CHAPTER 2

LITERATURE REVIEW

This chapter combines several aspects of the preliminary study of this research which includes overview on cryptography algorithm followed by type of encryption in cryptography. Brief explanation on location-based algorithm which also known as geo-encryption method have been summarised in the literature review. It also summarises the related existing works on differences between symmetric and asymmetric encryption, comparison of Data Encryption Standard (DES) and Advanced Encryption Standard (AES) algorithms, location-based cryptography using symmetric, asymmetric and hybrid algorithm. The protocol used in location-based cryptography has been described.

2.1 Cryptography

Cryptography is a process of discreetly conveying secret information between one party and another (Kandola and Torrey 2013). It is associated with the process of converting ordinary plain text into unintelligible text and vice-versa. It is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. It is necessary when communicating through untrusted network especially on the open internet. Cryptography not only protects data from theft or alteration but can also be used for user authentication. In other words, cryptography is a secret of writing where it enables people to send or store sensitive information in the form of unreadable or non-understandable language.

In general, a plaintext is encrypted using a cryptography algorithm. The plaintext or original message becomes cipher text, and its original content is completely concealed. The cipher text can then be sent safely to the recipient. When the recipient wanted to reveal the message, they can do so by applying a decryption algorithm which will reveal the original plaintext. Only the recipient can apply the decryption algorithm because the recipient knows which key will be used to decrypt the message. Keys are used to personalize and secure a cryptography algorithm to only the sender and recipient (Kandola and Torrey 2013).

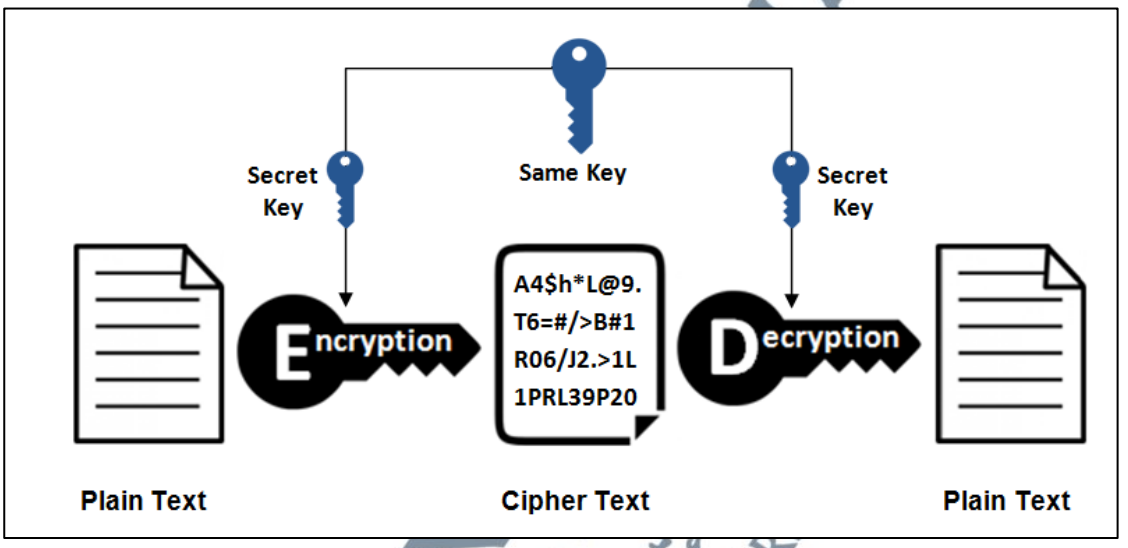
2.1.1 Encryption and Decryption

Encryption and decryption are part of cryptography processes. Encryption is the transformation of any kind of data into a form that is not understandable while decryption is the opposition of the encryption which converts encrypted data into understandable form (AJalab and Al-Nabhani 2010). The decrypted text of the original message or signal called as cipher. In order to decrypt the encryption, a key called decryption key is required for reverse operations. Without a correct decryption key, a message may not be decrypted as well as when a loss of the decryption key, meaning loss of decrypted message. Therefore, a decryption key must be secured and protected properly. The more complicated the encryption algorithm, the more difficult it becomes to break the cipher.

Since the availability of earlier computer communication, there are different techniques of modern cryptography identified based on the key used during the encryption and decryption process, which are symmetric cryptography and asymmetric cryptography.

2.1.2 Symmetric Cryptography

Symmetric cryptography also known as secret key only used a single key to encrypt and decrypt the data. The message can only be decrypted if the authorized person knows the key. Example of symmetric key algorithm are DES, Triple DES, AES and Blowfish (Surya and Diviya, 2014). Figure 2.1 shows the process of symmetric cryptography.



Source: (SSL2Buy 2017)

Figure 2.1: Symmetric Encryption

i) Data Encryption Standard (DES)

DES was the first encryption standard introduced by NIST (National Institute of Standards and Technology). It is developed by IBM in the early 1970s and designed by Lucifer Cipher. It is an implementation of Feistel Cipher. DES is a block cipher use the same secret key for encryption and decryption process. It encrypts the data in a block of 64 bits and produce 64 bits of cipher text. The key length is 56 bits to customize the transformation so that decryption can be performed only by those who know the key to encrypt the message (Mitali et al., 2014).

ii) Triple DES

Triple DES stands for Triple Data Encryption Algorithm block cipher (DES) which replaced the Data Encryption Standard. Originally, DES cipher key size was 56 bits which are sufficient enough when the algorithm was designed. But, increasing in computational power make it vulnerable to attack such as brute-force (Surya and Diviya, 2014). Triple DES provides a better protection for DES attack as it takes three times of DES key which is 192 bits key. In Triple DES, the data will be encrypted with the first key, decrypted with the second key and decrypted with the third key. It runs three times slowly compared to DES, but Triple-DES is more secured.

iii) Advanced Encryption Standard (AES)

AES is a symmetric block cipher was published by NIST (National Institute of Standards and Technology) in December 2001. It is developed by two Belgium cryptographers, Vincent Rijments and Joan Daemen (Mitali et al., 2014). AES consists of three block cipher which is AES-128, AES-192 and AES-256. It encrypts and decrypt a data block of 128 bits and it is a non-Feistel cipher. The number of rounds can be 10, 12 and 14 depends on the key size. Each processing round in AES involves substitute bytes, shift row, mix column and add round key.

