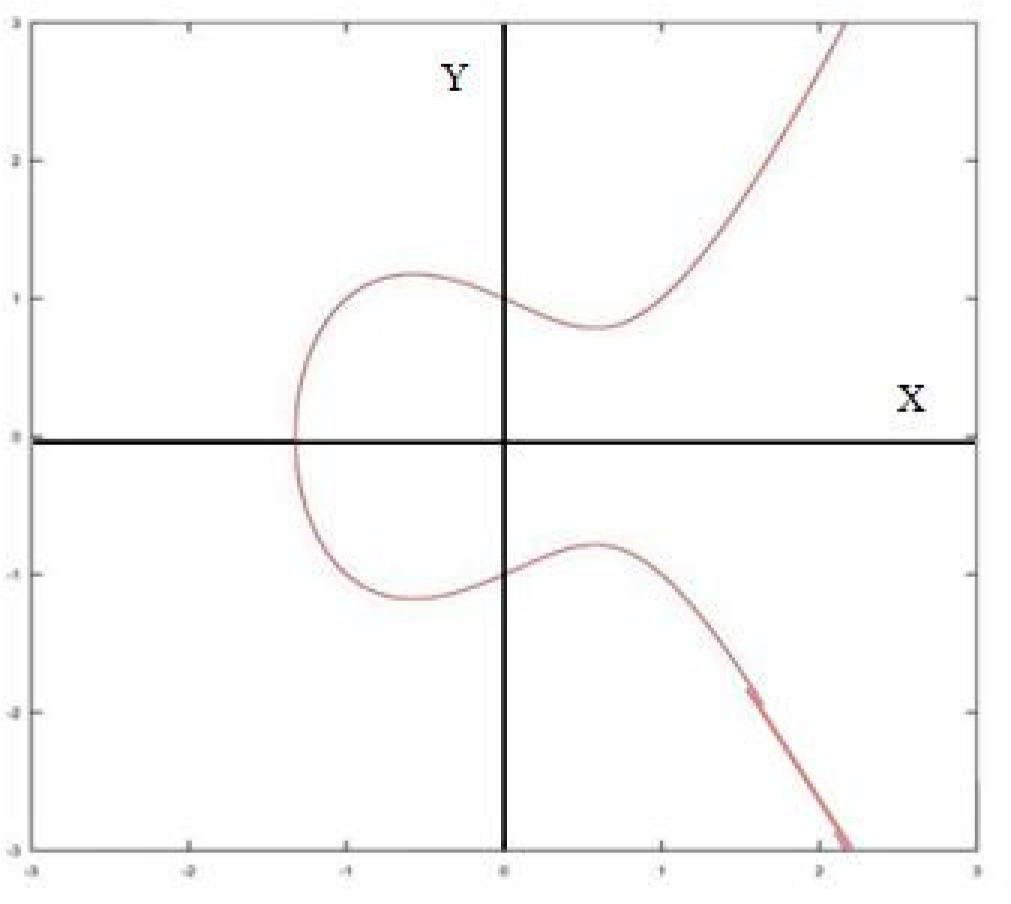
Secure SMS

”

Figure 2.6: Encryption/Decryption with Public Key Cryptography Algorithms (Sonali, and Malik, 2013).

Table 2.3: Equivalent Key Sizes of ECC and RSA (Alese, Philemon, and Falaki, 2012)

|  |  |
| --- | --- |
| ECC key size (bits) | RSA key size(bits) |
| 112 | 512 |
| 128 | 704 |
| 160 | 1024 |
| 192 | 1536 |
| 224 | 2048 |
| 256 | 3072 |



**Figure 2.7:** Example of Elliptic Curve over Real Numbers (Sahoo and Gunamani, 2013)

Elliptic Curve is a collections of points that satisfy an equation with the form “y2 = x3 + ax + b” Figure 2.7 shows an example of elliptic curve over the real numbers where a is –1 and b is 1 (Sahoo and Gunamani, 2013). Some properties of elliptic curve which are relevant for Elliptic Curve Cryptography are listed below (Sahoo and Gunamani, 2013):

* The curve is symmetric around the x axis, so that if (x, y) is a point on the curve, then (x,–y) is also on the curve.
* If we draw a line between any two points on the line with different x coordinates, they will intersect the line at a unique third point and

For each point on the curve, if you draw a straight line tangent to the curve from that point, it will intersect the curve once again at another point. Elliptic Curve Cryptographic algorithms are implemented using different point operations on the elliptic curve. Some of the mathematical operations on elliptic curve which are useful for cryptography purposes are; addition, doubling a point and scalar multiplication. To understand elliptic curves cryptography each mathematical operation are discussed below.

