Institute of Advanced Engineering and Science

Secrecy and Performance Analysis of Symmetric Key Encryption Algorithms

T.D.B Weerasinghe
BSc.Eng(Hons), MIEEE, AMIE(SL), MSc.Eng candidate, Department of Electrical and Electronic Engineering, University of Peradeniya, Peradeniya 20400, Sri Lanka

Article Info

Article history:

Received May 5th, 2012
Revised May 31st , 2012
Accepted June 2nd, 2012

Keyword:

Block ciphers
Stream ciphers
Encryption time
Secrecy of ciphers

ABSTRACT

In open literature there is a lack of focus on Shannon's secrecy of ciphers as a security measurement of symmetric key encryption, hence in this research, Shannon's theories on secrecy of ciphers were used to calculate the average secrecy of each symmetric cipher used in this research. All secrecy and performance analysis were done using a newly created tool. Analysis is done based on the secrecy level and performance of the algorithm. This paper presents an analysis of some of the widely used symmetric key algorithms which fall under the categories of block and stream ciphers together with the two combined algorithms. [DES, TripleDES, AES, RC2, RC4, Hybrid1 (TripleDES+RC4) and Hybrid2 (AES+RC4) are used]. Analysis is pivoted around on two measurement criteria under two circumstances which are described later in this paper. All the algorithms are implemented in Core Java using classes available in JAVA package javax.crypto. Separate classes are written to calculate the secrecy of ciphers and the encryption time. And also the tool is created using Core Java with the help of Netbeans IDE. As far as the outcome of the research is concerned, the performances of all stream ciphers are higher than that of block ciphers and the combined algorithms have similar performance level to block ciphers. Secrecy levels of block ciphers are comparatively higher than that of stream ciphers as the history says, it is further proved by Shannon's theories in this research. The combined algorithms have more stable secrecy levels.

Corresponding Author:

Department of Electrical and Electronic Engineering, University of Peradeniya,
Peradeniya 20400, Sri Lanka.
Email: tharindu.weerasinghe@gmail.com

1. INTRODUCTION

In this research, selected block cipher algorithms DES, Triple DES and AES (in ECB mode) as well as stream cipher algorithms RC2 and RC4 are analyzed together with the two hybrid algorithms. The idea of combining TripleDES (168bit) and RC4 (128bit) was presented in IFRSA' International Journal of Computing. *(2012 – Volume 2, Issue2)* Therefore the analysis of hybrids done here can be regarded as an extension of that work. Analysis is done based on two measurement criteria under two circumstances.

- **First circumstance:** Variable is the input plaintext size (input is given as text files, size is measured in kilobytes)
- **Second circumstance:** Variable is the input plaintext length, i.e. input character length (inputs are assumed as passwords)
- **Two measurement criteria:**
  (i) Secrecy of Ciphers (According to Shannon's secrecy theorems), (ii) Encryption time

There are two main objectives of this research. One is to identifying the suitable block or stream cipher algorithm which may suit for a better encryption for given scenario (e.g. for a password based authentication, for a hardware/embedded system based encryption requirement and so on) and compare them with the secrecy and performance levels of the two hybrids. And the other objective is to allow the users of the tool introduced by me, to identify the better algorithm based upon the performance of the algorithm and the secrecy of the cipher (this is applicable for the second scenario mentioned in the beginingof the INTRODUCTION) for their applications where they use password authentications. All the algorithms are implemented in Java using the NetBeans IDE and Java Cryptographic classes provided SunJCE. Following default key sizes are used. [3].

Table 1. Algorithm with Default Key

Algorithm	**Default Key Size/Bits**
DES	56
TripleDES	168
AES	128
RC2	128
RC4	128
Hybrid1 (TripleDES + RC4)	168 (for TripleDES) & 128 (for RC4)
Hybrid2 (AES + RC4)	128 (for AES) & 128 (for RC4)

1.1 Hybrid (Stream and Block cipher combined) Algorithms

Since the main objectives of this research is pivoted around symmetric key algorithms, the idea of combining block and stream cipher was in my mind and the combined algorithms, TripleDES + RC4 and AES+RC4 are also included herein for the evaluation. Idea of combining the block and stream ciphers evolved with the imagination of increasing the security. If it increases the security the secrecy values should be higher (as one measurement). Encryption and decryption procedures are shown below:

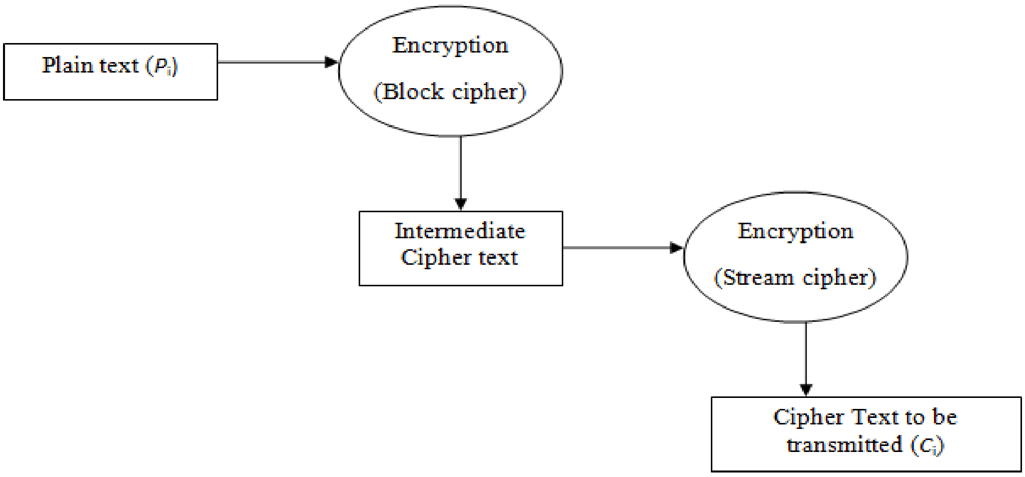


Figure1. Encryption Overview [1]

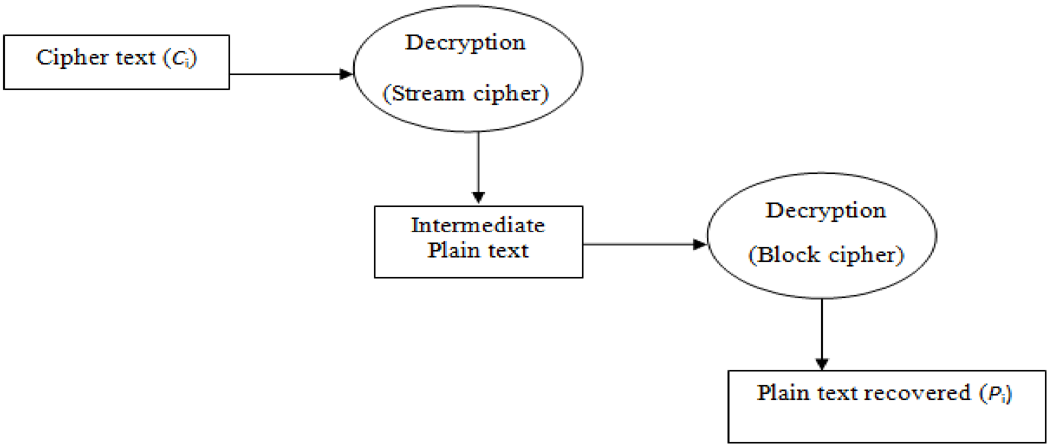


Figure 2. Decryption Overview [1]

Overview of the Java Cryptography Package that is used by me in this research: Tool created by me in-order to make work easier in analyzing the secrecy values and performance for passwords: (Second circumstance of this research). This tool helps the users to check the encryption time and secrecy value (numerically) on screen as soon as they press the relevant button. There is no evidence of any previous work on this type of a tool where it depicts the measurement criteria like performance and security level!

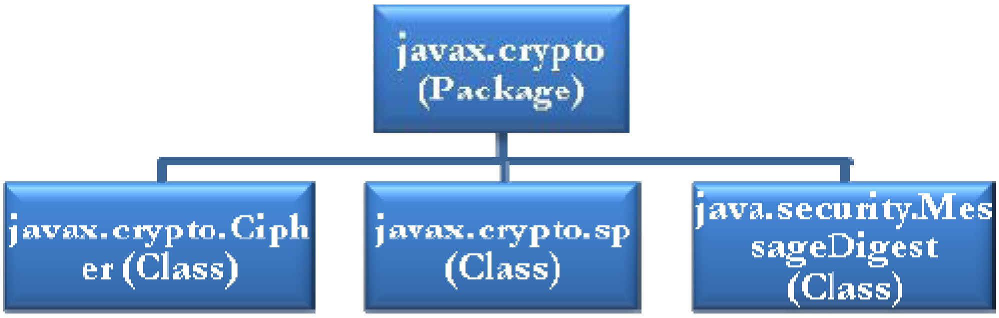
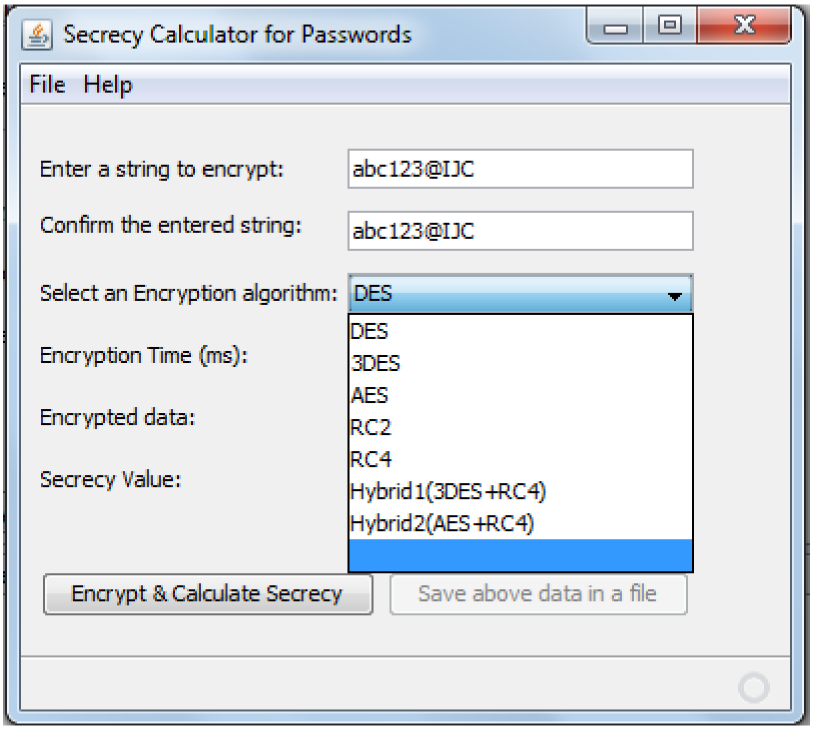