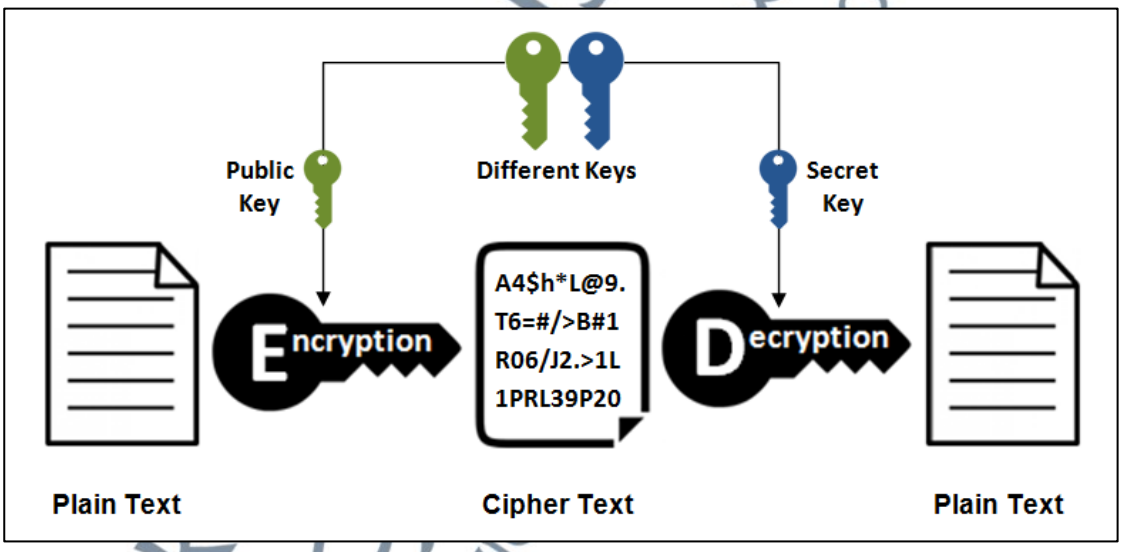
iv) Blowfish

Blowfish is a symmetric block cipher effectively used for encryption and to secure the data. It was introduced in 1993 by Bruce Scheneir, one of the world's leading cryptologists (Mitali et al., 2014). Blowfish algorithm was Feistel Network and encrypt 64-bit block cipher with a variable length key. The algorithm contains two parts which are subkey generation and data encryption. Subkey generation converts the key up to

448 bits and totalling all subkey to 4168 bits. Data encryption involves the iteration process up to 16 times. Each round consists of key dependent permutation and data dependent permutation.

2.1.3 Asymmetric Cryptography

Asymmetric cryptography also referred as public key cryptography needs two keys which is a public key and private key for encryption and decryption. Public keys can be accessed by anyone, but private key only known by the owner (Tripathi and Agrawal, 2014). Example of asymmetric algorithm is Rivest-Shamir-Adleman (RSA) and Diffie-Helman. Figure 2.2 illustrates how the asymmetric cryptography process and Table 2.1 shows the differences between symmetric and asymmetric cryptography.



Source: (SSL2Buy 2017)

Figure 2.2: Asymmetric Encryption

Table 2.1: Differences between Symmetric and Asymmetric Encryption

Features	Symmetric Encryption	Asymmetric Encryption
Key	Single key.	A pair of public key and private key.
Key sharing	Need to be shared among the people who need to receive the message.	Only public key needs to be shared.

Revolution	Old fashion technique.	Relatively new technique.
Complexity	Easier and best-known technique.	Harder and more complex.
Time required	Less time.	More time.
Security strength	Lack of security as key is being exchange within all parties involved.	More powerful.

v) Rivest-Shamir-Adleman (RSA)

RSA is the most common algorithm used in public key encryption. It can be used in both data encryption and digital signatures. It is first published by Ron Rivest, Adi Shamir and Leonard Adleman in 1977. RSA involves three steps which are key generation, encryption, and decryption (Sahajan and Sachdeva, 2013). The main operation of the RSA is considered as factoring. Generation of prime number gives extra strength and security of the algorithm.

vi) Diffie-Helman

Diffie-Helman used discrete logarithms in finite fields. The communication between two parties allow to exchange the secret key in insecure medium without any secret. It is widely used in the key exchange (Tripathi, Agrawal 2014).

2.1.4 Comparison between DES and AES Method

Both of DES and AES are from symmetric cryptography algorithm which using a same single secret key for encryption and decryption of data. The comparison on characteristics of both algorithms is illustrated as in Table 2.2.

Table 2.2: Comparison of DES and AES

Parameter	DES	AES
Key Length	Very short, 56 bits	128, 192, 256 bits
Block Size	64 bits	128, 192, 256 bits
Cipher text	Symmetric block cipher	Symmetric block cipher

Max data size per block size	32 GB	256 GB
Possible key combinations	2^{56}	2^{128}, 2^{192}, 2^{256}
Network Structure	Feistel Network structure	Permutation Substitution Network structure
Security	Vulnerable to Brute Force Attack	More secure due to long key length

Source: (Kolapwar 2015)

2.2 Transformations in AES

AES is a symmetric block cipher where a single key is used for both encryption and decryption process. Each input and output for the AES algorithm consists of sequences of 128 bits. The key used in this algorithm consists of 128, 192, or 256 bits. AES operates on 8-bit bytes which has four main transformations that are recursively replicated. These transformations are sub bytes, shift rows, mix columns and add round key. These transformations are replicated differently for each encryption phase. The encryption phase of AES can be broken into three phases which are the initial round, the main rounds and the final round. The initial round has add round key, the main rounds have sub bytes, shift rows, mix columns and add round key, while final round has only sub bytes, shift rows and add round key only. Figure 2.3 shows the overall structure of encryption and decryption phase in AES.

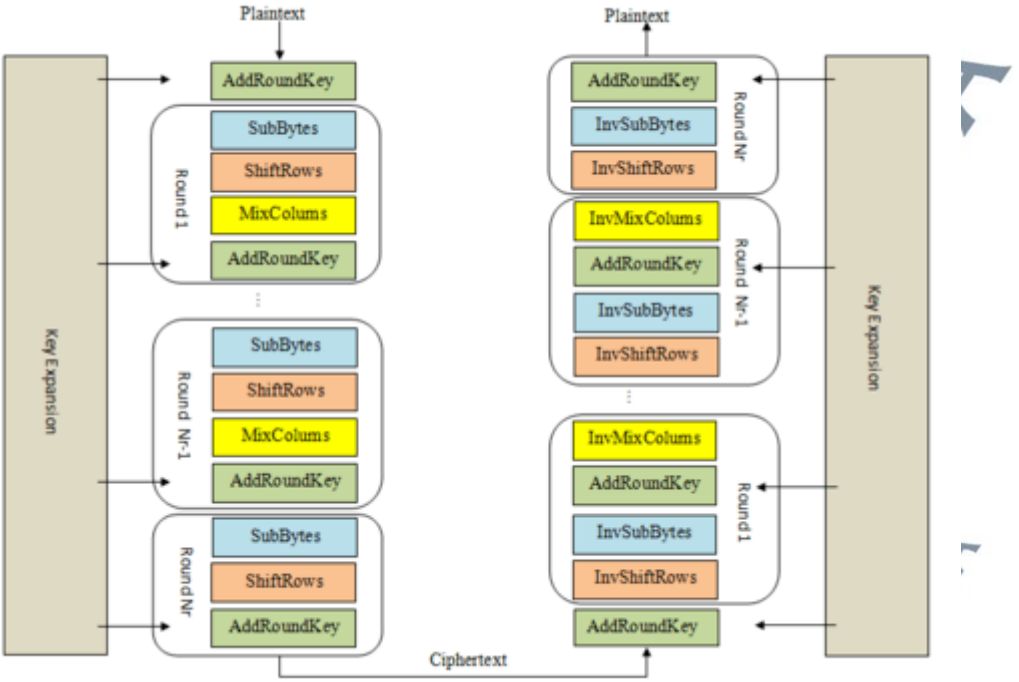


Figure 2.3: Overall structure of transformation phase in AES

The next subsections will discuss the detailed of the round transformation that recur repeatedly which the number of rounds is based on the phases and the key length.

2.2.1 Add Round Key

The Add Round Key operation is the only phase in AES encryption that directly operates on the AES round key. This transformation involves the conversion of input to the round by XORing it with the round key into the state matrix. The XORed key then denoted as the State-Round Key.