* + - 1. **Point Addition**

Point addition is the addition of two points P and Q on an elliptic curve to generate another point R on the same elliptic curve. As shown in Figure 2.8 (Sahoo and Gunamani, 2013), if Q ≠ -P then a line drawn through the points P and Q will intersect the elliptic curve at exactly one more point –R. Finally, we find the negative of R, which is the result of addition of points P and Q, i.e., R = P + Q. If Q = -P the line drawn through the points P and Q will intersect at infinity O. Hence, P + (-P) = O. O is the additive identity of the elliptic curve group. A negative of a point is the reflection of that point with respect to x-axis. Figure 2.8 shows addition of two points over elliptic curve (Sahoo and Gunamani, 2013).

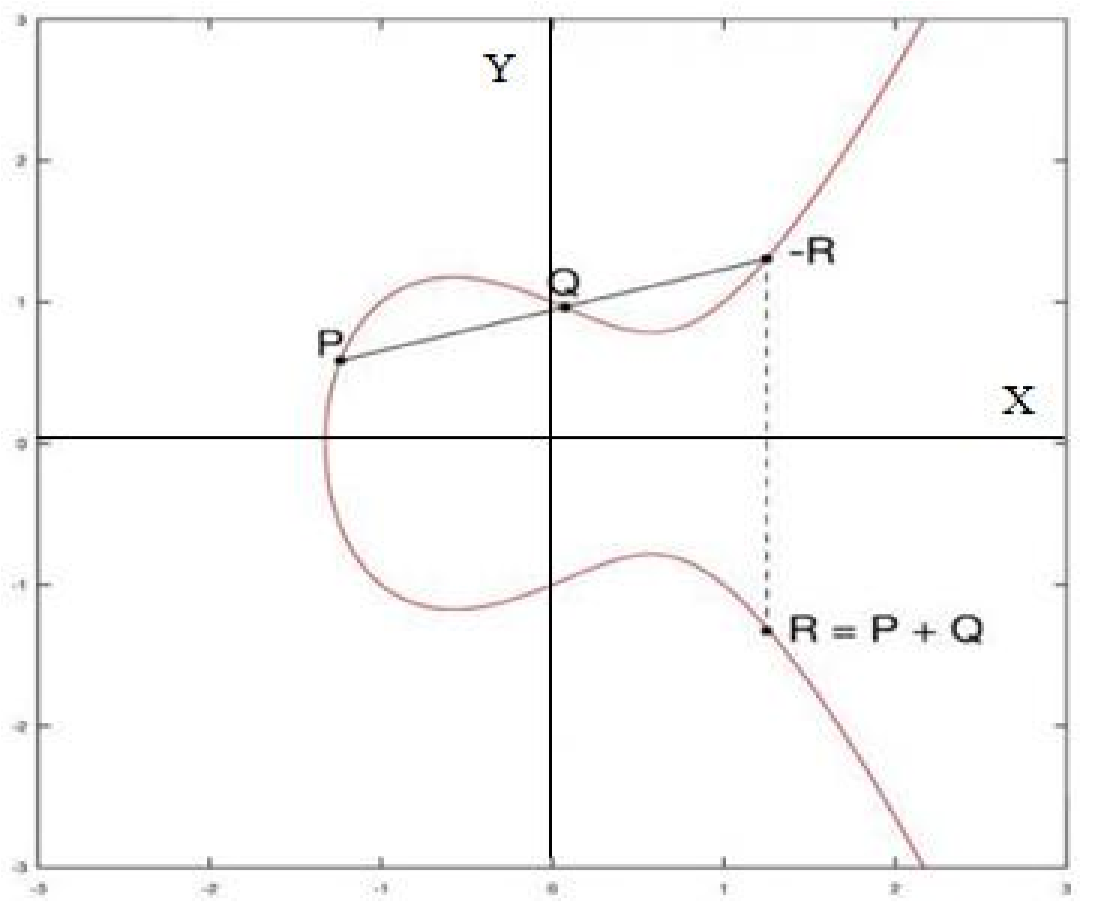


Figure 2.8: Adding Two Points on Elliptic Curve (Sahoo and Gunamani, 2013)