Figure 3. Java Cryptography Package [2]

1.2 Secrecy and Performance Analyzer: Tool created in this research



Secrecy Calculator for Passwords

Enter a string to encrypt: abc123@IJC

Confirm the entered string: abc123@IJC

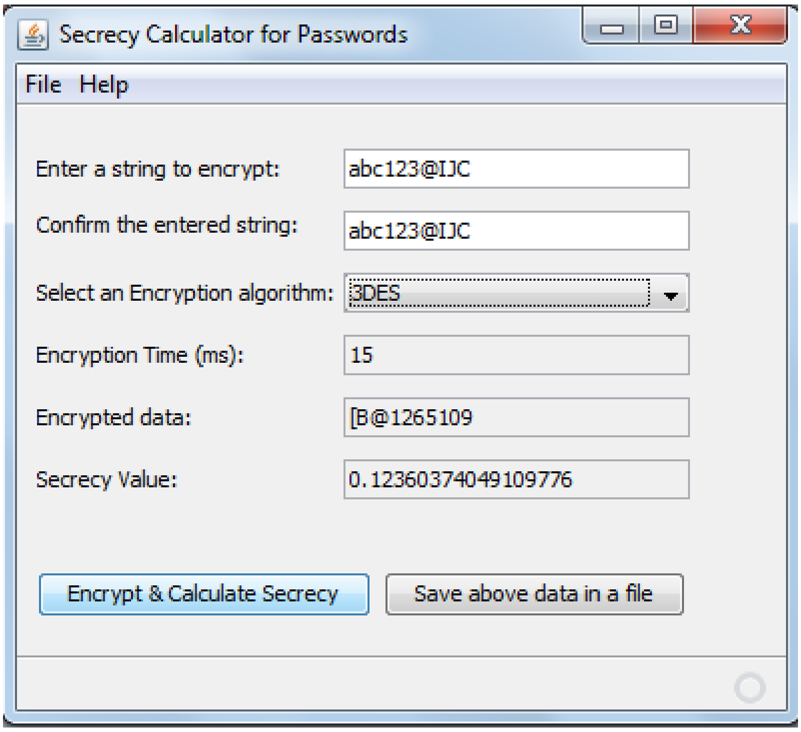
Select an Encryption algorithm: DES

Encryption Time (ms):

Encrypted data:

Secrecy Value:

Figure 4. How to select the intended algorithm from the list



Secrecy Calculator for Passwords

Enter a string to encrypt: abc123@IJC

Confirm the entered string: abc123@IJC

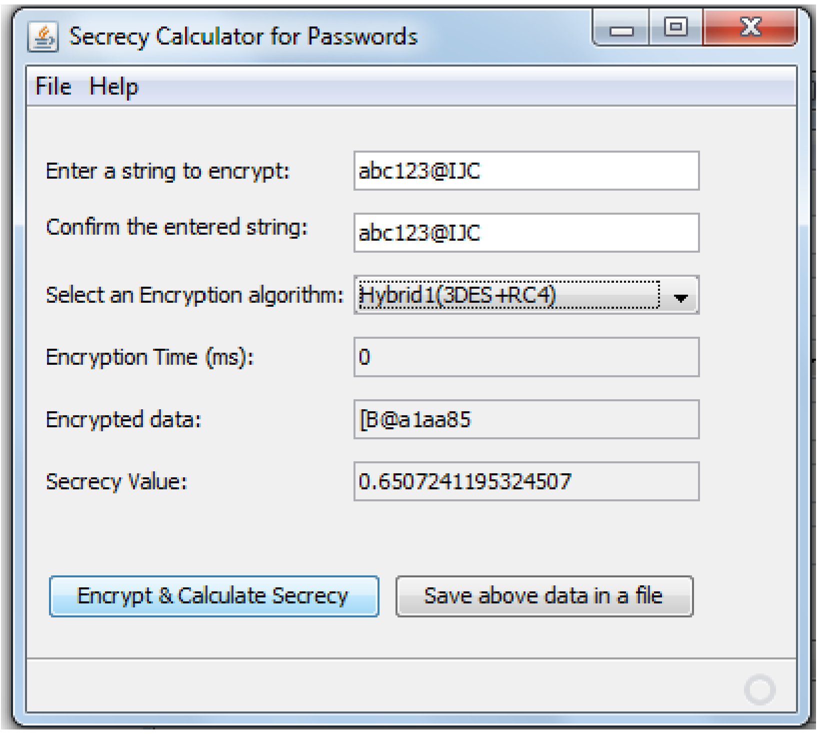
Select an Encryption algorithm: 3DES

Encryption Time (ms): 15

Encrypted data: [B@1265109

Secrecy Value: 0.12360374049109776

Figure 5. After the encryption: Performance and Secrecy values shown for TripleDES