2.2.2 Sub Bytes

The Sub Bytes phase is involving the process of splitting the input into bytes and passing each of it through a Substitution Box or called as S-Box. In cryptography, S-Box considered as essential constituent of symmetric algorithm. Block ciphers use S-Box to make relationship between the cipher text and the plain text difficult to comprehend. Sub Bytes transformation also considered as non-linear byte substitution

which it has an independent operation on each of the State byte using the S-Box. Figure 2.4 shows the S-Box lookup table.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
0a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
0b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
0c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
0d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
0e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
0f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Figure 2.4: S-Box lookup table

To read S-Box table, the byte input is brake into two 4-bit halves. The first half determines the row, and second half determines the column. For example, the S-Box transformation of 59 or s_{11} = {59} can be found in the cell at the intersection of the row labelled 50 and the column labelled 09. Therefore, s_{11} would have value of {cb}.

2.2.3 Shift Rows

In the Shift Rows phase of AES, every row of state will be shifted. The rows in this stage refer to the standard representation of the internal state in AES which is 4 x 4 matrix where each cell contains a byte. Bytes of the internal state are placed in the matrix across rows from left to right and down columns. The intention of this phase is to achieve a combination of bytes that are positioned in various location of plaintext message block.

In this layer, the diffusion property is increase by the distribution of the byte's positions to other positions while keeping the element values the same. Each of rows is shifted to the left by a set amount. The top row is not shifted at all while the next row is shifted by one and so on. Figure 2.5 illustrates the shift rows.

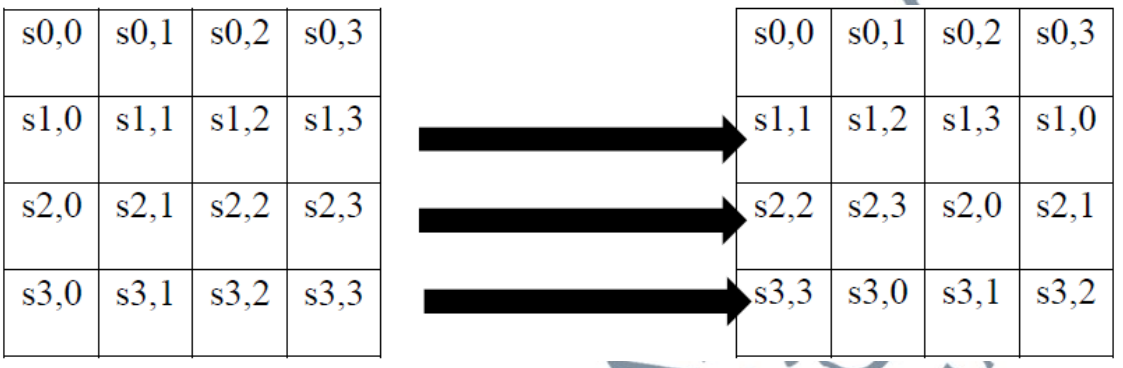


Figure 2.5: Shift rows operation

2.2.4 Mix Columns

As in the Shift Rows phase of AES, the Mix Columns phase provides diffusion by mixing the input around but in Mix Columns, it performs its operations through splitting the matrix by columns instead of rows. Figure 2.6 shows the Mix Column operation.

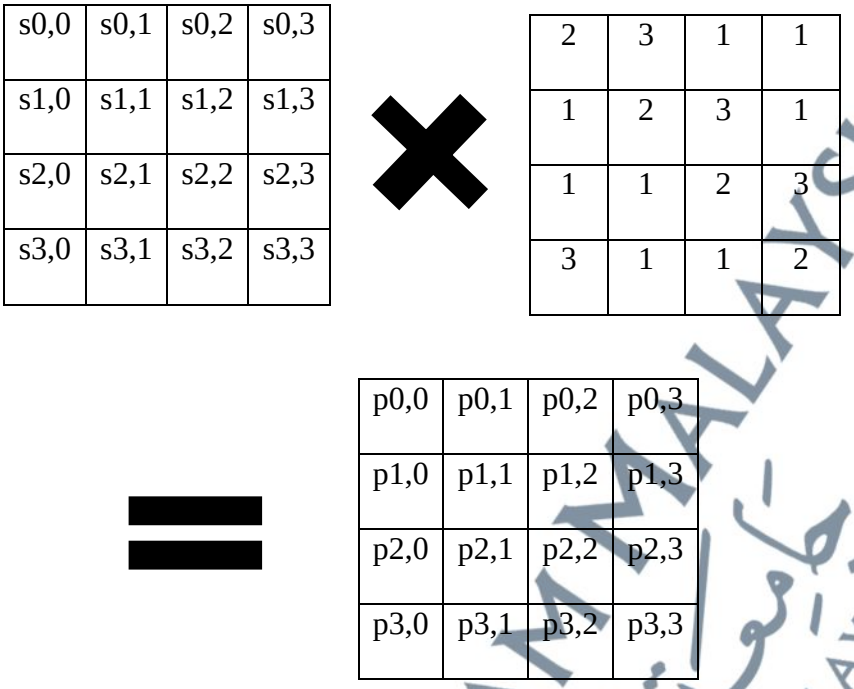


Figure 2.6: Mix columns transformation

Mix Columns performs matrix multiplication not as the standard matrix multiplication, but it follows as per the Galois Field 2^8. It is important to note that this multiplication has the property of operating independently over each of the columns of the initial matrix. For instance, the first column when multiplied by the matrix, produces the first column of the resultant matrix.

2.3 Location-based Cryptography

A.S. Amin (2010) defines location-based cryptography or geo-encryption as a method of encryption where the information, ciphertext were encrypted and can be decrypted at only a specific location. This GPS-based encryption was an innovative technique used to encode location information into the encryption keys to provide security (Al-Fuqaha, et al., 2007). Geo-Encryption which is based on cryptographic, also adding a new security layer on the available encryption protocol structure using the

recipients location information (Vandana et al. 2015). It allows data to be encrypted for a specific or broad geographic area to fully protect against any attempts to bypass the location feature. The original information will not be revealed if there are any attempt to decrypt the data at other location as it failed the decryption process.

This location-based cryptography can be used to ensure the encrypted data cannot be decrypted outside a particular facility. Some examples by (Scott and Denning 2003b), the headquarters of a government agency, corporation, an individual office or someone home. It may be used to confine access to a broad geographic region as an alternative. Time and space constrain also may be placed on the decryption location.

Regarding Scott & Denning (2003), the principle of geo-encryption is when a set of location and time specifications bind to the encrypted cyphertext file and build device which would decrypt the file within the specific location and time only. However, there are several potential problems which is, the possibility of the resultant file revealing the physical location of the intended recipient. If the device is vulnerable to tampering, it may be possible to be modified and lead to bypass the location completely. The modified device would decrypt all received data without acquiring its location and verify if it was correct. Another possibility, one might consider using the location itself as cryptographic key of another strong encryption algorithm like AES. There may be enough information to enable rapid brute force attack even if an adversary does not know precise location. This approach could be strengthened by applying an obfuscation function to the location value before using it as a key, but the function would have to be kept secret in order to prevent the adversary from do the same thing.

2.3.1 Location-based Cryptographic Methods

Cryptography is a secret of writing where it enables people to send or store sensitive information in the form of unreadable or non-understandable language (Pitchay et al. 2015) through either symmetric cryptography or asymmetric cryptography. In location-based cryptography, it also builds by established cryptographic algorithm including both symmetric as well as asymmetric algorithm. However, there are also some of the existing works that used both of these cryptographic techniques at the same time known as hybrid algorithm (Scott and Denning 2003b) for the implementation of location-based cryptography.

i) Symmetric Algorithm (private-key cryptography)

Numbers of very fast symmetric algorithms are used widely in location-based cryptography including Data Encryption Standard (DES), Triple-DES and Advanced Encryption Standard (AES) (Scott and Denning 2003b). Symmetric algorithm has better performance compared to asymmetric algorithms based on its speed (Qiu et al. 2007). Symmetric algorithm has about 1000 times faster than the asymmetric algorithm because it has the mutual key for encryption and decryption. Table 2.3 summarizes the existing works on location-based cryptography that used symmetric algorithm and its techniques.