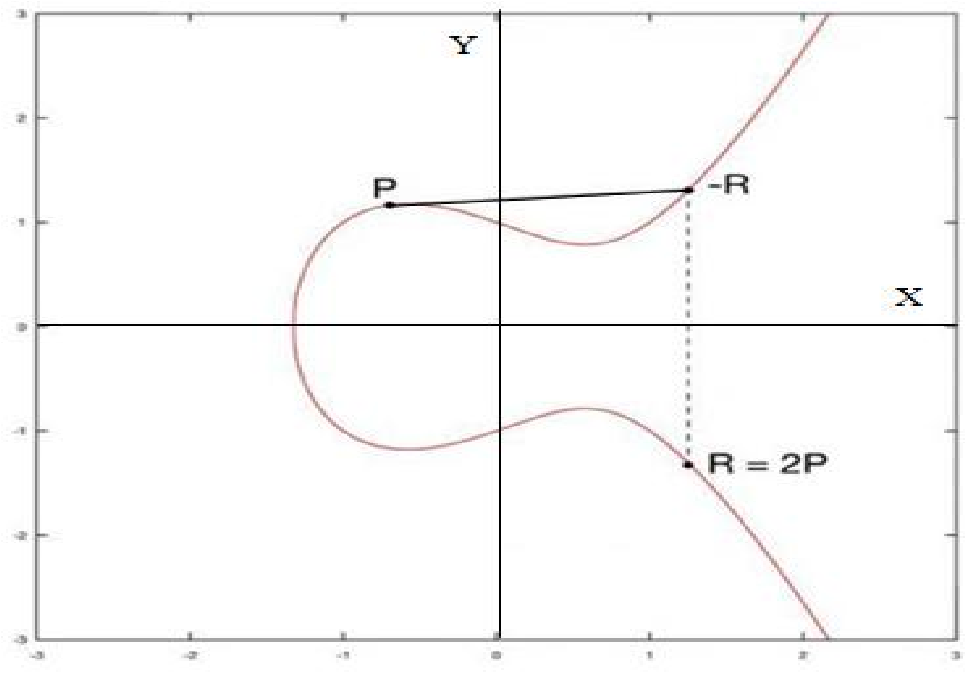


Figure 2.9: Doubling a Point on Elliptic Curve (Sahoo and Gunamani, 2013)

* + - 1. **Point Doubling**

Point doubling is the addition of a point P on the elliptic curve to itself to generate another point R on the curve. To find R= 2P, consider a point P on an elliptic curve as shown in Figure 2.9 (Sahoo and Gunamani, 2013). If y coordinate of the point P is not zero then the tangent line at P will intersect the elliptic curve at exactly one more point –R. The reflection of the point –R with respect to x-axis gives the point R, which is the result of doubling the point P.

**2.2.2.4 Elliptic Curves Over Finite Field ( Fp)**

Let E be an elliptical curve over Fp. From the cryptography point of view, we need elliptic curve group mod p, where p is a prime number (Sahoo and Gunamani, 2013). Then E may be described as 4a3 + 27b2 (mod p) ≠ 0, where a and b are two positive integers less than the prime number p. Then, Ep (a, b) denotes the elliptic group mod p, whose elements (x, y) are pairs of positive integers less then p satisfying y2 ≡ x3 + ax + b(mod p) together with the point at infinity O. As an example, if E is an elliptic curve over F7 described by the equation y2=x3+2x+4, then the points on E are E(F7)={∞, (0,2), (0,5), (1,0), (2,3), (2,4), (3,3), (3,4), (6,1), (6,6)}. Now there is a well-known method for adding two elliptic curve points (x1, y1) and (x2, y2) to produce a third point on the elliptic curve.