Secrecy Calculator for Passwords
File
Help
Encrypt & Calculate Secrecy
Save above data in a file

Enter a string to encrypt: abc123@IJC

Confirm the entered string: abc123@IJC

Select an Encryption algorithm: Hybrid1(3DES+RC4)

Encryption Time (ms): 0

Encrypted data: [B@a1aa85

Secrecy Value: 0.6507241195324507

Figure 6. After the encryption: Performance and Secrecy values shown for the first Hybrid

1.3 Secrecy of Ciphers – Definitions used and Method of Calculation

For the calculation of the secrecy value of a cipher, the following theories have been followed!

Definition of 'entropy':

The ***entropy*** of a message M, denoted H(M), is the amount of information in the message. It corresponds to the minimum number of bits needed to encode all possible meanings of the message, assuming all messages are equally likely. [4]

- The ***entropy*** of a given message X is defined by the weighted average:

$$H(X) = - \Sigma\{1 \leq i \leq n\}\ p(x_i) \log p(x_i)$$

Definition of 'uncertainty':

The ***uncertainty*** of a message corresponds to the number of plaintext bits that must be recovered when the message is scrambled in ciphertext in order to learn the plaintext. The uncertainty of a message is measured by its entropy. [4]

Definition of 'equivocation':

The ***equivocation*** *is* the uncertainty of a message that can be reduced by given additional information. [4]

- The ***equivocation*** is the conditional entropy of *X* given *Y*:

$$\begin{aligned} H_Y(X) &= - \Sigma\{X,Y\}\ P(X,Y) \log_2 pY(X) \\ &= -\Sigma\{Y\}\ P(Y)\ \Sigma\{X\}\ P_Y(X) \log_2 (P_{Y}(X)) \end{aligned}$$

Definition of secrecy:

The ***secrecy*** of a cipher is measured in terms of the key equivocation Hc(K) of a key K for a given ciphertext C; that is the amount of uncertainty in K given C: [4]

$$\boxed{H_c\,(K) = -\Sigma\{C\}\ P(C)\ \Sigma\{K\}\ P_c(K)\log_2\,[P_c(K)]}$$

Note: *This is used in the secrecy calculation in this research and the above equations are illustrated from theories of Shannon related to entropy and secrecy.* ***Claude Elwood Shannon*** *[April 30, 1916 – February 24, 2001] is called the Father of Information Theory.*

2. RESEARCH METHOD

STEP 1: Method of testing the first scenario

Input is given as text files, size is measured in kilobytes. Numeric values for Secrecy and Performance are obtained! To perform the required tasks separate Java programs are written. Input data size is varied from 5 KB to 100 KB. Input is given as text files. Particular input is read by the relevant Java program and the encryption time and secrecy are calculated and output on the screen. (Note: Text inside the text file comprises alpha numeric characters as well as special characters; always the same input is used to test all the algorithms).

Average encryption time and secrecy of cipher are calculated after 5 rounds of testing for each input (each text file). The aim was to produce two graphical outcomes which show the variation of the Average Encryption Time and Secrecy Value over the input data size.

Algorithms used:

DES, TripleDES, AES, RC2, RC4, Hybrid1 (TripleDES+RC4) and Hybrid2 (AES+RC4)

STEP 2: Method of testing the second scenario

For this step the new tool can be used. (The tool was introduced in the INTRODUCTION section of this paper). Input is considered as a password which comprises alpha-numeric characters with special characters. The variable (the criteria that is taken as the 'X' axis of the graph drawn later) is the type of the password. We know that a password to be strong, the length of it should at least be 8. Hence I have chosen the password length as 10 (thinking it is memorisable). And the type of the password is varied as,

a) All alpha characters (all lowercase letters)
b) All alpha characters (all uppercase letters)
c) All alpha characters (mixed letters, one after other lowercase and uppercase)
d) All numeric characters
e) Alpha numeric characters (lowercase literals mixed with numeric)
f) Alpha numeric characters (uppercase literals mixed with numeric)
g) All special characters
h) Alpha characters mixed with special characters (literal followed by a special character)
i) Numeric characters mixed with special characters (Numeric followed by a special character)
j) Alpha numeric characters mixed with special characters

People who need to compare the password encryption algorithms can use this tool easily and get an idea about the algorithm to be used.