Table 2.3: Location-based Cryptography using Symmetric Algorithm.

No	Existing Works	Year	Symmetric Technique	Strength	Deficiency
1.	(Karimi 2011)	2011	DES	Effective and practical for data transmission.	Decryption successful rate decreased.
2.	(Karimi and Kalantari 2011)	2011	DES	Effective transmission between mobile client.	Decryption successful rate decreased.

3.	(Sasi et al. 2014)	2014	DES	Message access optimized only to specific area.	MAC ID changed if device rooted, upgrade version or factory reset.
4.	(Pranjala G Kolapwar 2015)	2015	Modified AES	Low algorithm complexity.	Missing mix column transformation.
5.	(Himanshu Pant et al. 2016)	2016	AES	Straightforward and less effort for administrative overhead.	-
6.	(Auti, Landage, and Chavan 2016)	2016	DES	Privilege setting available.	High budget to afford anti-spoof device.
7.	(Dalvi et al. 2017)	2017	AES	Available for multimedia files.	-
8.	(Sanjay et al. 2017)	2017	AES	Good defence against cryptanalytic attack.	-
9.	(Chaudhari 2017)	2017	AES	Best suit for mobile application.	Need velocity as additional parameter.

ii) Asymmetric Algorithm (public-key cryptography)

Asymmetric algorithm used public key and private key to encrypt and decrypt the data. In location-based cryptography, Rivest-Shamir-Adleman Algorithm (RSA) was found as the strongest public key available encryption technique (Khan 2013) and the most used technique by the researchers (Scott and Denning 2003a). The generation and difficulty of factoring large prime number gives extra strength as well as extra security towards the algorithm (Scott and Denning 2003b). However, the asymmetric algorithm is vulnerable to chosen-plaintext attack [5]. Table 2.4 shows existing works on location-based cryptography that used asymmetric algorithm and its technique.

Table 2.4: Location-based Cryptography using Asymmetric Algorithm.

No	Existing works	Year	Asymmetric Technique	Strength	Deficiency
1.	(Khan 2013)	2013	RSA	Strong protection against location spoofing.	Too small grid space cause wrong geo-lock.
2.	(Gupta et al. 2014)	2014	RSA	Assure secure data access.	Exposed to mathematical attack and brute force attack.

iii) Hybrid Algorithm (public and private key cryptography)

In terms of computational and implementation, it was very fast when using symmetric algorithm but slower using asymmetric algorithm due to difficulty in its computational. However, asymmetric algorithm offers very high security (Deshpande et al. 2015). Therefore, a combination of symmetric and asymmetric encryption is used called hybrid algorithm. The public key algorithm is used to secure and distribute session keys while the symmetric encryption is used to encrypt the information. Table 2.5 summarises the existing related works on location-based cryptography that used hybrid algorithm and its techniques.

Table 2.5: Location-based Cryptography using Hybrid Algorithm.

No	Existing works	Year	Asymmetric Technique	Symmetric Technique	Strength	Deficiency
1.	(Scott and Denning 2003b)	2003	RSA	AES	Support location-based data encryption.	Hard to meet same mapping function output.
2.	(Al-Fuqaha et al.2007)	2007	Not stated	Not stated	Support mobility	Low decryption ratio.
3.	(Qiu et al. 2007)	2007	RSA	AES	Protected against location spoofing	Too small grid space cause wrong geo-lock .
4.	(Al-Ibrahim et al. 2007)	2007	Not stated	Not stated	Support moving decryption zone	Increasing message queue cause low decryption ratio.
5.	(Liao and Chao 2008)	2008	Not stated	Not stated	Skip mapping function	Limited toleration distance.
6.	(Hamad and Elkourd 2010)	2008	Not stated	Not stated	Support dynamic toleration distance	Decryption failed if movement too fast.

No	Existing works	Year	Asymmetric Technique	Symmetric Technique	Strength	Deficiency
7.	(A.S. Amin 2010)	2010	Not stated	Not stated	Support mobility	Low rate of decryption.
8.	(Abolghasemi et al. 2013)	2013	RSA	AES	Limit data access to specific room	High cost of anti-spoof and GPS device.
9.	(Kolapwar and Ambulgekar 2014)	2014	Not stated	AES	Low energy, high packet delivery ratio	Possible attack spoofing and replay network.
10.	(Vandana et al. 2015)	2015	Not stated	AES	Prevent unauthorized access in cloud	Challenging data access control.
11.	(Deshpande et al. 2015)	2015	RSA	AES	Customer can access account from anywhere	Person needs to stay stable during transaction.
12.	(Kumar and Murthy 2015)	2015	RSA	AES	Accurate user location	Challenging access control.
13.	(Anju and Joseph 2015)	2015	Not stated	Not stated	Very accurate results	Expensive and difficult computational technique.

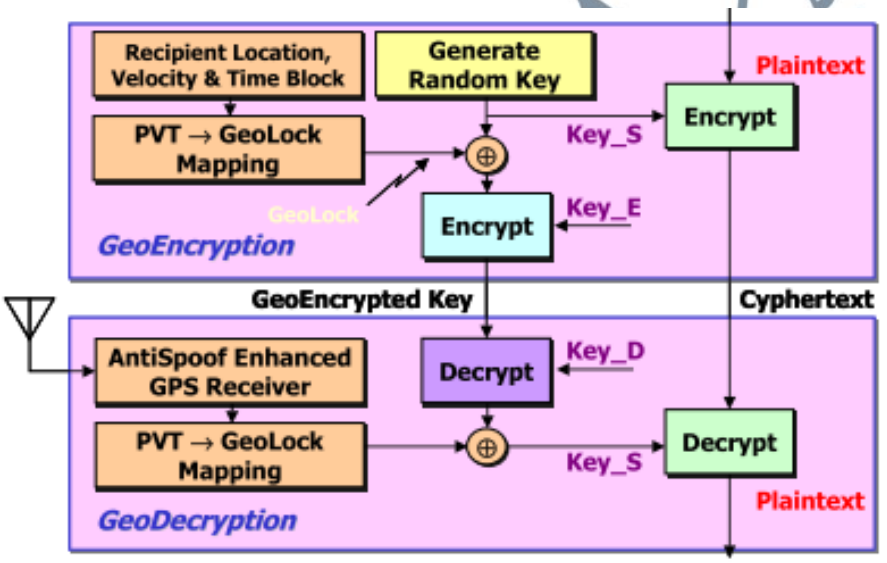
2.3.2 Location-based Cryptographic Protocols

Location-based cryptographic requires location information as parameters such as latitude and longitude coordinates, time and velocity to generate an additional key for encryption and decryption process. Different protocols use different parameters for its implementation. Some of the protocols in location-based cryptographic techniques are Geo-encryption Algorithm (Scott and Denning 2003a), Timed, Efficient Stream Loss-tolerant Authentication (TESLA) (Qiu et al. 2007), Location Dependent Encryption Algorithm (LDEA) (Liao and Chao 2008), Dynamic Toleration Distance (DTD) (Hamad and Elkourd 2010) and Improved Geo-encryption Protocol (IGEP) (A.S. Amin 2010).

i) Geo-Encryption Algorithm

The Geo-encryption Algorithm was firstly invented and developed by Logan Scott and Dorothy E Denning in 2003 based on the traditional encryption system and communication protocol. The data is encrypted according to the expected

position, velocity and time (PVT) of the receiver. Mapping function is used to convert the PVT into GeoLock key. The GeoLock key computes a bitwise exclusive-OR with generated random key to produce another key called GeoLock session key. Then, it is transmitted to the receiver by using asymmetric encryption. Receiver will used anti-spoof GPS device to acquire the PVT data and produce the final session key through the same process. The final session key is then used to decrypt the ciphertext. Figure 2.7 illustrates how geo-encryption algorithm take place.