**2.2.2.5 Elliptic Curve Key Generation**

Let E be an elliptic curve over finite field Fp. Let P be a point on E (Fp) and suppose that P has prime order n. Then the cyclic subgroup of E (Fp) generated by P is <P>= {∞, P, 2P, 3P……….(n-1)P} (Sahoo and Gunamani, 2013). The prime p, the equation of the elliptic curve E, and the point P and its order n are the public domain parameters. A private key is an integer d that is selected uniformly at random from the range [1, n-1], and the corresponding public key is Q=d\*P.

**2.2.2.6 Elliptic Curve Encryption/Decryption**

The first task in ECC encryption is to encode the plain text message m to be sent as an x-y point Pm. It is the point Pm that will be encrypted as a cipher text and subsequently decrypted. To encrypt and send a message Pm to B, the sender chooses a random positive integer k and produces the cipher text Cm = {kP, Pm + k Q}, where Q is receiver’s public key. The sender transmits the points C1= kP, and C2= Pm + k Q to the recipient.

**2.3 Related Works**

Many security solutions were proposed for mobile messaging services. For clear understanding we group the research done before in to three groups; the first one is SMS security solutions using symmetric cryptographic technique, the second one is SMS security solutions using asymmetric cryptographic technique and the last one is solving SMS security problems by using hybrid encryption.

**2.3.1 SMS Security Solutions using Symmetric Cryptography**

As we can see from different literatures we can understand that many of SMS security solutions are based on symmetric key cryptography (Rohan, Sanket, and Priyanka, 2012) due to limited infrastructure of mobile devices: such as limited memory, processing capacity and power supply (Raghavendra, Sunanda, and Maruthi P, 2011). Most of the proposed symmetric encryption involves an assumption that the communicating parties has prearranged shared secret key in order to encrypt the message.

Catalin et al. (2011) had proposed an approach for SMS security on mobile phone in order to improve the security of SMS based M-Learning. The approach had focused on improving the security by using symmetric encryption based on a secret key known by both parties. The solution implements a symmetric encryption technique based on AES. As mentioned in the paper AES is the fastest and efficient types of symmetric key cryptography techniques and also needs less computing resource, hence, it is good solution for mobile applications.

Another paper by Hassan et al. (2011) proposes AES encryption technique to exchange secure SMS. This paper describes a system for securing SMS messages during and after its transmission over mobile network. As mentioned in this paper, the system can send encrypted messages and allow users to encrypt/decrypt messages for personal usage. The system depends on secret key embedding, where the message’s secret key is distributed inside the cipher text after message encryption process. Secret key embedding is used for checking the correctness of a decryption key which is entered by the user. As mentioned in the paper, this scheme saves time and space as there is no need for a database to store the secret key related to each message. But this system has numbers of limitations; the first one is the receiver must remember the secret key of associated sender because, the secrete key is not stored on the mobile phone. The other limitation is, it increase the size of the message because of the embedded secret key on the encrypted message. As mentioned in the paper embedding and extracting secret key is not implemented and not tasted.

The paper by Singh et al. (2012) describes an approach to android or mobile application that allows the user to share textual information via SMS without being intercepted by any unauthorized user. A cryptography technique had proposed which avoids the leakage of clear messages being transferred from one end to the other, by encrypting it by using an already exchanged secret key. As described in the paper, the secret key exchange is done using the Diffie-Hellman key exchange algorithm, and AES-128 is used as the encryption/decryption algorithm. The system supports only android mobile phones.

Rohan et al. (2012) proposed and developed a framework for SMS on Android platform which allows the user to encrypt the messages before sent over the GSM network and decrypt it after receiving the message. The system uses Advanced Encryption Standards (AES) algorithm for encryption and decryption of the data. In order to use this application the user must login in to the system by entering the correct password. The application can be used to authenticate the sender of a message. Also, it is possible to detect, if the message has been corrupted or tampered during transmission. This application is developed on Android platform. Therefore, it is used by devices that operate on android platform only.

Raghavendra et al. (2011) had proposed a solution for secure messaging channel using IdentityBased symmetric (AES) cryptography. The advantage of this solution is that, it does not require a large storage on mobile terminal side, which is especially essential for user-to-user communication.