Algorithms used:

DES, TripleDES, AES, RC2, RC4, Hybrid1 (TripleDES+RC4) and Hybrid2 (AES+RC4)

How the secrecy is calculated in both the scenarios:

- Calculate how often each key byte has appeared in the key. [5]

- And then calculate the probability of each byte appears (given the cipher) in the key and get the summation of Pc(K) * $\log_2 P_c(K)$ [for all 256 possibilities of 1 byte key]
- Calculate how often each cipher byte has appeared in the cipher.
- And then calculate the probability of each byte appears in the key and get the summation (for all possibilities of the cipher bytes). This cipher is related to the above key; i.e. this cipher is obtained by encrypting plain text with the above key. Then get the multiplication of the highlighted part and P(C) and finally the summation is considered.
  - Note: This cipher is related to the above key; i.e. this cipher is obtained by encrypting plain text with the above key. Hence I assumed the $P_c(K)$ is the probability of K given C, thus the above probability of K is denoted as $P_c(K)$. Since this measurement is common for all the algorithms this is a reasonable measurement.
  - $P_c(K)$ and P(C) are related to the definitions of secrecy of ciphers mentioned in the INTRODUCTION section.
  - This secrecy value is considered for 1 byte key (1 byte of the key is taken as a sample) as it very hard to calculate the values for all the possible keys (For example TripleDES uses 168 bit key and there are 2^168 possible keys it will take a very long time to calculate a secrecy value related to this (even for a single plaintext). If you consider all the key combinations and take ciphertext you will get a really large number of ciphertext as well.

3. RESULTS AND ANALYSIS

Table.2 Average Secrecy of Ciphers for variable input data sizes

Input Data Size/KB			*Algorithm Name*				
	DES	**3DES**	**AES**	**RC2**	**RC4**	**Hybrid1**	**Hybrid2**
5	0.292269012	0.128433408	0.19964566	0.200991116	0.17480747	0.215101274	0.164702735
10	0.23611892	0.308858194	0.18967428	0.053517532	0.16832912	0.229507753	0.140570463
15	0.129788416	0.200804304	0.22718612	0.063095395	0.1585643	0.223708079	0.154864983
20	0.187728204	0.363490184	0.23297121	0.158409316	0.16556034	0.220657567	0.157609315
25	0.14373826	0.26001987	0.2212405	0.293946407	0.17118074	0.231303128	0.153078474
30	0.189753629	0.237138551	0.24416342	0.163535291	0.17770687	0.22593549	0.163368202
35	0.216438652	0.201835278	0.13793432	0.265800044	0.18044321	0.237518095	0.148283803
40	0.347048094	0.327427522	0.19848262	0.247313884	0.17294332	0.237910337	0.152909746
45	0.22449584	0.239508702	0.29436429	0.178517182	0.17469719	0.239328159	0.152700096
50	0.241861827	0.174385837	0.21884574	0.158331948	0.15607884	0.235069715	0.157499253
55	0.233412102	0.354634619	0.2242389	0.251757115	0.17911076	0.225973132	0.14690322
60	0.233999095	0.324950271	0.38960835	0.156500983	0.16998157	0.23258963	0.142584826
65	0.274178996	0.407747905	0.38679598	0.197893099	0.18945935	0.234729998	0.159486399
70	0.297716416	0.139060063	0.34552624	0.178304332	0.15935405	0.237956204	0.155665889
75	0.17765813	0.128660882	0.24488225	0.187998866	0.15917461	0.233547191	0.160830047
80	0.241487962	0.154629452	0.38880825	0.191202713	0.18243486	0.242777805	0.158196695
85	0.299523626	0.146645331	0.27809376	0.145067195	0.16382689	0.240819397	0.150094016
90	0.251160515	0.159713292	0.15059372	0.15251508	0.18480739	0.242417182	0.151794404
95	0.178627702	0.140878024	0.39539199	0.24121274	0.19212304	0.232567338	0.151607595
100	0.302034118	0.162850973	0.16898309	0.375752334	0.18449194	0.229123519	0.161371889