Source: Scott and Denning (2003b)

Figure 2.7: Geo-encryption Algorithm

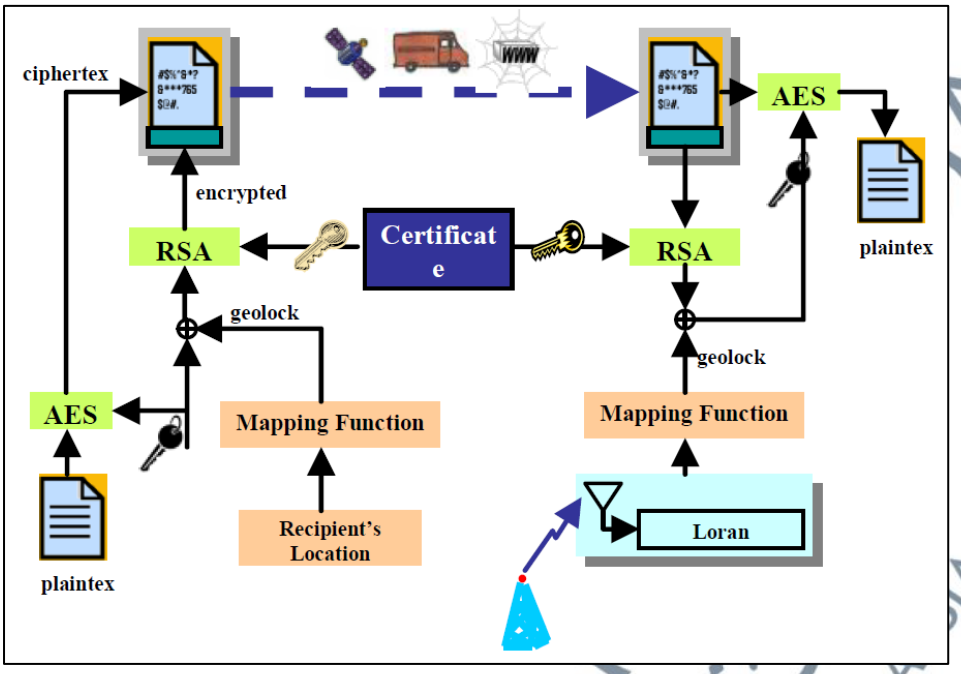
However, the PVT-to-GeoLock mapping function is the primary mechanism to ensure that the data can be decrypted successfully. It is troublesome for sender and receiver to own the same mapping function before the data transmission if they communicate occasionally (Scott and Denning 2003b, 2003a). The following Table 2.6 summarizes the existing works on encryption using geo-encryption algorithm protocol.

Table 2.6: Encryption using Geo-encryption Algorithm Protocol

No	Existing Works	Year	Protocol	Parameter	Cryptographic Technique
1.	(Scott and Denning 2003a)	2003	Geo-encryption	Latitude, longitude, time & velocity	RSA & AES
2.	(Al-Fuqaha et al. 2007)	2007	Geo-encryption	Velocity, direction, speed maneuverability & breadth maneuverability	Not stated
3.	(Al-Ibrahto et al. 2007)	2007	Geo-encryption	Velocity, direction, speed maneuverability & breadth maneuverability	Not stated
4.	(Qiu et al. 2007)	2007	Geo-encryption	Time, cycle difference, signal noise & shape of cycle	RSA & AES
5.	(Abolghasemi et al. 2013)	2013	Geo-encryption	Latitude, longitude time & velocity	RSA & AES
6.	(Gupta et al. 2014)	2014	Geo-encryption	Latitude, longitude & time	RSA
7.	(Vandana et al. 2015)	2015	Geo-encryption	Longitude, latitude, time, velocity & co-ordinate system	Asymmetric & AES
8.	(Kumar and Murthy 2015)	2015	Geo-encryption	Latitude, longitude & time	RSA & AES
9.	(Sanjay S et al. 2017)	2017	Geo-encryption	Latitude, longitude & time	AES
10.	(Dalvi et al. 2017)	2017	Geo-encryption	Latitude, longitude & time	AES

ii) TESLA Protocol in Geo-encryption using Loran

Loran is a terrestrial and low frequency pulsed navigation system that has good repeatable accuracy position (Qiu, et al., 2007). It has high power of a low frequency signal where it is hard to spoof and able to reach indoor environment that may not be reachable by GPS. Figure 2.8 illustrates where the Loran being implemented into geo-encryption mapping function.



Source: (Qiu et al. 2007).

Figure 2.8: Geo-encryption using Loran Overview

In order to implement Loran into geo-encryption, a signal authentication protocol named Timed Efficient Stream Loss-tolerant Authentication (TESLA) is being used (Qiu et al. 2007). TESLA protocol provides authentication and improve system integrity. Loran transmitter and receiver synchronize loosely with sender's local time. Hash function used to compute the TESLA one-way chain key values. Each segment of chain consists of a message. Loran transmitter then uses the last one-way chain values to generate message authentication code (MAC) key for previous MAC. For each segment of the message, it consists of MAC of current message and MAC key of previous message. Thus, the messages, MAC and previous message MAC's key are transmitted in a segment. When the segment received by the receiver, MAC key is verified by hashing the current segment key and compare with previous segment key.

The performance of geo-encryption in this work, (Qiu et al. 2007) is depends on the TESLA authentication and the receiver accuracy. Once TESLA is failed, user cannot proceed to the next step which is to compute the geo-lock. TESLA performance is

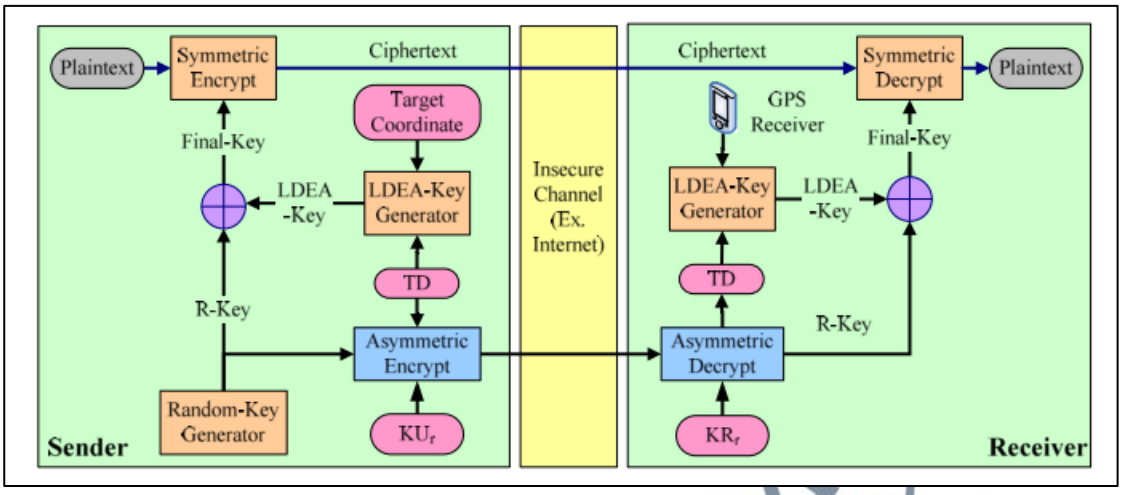
depended on the signal noise ratio (SNR) of Loran signal and authentication bandwidth. Noise variance used to determine the probability error. If the SNR increase, the probability of error is decreases and the message loss rate decrease. For the receiver accuracy, geo-encryption performance depends on the grid size used by user. If the size chosen is too small, receiver will receive a different grid of the sender location as a result, the random key would not be recovered because wrong geo-lock is computed. Table 2.7 shows work that using TESLA protocol in geo-encryption using Loran, its cryptographic technique and parameter used.