**2.3.2 SMS Security Solutions using Asymmetric Cryptography**

Shubat et al. (2009) proposed an application for sending end-to-end encrypted SMS messages using Identity-based asymmetric cryptographic. The main concern in public-key cryptography is the authenticity of the public key; this issue can be resolved by Identity-Based (ID- Based) cryptography where the public key of a user can be derived from public information that uniquely identifies the user. This paper presents an encryption mechanism based on the Identity-Based scheme using Elliptic Curves Cryptography to provide end-to-end security for SMS. The encryption overhead has been estimated and compared with RSA scheme. This study indicates that, the ID-based mechanism has advantages over the RSA mechanism in key distribution and scalability of increasing security level for mobile service. For example, the public key of a user can be simply his/her telephone number or email address, and hence implicitly known to all other users. But, encrypting the whole content using ECC is not suitable due to two main reasons. The first one ECC produces more large cipher text than the original text. The second one is it makes the system very slow.

Sameer et al. in 2011 also proposed public key cryptography for mobile phone application. The solution implements NTRU public key cryptography in non-server architecture. As described by Sameer et al. mobile phones will be able to achieve all cryptographic operations such as key generation, encryption /decryption and signing /verifying without relying on the third party’s server. The users will also gain the confidentiality, authentication, integrity and non-repudiation security services for their mobile phone communication. However, the problem of how the communicating parties authenticate each other appears. As we know, the main drawback of asymmetric cryptograph is the time and space needed to generate key pairs; hence, it is difficult to assign such task for devices that have limited resources. Besides, the keys in NTRU are not interchangeable. It needs NTRU Encrypt key pair for encryption or decryption processes and another NTRU Sign key pair for digital signature processes. As a result, this solution doubles the key required.

**2.3.3 SMS Security Solutions using Hybrid Encryption**

Anuar et al. (2008) proposed a framework for SMS by using both symmetric (AES) and asymmetric (RSA) cryptography. In this system the message classified as normal SMS (no encryption), internal SMS (symmetric encryption), and confidential SMS (asymmetric encryption). The paper proposed m–PKI to provide public key encryption to the mobile SMS. This approach allows the end-user to send private and classified message via SMS. In this solution the content of the SMS can be encrypted by using AES or by using RSA based on the classification.

Hybrid cryptographic scheme had been proposed by Al-bakri et al. (2010) to solve security limitation of SMS messaging. As mentioned in the paper, they had used the combination of asymmetric (NTRU) and symmetric (AES-Rijndael) cryptography algorithms to achieve more robust functionality. One of the limitations of this system is, it requires two pair of keys. In other word the keys in NTRU are not interchangeable. This means the user needs NTRU Encrypt key pair for encryption or decryption processes and another NTRU Sign key pair for digital signature processes. The other drawback of this application is the cipher text size produced after encryption is large. This system produces 1757 byte from 140 byte SMS. Thus, using this application is not advisable for SMS based applications.

**2.3.4 General Observations from the Study of Related Works**

In general symmetric cryptography solution for SMS security had some limitations; first it needs secure channel for key exchange or the security of the message is strongly depends on the secrecy of the key. One other limitation is the sender and the receiver must maintain large numbers of key approximately n(n−1)/2 keys where n is numbers of users. Also, this approach not supports some security requirements of SMS such as non-repudiation.

Some drawbacks of asymmetric cryptography based approach are: it needs more computational resource such as, memory and processer power than symmetric cryptography, also in asymmetric cryptography size of cipher text greater than the original message. Therefore, it is not advisable encrypt the whole SMS content using asymmetric key encryption.

In this final year project, an effective encryption based solution is proposed. In the proposed solution, symmetric (Blowfish) cryptography is to be used to encrypt/decrypt the SMS content and asymmetric (ECC) is to be used to transfer the secrete key which is used to secure the SMS content. To solve key transmission problem of symmetric key cryptography, in the proposed solution, the secret key is to be transmitted together with the encrypted SMS content in a secure manner. This solves the requirement of large key size. To do this, the proposed system generate random secrete key, encrypt it by using ECC and transmit to the receiver together with the SMS content at a time, thus, in this system, a single user needs only one private-public key pair. As observed in the study of related works, encrypting the whole content of SMS message using asymmetric cryptography is not advisable. To reduce this problem, the proposed system only encrypt the secrete key by using asymmetric (ECC) cryptography.