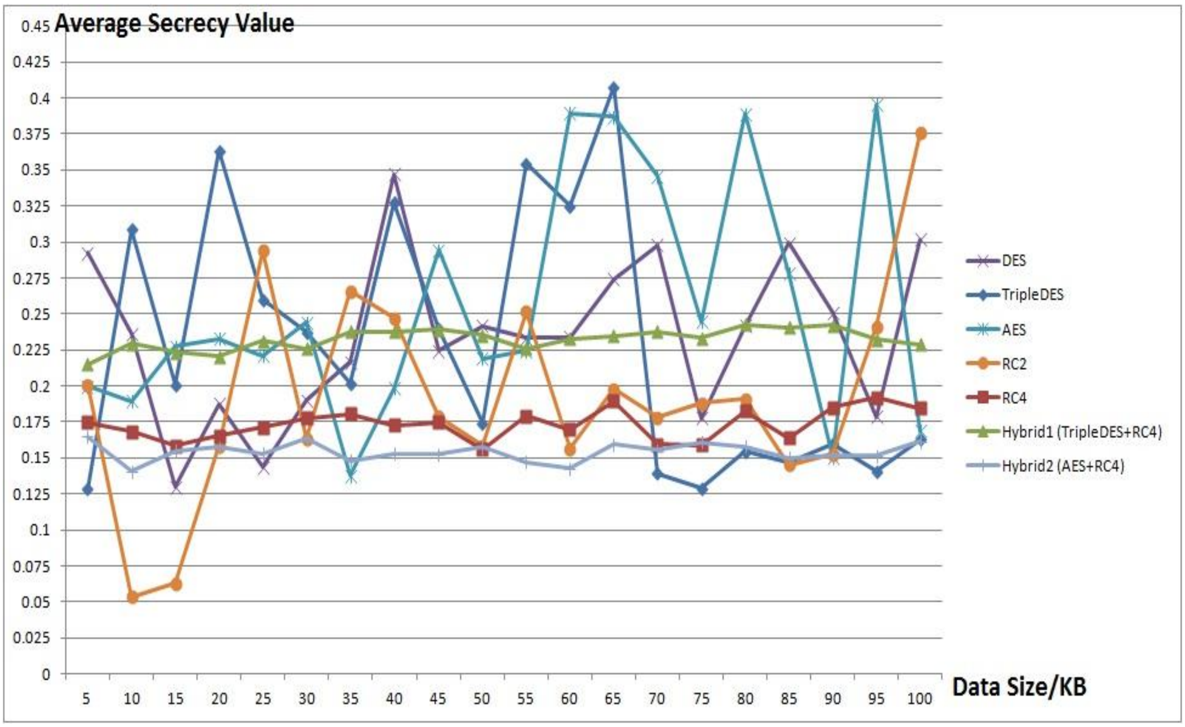


Figure.7 Average Secrecy Value vs. Data Size

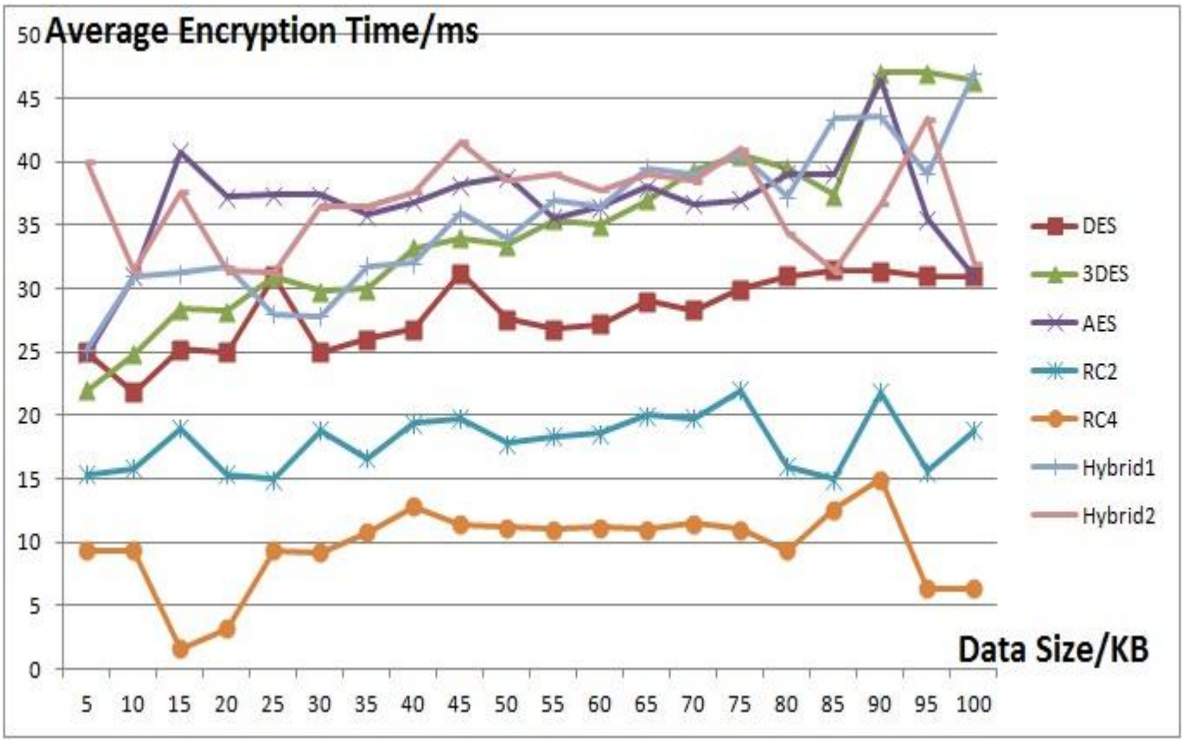


Figure 8. Average Secrecy Value vs. Data Size

Table 3. Average Encryption Times of the Algorithms for variable input data sizes

	Algorithm Name						
Size of Data/KB	**DES**	**3DES**	**AES**	**RC2**	**RC4**	**Hybrid1**	**Hybrid2**
5	25	22	24.8	15.4	9.4	25.2	40
10	21.8	24.8	31	15.8	9.4	31	31.4
15	25.2	28.4	40.8	19	1.6	31.2	37.6
20	25	28.2	37.2	15.4	3.2	31.8	31.4
25	31	31	37.4	15	9.4	28	31.2
30	25	29.8	37.4	18.8	9.2	27.8	36.4
35	26	30	35.8	16.6	10.8	31.8	36.4
40	26.8	33.2	36.8	19.4	12.8	32	37.6
45	31.2	34	38.2	19.8	11.4	36	41.6
50	27.6	33.4	38.8	17.8	11.2	34	38.6
55	26.8	35.4	35.6	18.4	11	37	39
60	27.2	35	36.4	18.6	11.2	36.4	37.8
65	29	37	38	20	11	39.5	39
70	28.3	39.3	36.7	19.8	11.5	39	38.5
75	30	40.5	37	22	11	40.5	41
80	31	39.5	39	16	9.4	37.2	34.4
85	31.5	37.4	39	15	12.6	43.4	31.4
90	31.4	47	46.4	21.8	15	43.6	36.7
95	31	47	35.5	15.6	6.4	39	43.4
100	31	46.4	31	18.8	6.4	47	32