Table 2.7: TESLA Protocol in Geo-encryption using Loran

No	Existing Works	Year	Protocol	Parameter	Cryptographic Technique
1.	(Qiu et al. 2007)	2007	Geo-encryption & TESLA in Loran	Time, cycle difference, signal noise & shape of cycle	RSA & AES

iii) Location Dependent Data Encryption Algorithm (LDEA)

LDEA was proposed by (Liao and Chao 2008). They propose a static location-dependent data encryption for mobile information system and skip the mapping function in previous Geo-encryption algorithm protocol (Sasi et al. 2014). The approach is based on a reverse hashing principle. LDEA is mainly to include the latitude and longitude coordinates in the data encryption as well as to restrict the data decryption location. Toleration distance (TD) protocol is designed to overcome the inaccuracy of static location and inconsistent problem of GPS device receiver. When the target coordinate and TD is given by the sender, an LDEA-key is generated. If the acquired coordinate is matched with the target coordinate within the range of TD, the cipher text decrypted into original plaintext. Figure 2.5 shows how LDEA process took place.



Source: (Liao and Chao 2008)

Figure 2.9: The Process of LDEA

However, LDEA have lack of coordinate accuracy due to inconsistent problem of GPS receiver. Thus, a Toleration Distance (TD) has been designed and implemented in LDEA to overcome the problems (Deshpande, et al., 2015; Prasanna & Reddy, 2014; Liao & Chao, 2008). LDEA also lack in its security as it was static encryption which depends only on mobile nodes and static TD. If being tampered by physical attack such as device spoofing, the device is vulnerable to an adversary party to modify and bypass the location check. Table 2.8 shows LDEA protocol in existing location-based cryptography works.

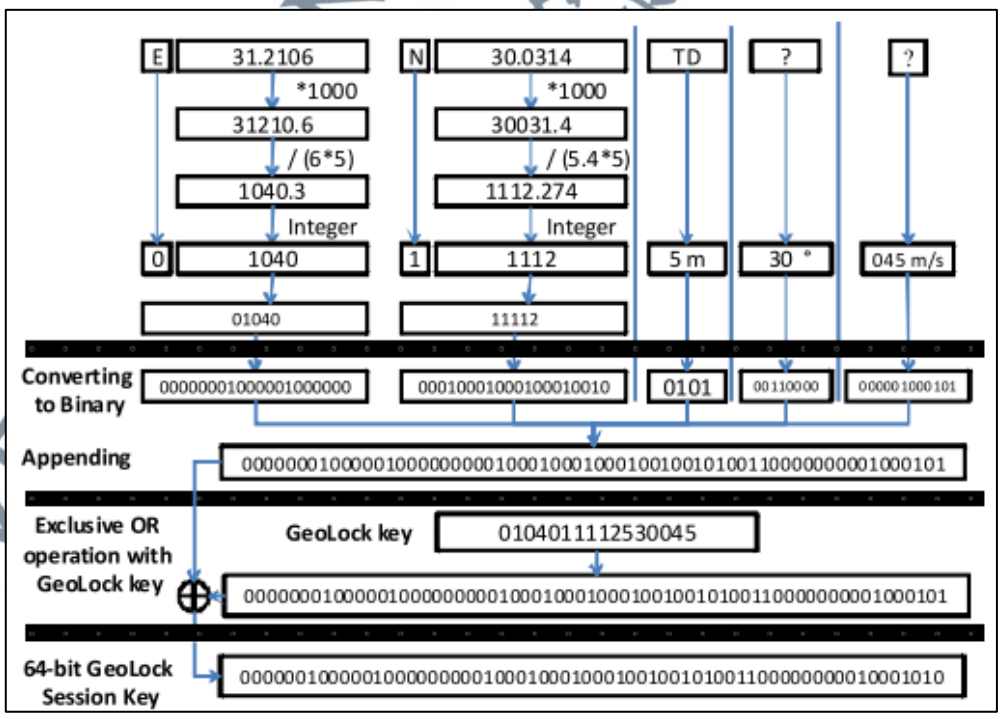
Table 2.8: Location-based Cryptography using LDEA Protocol.

No	Existing Works	Year	Protocol	Parameter	Cryptographic Technique
1.	(Liao and Chao 2008)	2008	LDEA	Latitude & longitude	Not stated
2.	(Reddy et al. 2010)	2010	LDEA	Latitude & longitude	Permutation cipher
3.	(Deshpande et al. 2015)	2015	LDEA & DTD	Latitude & longitude	RSA & AES

iv) Improved Geo-Encryption Protocol (IGEP)

The key for data encryption in IGEP is using latitude and longitude coordinate by designing a TD (A.S. Amin 2010). As there were inconsistent on how many satellite signals received by the GPS receiver, it is impractical to use the GPS coordinate as a key for data encryption without the TD. However, some problem might encounter this approach as it provides vital information to someone who want to spoof the device. Besides, the resultant file can reveal the physical location of intended recipient.

IGEP proposed model was based on the geo-encryption technique which for both sender and receiver are mobile. Receiver mobile node needs to deliver their mobile node's movement information and called movement parameter in order to estimate their expected location at any point in time to the potential sender mobile node through a sequence of message exchanges. Figure 2.10 shows how the IGEP model being proposed.



Source: A.S. Amin (2010)

Figure 2.10: IGEP Proposed Model with Tolerance Distance

Simulation has been done to analyse the different between an improved protocol model which is IGEP and the existing geo-encryption protocol model (GEP) in term of the decryption ration and the protocol overhead. Both been measured using different network size and some fixed Tolerance Distances. The simulation results are as shown in Table 2.9.

Table 2.9: Decryption Ratio and Protocol Overhead

Protocol model	IGEP algorithm with Tolerance Distance 2.5			GEP algorithm with Tolerance Distance 10		
Network size (no. of sender and receiver)	10	20	30	10	20	30
Decryption ratio	97%	95.35%	93.6%	95.8%	94.71%	93%
Protocol overhead	5.81%	7.23%	9.38%	6.57%	8.44%	9.41%
Decryption ratio level	Higher			Lower		
Protocol Overhead	Lower			Higher		

Source:A.S. Amin (2010)

It has been shown that IGEP having higher decryption ratio rather than GEP. Means, it has been proved that IGEP algorithm was better than GEP. However, it looks biased when the comparison between IGEP and GEP applied a difference value of Tolerance Distance.

v) Dynamic Toleration Distance in Data Encryption

DTD was proposed by (Hamad and Elkourd 2010) to overcome the inaccuracy and inconsistent problem of GPS receiver and to increase its practicality. They propose a protocol which use dynamic location of mobile node and turn into a very strong towards attack. The mobile receiver registers a set of coordinates with velocity during movement and estimate the next position to apply on the secret key with DTD (Hamad and Elkourd 2010). These parameters with type of movement makes this protocol more secure than the static encryption which depends only on a position of mobile nodes and static toleration distance (TD). The following Table 2.10 summarizes the existing works.