Table 4. Average Secrecy Value of the Algorithms for variable password

PASSWORD	AVERAGE SECRECY VALUE						
	DES	**TipleDES**	**AES**	**RC2**	**RC4**	**Hybrid1**	**Hybrid2**
abcdefjhij	0.120064	0.211442	0.220294	0.152196	0.100323	0.14323	0.083943
ABCDEFGHIJ	0.193014	0.11469	0.171083	0.189834	0.181156	0.093939	0.201284
aAbBcCdDeE	0.181084	0.212044	0.146352	0.273072	0.188536	0.386806	0.178586
1234567890	0.28503	0.249001	0.052026	0.116567	0.203247	0.162768	0.154636
a1a1a1a1a1	0.184799	0.237323	0.213369	0.130003	0.170538	0.086776	0.32096
A1A1A1A1A1	0.436293	0.228906	0.133558	0.145045	0.136033	0.127259	0.129023
!@#$%^&*()	0.245562	0.102037	0.21416	0.105268	0.206375	0.133919	0.171644
a#a#a#a#a#a#	0.169108	0.070293	0.146133	0.252558	0.215879	0.09713	0.162394
1$1$1$1$1$	0.15684	0.127811	0.209567	0.182847	0.21089	0.129449	0.09659
a1#a1#a1#a	0.262433	0.254738	0.089083	0.057755	0.013588	0.089514	0.055769