Table 2.10 Location-based Cryptography using DTD Protocol

No	Existing works	Year	Protocol	Parameter	Cryptographic technique
1.	(Hamad and Elkourd 2010)	2008	DTD	Latitude & longitude	Not stated
2.	(Karimi 2011)	2011	DTD & MAC	Longitude, latitude, time & velocity	DES
3.	(Karimi and Kalantari 2011b)	2011	DTD & MAC	Longitude, latitude, time & velocity	DES
4.	(Kolapwar and Ambulgekar 2014)	2014	DTD	Longitude, latitude, time & velocity	Asymmetric & AES
5.	(Pranjala G Kolapwar 2015)	2015	DTD	Longitude, latitude, time & velocity	Modified AES
6.	(Deshpande et al. 2015)	2015	DTD & LDEA	Latitude & longitude	RSA & AES
7.	(Himanshu Pant et al. 2016)	2016	DTD	Latitude & longitude	AES

Table 2.11 shows the summary on some of the existing related works for the location-based encryption. This summary highlights the techniques, protocols and parameter used of each works. The comparison study motivated this research to work more on enhancing the existing works. AES algorithm has been used in most of the related existing works to perform the data encryption compared to another algorithm technique. There are also other algorithms that being used for data encryption such as DES algorithm, but (Kolapwar 2015) found that AES algorithm is the best contemporary algorithm when it compared to DES algorithm. Table 2.9 shows the comparison between AES and DES. However, AES algorithm only widely use to encrypt the plain text into cipher text. To generate the encryption key, which is based on location information, most works use the geo-mapping technique.

2.3.3 Comparison on Existing Related Works

The implementation of location-based cryptography consist of varies cryptography techniques and different protocol. Table 2.11 summarize the comparison of existing works that used location-based cryptography in their works.

Table 2.11: Summary on Existing Related Works on Location Based Cryptography

No.	Title/Author	Feature	Cryptography Technique	Protocol	Parameter
1.	A Location Based Encryption Techniques and Some of Its Application (Scott and Denning 2003b)	• Use PVT (position, velocity, time) to geo-lock mapping function.	AES & RSA	Geo-encryption algorithm	Latitude, longitude, time & velocity
2.	Geo-encryption protocol for mobile networks (Al-Fuqaha, Al-Ibrahim, and Rayes 2007)	• Proposed mobility parameter • Add significant layer of network transmission	Not stated	Geo-encryption algorithm	Velocity, direction, speed maneuverability & breadth maneuverability
3.	Geo-encryption Using Loran (Qiu et al, 2007)	• Propose a verification using Loran signal which is a low frequency pulsed navigation system. • A signal authentication called TESLA (Time Efficient Stream Loss-Tolerant Authentication) is proposed and implemented on Loran for authentication.	AES & RSA	Geo-encryption algorithms	Velocity, direction, speed maneuverability & breadth maneuverability
4.	Mobility support for Geo-Encryption (Al-Ibrahto, et al., 2007)	• Propose a protocol to allow mobile nodes exchanges movement parameter. • Contains a receiver node's estimated location.	Not stated	Geo-encryption algorithm	Time, cycle difference, signal noise & shape of cycle

No.	Title/Author	Feature	Cryptography Technique	Protocol	Parameter
5.	A New Data Encryption Algorithm Based on the Location of Mobile Users (Liao and Chao 2008)	• Proposed location dependent approach, Location Dependent Data Encryption Algorithm (LDEA). • Determine longitude and latitude. • Add toleration distance (TD).	• Symmetric algorithm for the plaintext encryption • Asymmetric algorithm for key encryption.	LDEA	Latitude & longitude
6.	Data Encryption Using the Dynamic Location and Speed of Mobile Node (Hamad and Elkourd 2010)	• Propose a new method of key security where the receiver mobile node (MN) registers some coordinate and speed during the travel.	• Symmetric encryption for plaintext, AES. • Asymmetric encryption for keys.	DTD	Latitude & longitude
7.	Improved Geo-Encryption Protocol for Mobile Networks (A.S. Amin 2010)	• Proposed Improve Geo-Encryption Protocol, IGEP • Latitude and longitude use as key for encryption. • Design Toleration Distance (TD)	• Not stated	IGEP	Latitude & longitude
8.	A Modified Location-Dependent Image Encryption for Mobile Information System (Reddy, et al., 2010)	• Cipher developed by permutation and rotation of Location Dependent Encryption Algorithm (LDEA) key. • Modify the cipher by introducing the concept of key dependent circular rotation.	• Permutation cipher	LDEA	Latitude & longitude
9.	Enhancing Security of Confidentiality on Mobile Device by Location-based Data Encryption (Karimi and Kalantari 2011b)	• Modify Geo-encryption technique which receiver can decrypt the message if the receiver at the specific location and limited time.	• MD5 hash • DES algorithm	DTD	Latitude, longitude, time & velocity
10.	Geo Location Based RSA Encryption Technique (Khan 2013)	• Latitude and longitude of source and destination as public and private keys	• RSA algorithm	Geo-encryption	Latitude & longitude

No.	Title/Author	Feature	Cryptography Technique	Protocol	Parameter
11.	Using Location Based Encryption to Improve the Security of Data Access in Cloud Computing (Abolghasemi, et al., 2013)	• Use user's location and geographical position to add security layer to existing security measure. • Use new generation of Anti-Spoof GPS • Make information inside the cloud available only within the particular room that has been defined in the user's location.	• AES algorithm for plaintext • RSA for geo-key	Geo-encryption	Latitude, longitude, time & velocity
12.	A Novel Security Approach using Location based RSA Encryption (Gupta, et al., 2014)	• Proposed Cloud Storage Methodology and Data Security in cloud by the Geo Location based RSA encryption.	• RSA algorithm	Geo-encryption	Latitude, longitude & time
13.	A Generalized Study on Encryption Techniques for Location Based Services (Prasanna & Reddy, 2014)	• Use Location Dependent Encryption Algorithm (LDEA) • Use Toleration Distance (TD) • Proposed Mobile User Location-specific Encryption (MULE)	• Exclusive-OR operation • Hash function	LDEA	Latitude & longitude
14.	Location Based Encryption Using Message Authentication Code in Mobile Networks (Sasi, et al., 2014)	• Propose a secure communication using a location dependent encryption technique. • Use of the unique ID of the device and MAC concept	• DES algorithm	Geo-encryption	Latitude, longitude & time
15.	Use of Advanced Encryption Standard to Enhance the Performance of Geo Protocol in Location Based Network (Kolapwar and Ambulgekar 2014)	• Modified AES algorithm with the Geo-encryption with Dynamic Tolerance Distance (AES-GEDTD) • Enhanced packet delivery ratio and data delay.	• AES algorithm	DTD	Latitude, longitude, time & velocity
16.	Location Based Encryption-Decryption Approach for Data Security (Manoj, et al., 2014)	• Adding location base service with the encryption process.	• DES and AES algorithm	LDEA	Latitude & longitude

No.	Title/Author	Feature	Cryptography Technique	Protocol	Parameter
17.	Improve Security of Data Access in Cloud Computing Using Location (Vandana et al. 2015)	• Improve the security of data access in the cloud computing using PVT (Scott and Denning 2003b). • Add additional security layer for bank data on cloud.	• Hybrid Algorithm • AES as symmetric algorithm.	Geo-encryption	Latitude, longitude & time
18.	Security to Mobile Banking Using Location Based Encryption (Deshpande et al., 2015)	• Develop banking application using Location Based Encryption. • Provide solution to physical attack using virtualization. • Use LDEA (Location-dependent Data Encryption) algorithm to provide mobility.	• AES and RSA.	LDEA & DTD	Latitude & longitude
19.	An Improved Geo-Encryption Algorithm Location Based Services (Kolapwar 2015)	• Improve the existing Geo-protocol, Data Encryption Standard in Geo Encryption with Dynamic Tolerance Distance and performance by using AES-GEDTD. • Evolve a new algorithm called M-AES-GETD.	• Modified AES algorithm.	DTD	Latitude, longitude, time & velocity
20.	Geo-encryption to Access the Data Using AES Algorithm (Pant et al. 2016)	• Use a security system to safeguard data by using AES encryption algorithm and coordinate of the receiver with tolerance distance. • Design a derivation protocol that allow laptops to derive a key to access the information based on location automatically.	• AES algorithm	DTD	Latitude & longitude
21.	Location Based Security for Online Transaction (Auti, et al., 2016)	• Proposed context-based access control (CBAC) mechanism for Android system.	• AES algorithm	DTD	Latitude, longitude, time & velocity

No.	Title/Author	Feature	Cryptography Technique	Protocol	Parameter
22.	GIS Map Encryption Algorithm for Drone Security Based on Geographical Features (Ngoc, et al., 2016)	• Used Geographic Information System (GIS) map data in drone security. • Geometric objects are extracted from GIS map. • GIS map perform selective encryption in frequency domain of discrete cosine transform.	• GIS map encryption algorithm	Not stated	Not stated
23.	Enhancing Security Using Location and Time (Dalvi et al., 2017)	• A new system is introduced uses location and time using AES algorithm to secure the data.	• AES algorithm	Geo-encryption	Latitude, longitude & time
24.	Enhancing Security of Confidentiality for Mobile Device (Sanjay et al., 2017)	• Provide a security by using Geo encryption techniques based on location or restrict the location and time of the message.	• AES encryption	Geo-encryption	Latitude, longitude & time
25.	"Geo-Encryption Lite" – A location-based Encryption Application for Android (Chaudhari 2017)	• Modified traditional geo-protocol. • Implemented as an Android application.	• AES algorithm	Geo-encryption	Latitude, longitude & time

2.4 Key-generation in RSA Algorithm

RSA algorithm is in the category of cryptography implementation based on public-key (Milanov 2009). It works in pair with a public key and private key. The public key is completely safe to distribute to anyone because it cannot be applied for the reverse process and the decryption process can only be done with the private key. Both keys for the RSA algorithm are generated based on the mathematical theorems and formulations which is equivalent to prime number factorization. It is factoring the large integers and return them back to their original values through reverse steps. The algorithm of prime factorization as shown in eq. (2.1).

$$n \equiv pq$$
$$\textbf{where } p \text{ and } q \text{ are prime numbers.} \qquad \textbf{(2.1)}$$

The private key use for decrypting the cipher text must be kept secret all the times. The calculation of the private key is defined as followed:

$$de \equiv 1 [mod\ \emptyset\ (n)]$$
$$[e, \emptyset(n)] = 1 \qquad \textbf{(2.2)}$$
$$\textbf{where}$$
$$\emptyset\ (n) \text{is the totient function}$$
$$(a, b) \text{is the greatest commomn divisor}$$

The encryption is done by using both public key and private key. The public key belongs to the person who is going to receive the message while the private key belongs to the one who encrypts the message. In the encryption process, the message is converted to a number, let say it is m, by applying the padding scheme method.

$$c \equiv m^e \ (mod\ n) \qquad (2.3)$$

The cipher text is then computing by following formula using exponentiation by squaring method. After the execution of the formula, instead of the original message m, the cipher text c is sent to the receiver.

$$m \equiv c \ (mod\ n) \qquad (2.4)$$

Decryption is the reverse process of the encryption method. The same formula is used by applying reverse padding scheme method. The received cipher text from the sender is applied the following formula in order to get the original message which is encrypted in the sender machine. In order to decrypt the message, the received must use the private key of its own.

2.5 Cloud Storage Security

In group communications such as project groups, there is often a need to distribute information and files electronically. There are needs to share and transfer information in a secure but user-friendly way. Today, the actual transfer method in all modern software solutions is encrypted and secure (Suuronen T., 2018). At the same time, service providers strive to reduce the risks over the clouds and increase their reliability in order to build mutual trust between them and the cloud customers (Harfoushi et al. 2014). However, there are still some discussions on how the software solution or cloud storage provider will guarantee the security of user data inside the cloud storage.

One of the good way to secure data storage, commercial cloud services systems encode each user's data with a specific encryption key (Haibin Zhang 2018). When the user requests to view the data, the decryption key is applied to decrypt the data and then viewed by the users. Decrypted key can be stored either by the service itself, or by

individual users. Most services keep the key themselves, letting their systems see and process user data. These services also access the key when a user logs in with a password and unlocking the data in order to use it. This is much more convenient than having users keep the keys themselves. However, there are still a possibility that their own application might be compromised or hacked, allowing an intruder to read user's files either before they're encrypted for uploading or after being downloaded and decrypted

Another way to maximize cloud storage security, it is advisable to encrypt files using our own encryption software before uploading data to the cloud (Haibin Zhang 2018). To access the encrypted file again, log in to the cloud storage service, download it and decrypt it. This prevents users from taking advantage of many cloud services, like live editing of shared documents and searching cloud-stored files. The best way to protect against that is to use authenticated encryption. This method stores not only an encrypted file, but additional metadata that lets a user detect whether the file has been modified since it was created.

2.6 Summary

This chapter summarizes a description of some type of cryptography algorithm and encryption algorithm base on location. This chapter has identified several techniques of algorithm such as RSA, DES, Triple-DES, and AES that has been used in the geo-encryption algorithm. Besides the algorithm, there are several additional protocols that has been implemented by previous works in geo-encryption which are Geo-encryption Algorithm, Improved Geo-encryption Algorithm, TESLA protocol in Geo-encryption using Loran, Dynamic Tolerance Distance (DTD), and Location Dependent Data Encryption Algorithm (LDEA).

Many researchers have used a symmetric algorithm during encrypting the plaintext in their works either using DES, Triple-DES or AES algorithm. However, majority of the researchers (Abolghasemi, et al., 2013; Auti, et al., 2016; Deshpande et al., 2015; Hamad & Elkourd, 2010; Kolapwar & Ambulgekar, 2014; Manoj, et al., 2014; Pant et al., 2016; Qiu et al., 2007) used AES algorithm to perform the encryption of the plaintext and the decryption of encrypted cipher text. There are 14 out of 25 related existing works that used AES algorithm in their research. Therefore, this thesis will use AES algorithm as one of the techniques in the encryption and decryption process. Table 2.11 shows the comparison between DES and AES.

Based on 25 existing works that has been identified and the clarification in section 2.8 and 2.10 of this chapter, this thesis focuses to use AES algorithm and the encryption keys of the geo-encryption will be stored by the individual user. The encryption key will be based on geographical information which is longitude and latitude coordinate. The following chapter will discuss on the methodology of this research.