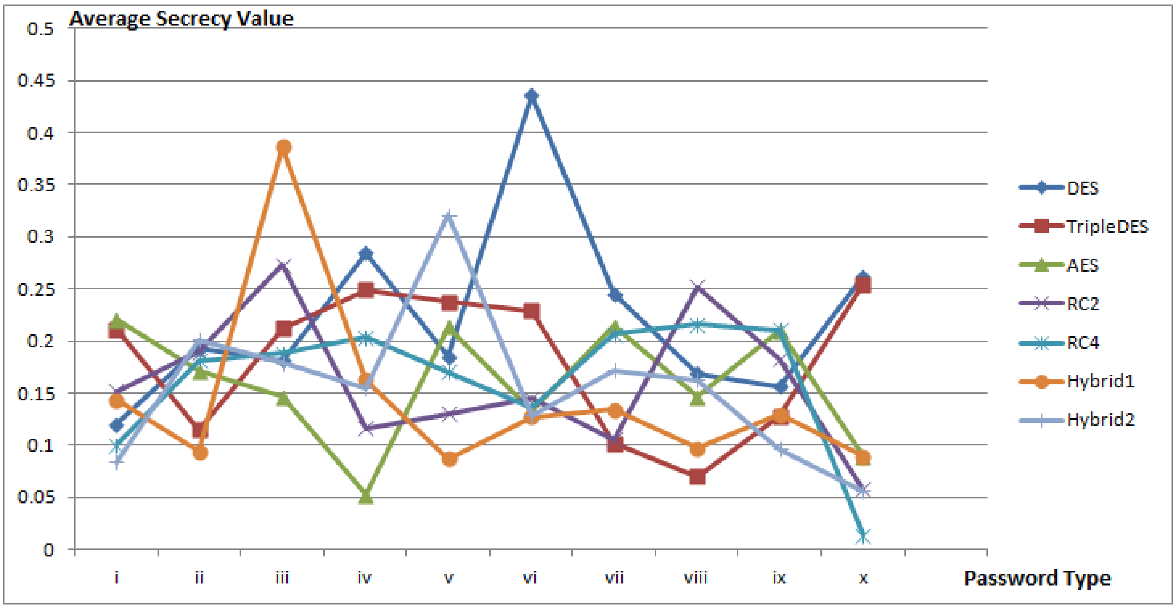


Figure 9. Average Secrecy Value Vs Password Type

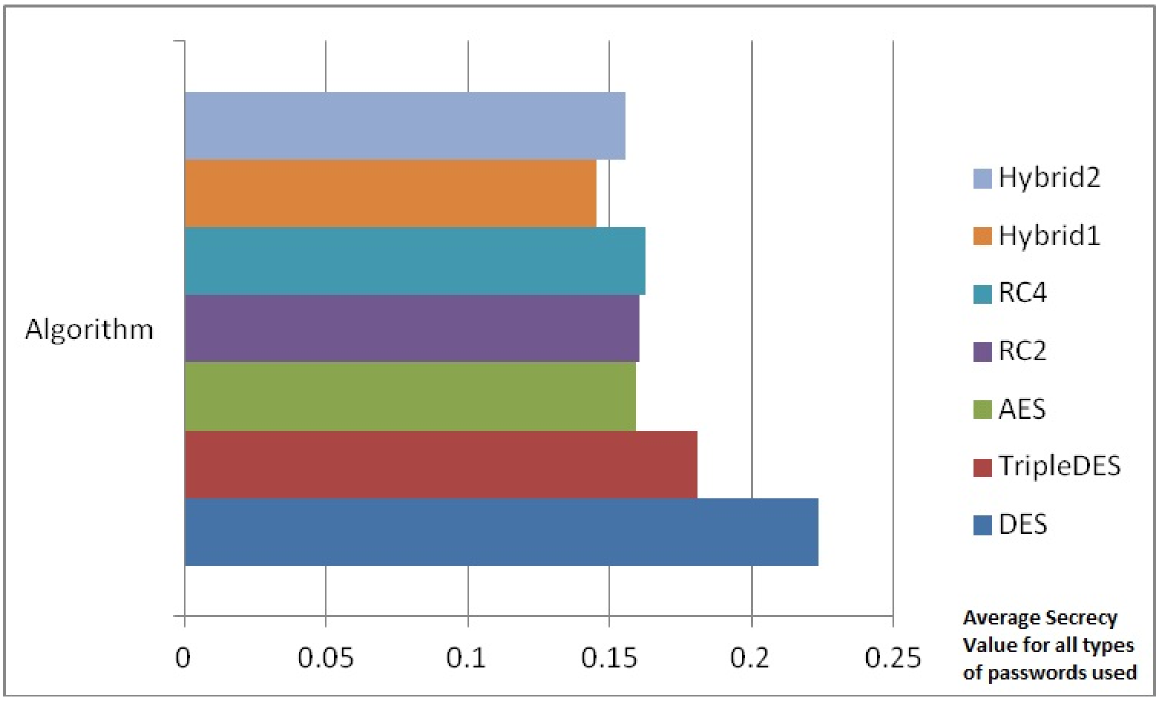


Figure 10. Algorithm Vs Average Secrecy Value for all the Password Types used in this research

4. CONCLUSION

As far as my opinion is concerned, combined algorithms show stability in the secrecy level than block and stream ciphers.What happens to the secrecy levels of AES and TripleDES compared to DES especially in the second scenario, must be studied further. Performances of the combined algorithms are low compared to that of block and stream cipher algorithms.

1. Conclusion w.r.t. scenario-1, Secrecy of ciphers of the algorithms: when the secrecy of ciphers is concerned it is clearly obvious that the hybrid ciphers obtained by combining TripleDES and RC4 as well as AES and RC4 are the most stable of all. They not show drastic changes over the input data size whereas others have shown dramatic changes over the input data size. Among the two hybrids the Hybrid1 (TripleDES+RC4) implies to be the steadiest and also the most secured as it has higher secrecy values. As expected, stream ciphers give us the lower secrecy and block ciphers give us the higher secrecy. (Refer Figure 7). Apart from the two stream cipher algorithms, all the others are grouped in one area. (As expected) – Refer Figure 8. As far as the block ciphers and the two hybrids are concerned the data size increases the encryption time also increases. But if we look at the stream cipher algorithms' performance it is obvious that the performance is steady than the block ciphers which also have lower performance.(execution time is lower means performance is higher)
2. Conclusion w.r.t. scenario-2, Different input types (passwords) give different secrecy values for different algorithms. (Refer Figure 9) The most significant feature as I see is the average secrecy value given by the algorithms for last password type, which comprises alpha numeric characters with special character which I assumed that would have given higher secrecy values for all the algorithms, but apart from DES and TripleDES all the other algorithms have given very low secrecy values. Thus there is a question mark whether we can select those types of passwords (alpha numeric character with special characters) as strong passwords for all the symmetric key encryption algorithms. If the Figure10 is concerned, DES has given overall best secrecy value (without considering the password type) which may lead to futher discussions among the interested researchers. As a future work in this area, the password length can be varied obtain the secrecy values and then can be compared with that of the criteria which has the variable password types. Anyway, the designers/researchers in the area of password based authentications can use the tool (Refer Figure 4, 5 and 6) that is created by me (if they require contacting me and request)

REFERENCES

[1] Mailewa, D.M.A.B., Weerasinghe T.D.B., Perera S.P.J., Munasinghe C.A., 2008: "Types and Modes Combined Algorithm for Data Encryption and Decryption", proceedings, *13th Peradeniya University Research Sessions, Peradeniya, Sri Lanka*, 181-182

[2] Sachin Majithia, Kumar Dinesh, 2010: "Implementation and Analysis of AES, DES and Triple DES on GSM Network", *IJCSNS International Journal of Computer Science and Network Security*, VOL.10 No.1, 298-303

[3] http://docs.oracle.com/javase/6/docs/technotes/guides/security/SunProviders.html

[4] www.ece.uvic.ca/~itraore/elec567-04/notes/elec6704-6-2.pdf

[5] http://web17.webbpro.de/index.php?page=entropy

BIOGRAPHY OF AUTHOR

T.D.B WEERASINGHE
BSc.Eng(Hons) Computer Engineering, MIEEE, AMIE(SL), MSc.Eng (Information & Communication Engineering) candidate, Department of Electrical and Electronic Engineering, University of Peradeniya, Peradeniya 20400, Sri Lanka

Contact No: 0094 716 860 396 Email: tharindu.weerasinghe@gmail.com
Postal Address (Home): 296, Kandy Road, Millawa, Kurunegala 60000, Sri Lanka.