

**CleverSecure ONLINE System**

By

**Shamimi Bt. Abd. Kadir**

Dissertation submitted in partial fulfillment of  
the requirements for the  
Bachelor of Technology (Hons)  
(Information & Communication Technology)

**JANUARY 2008**

Universiti Teknologi PETRONAS  
Bandar Seri Iskandar  
31750 Tronoh  
Perak Darul Ridzuan

# CERTIFICATION OF APPROVAL

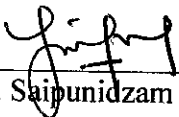
**CleverSecure ONLINE System**

By

Shamimi Bt Ab. Kadir

A project dissertation submitted to the  
Computer and Information Sciences  
Universiti Teknologi PETRONAS  
In partial fulfillment of the requirement for the  
BACHELOR OF TECHNOLOGY (Hons)  
(INFORMATION COMMUNICATION AND TECHNOLOGY)

Approve by,

  
\_\_\_\_\_  
(Mr. Saipunidzam Mahamad)


UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

January 2008

## CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



---

SHAMIMI BINTI ABD KADIR

## **ABSTRACT**

CleverSecure Online System (COS) implements on the use of popular algorithm which is Blowfish algorithm to encrypt and decrypt the data that can be save in directory or send through email. This application also can encrypt the folder that contains the private information such as business plan etc. The purpose of the development is to provide the security service to the all information in our PC and allow users to protect confidential sensitive messages and documents sent over insecure medium of Internet. With data encrypted such that only the intended recipients can decrypt and read it, it prevents others from accessing the confidential information. The system also enables users to attach the encrypted file into their email to send directly to the receiver. The email system used to send the encrypted file to the third party. During the development of COS, the specific methodology used which is Prototyping Process. It contains several phase; Requirement Analysis; Preliminary Design; Prototype Creation; Evaluation; Prototype Refinement; and lastly Final Delivery. The outcome of this project is the online system that enable user to use the application easily with the friendly user interface to secure their private information.

## **ACKNOWLEDGEMENT**

First and foremost, all my gratitude to The Almighty, Allah SWT that had given me the strength, wisdom and patient all the way in completing this project within the time given. My heartfelt thanks goes to everyone who has been supportive and giving me their point of views in realizing the project. Without the existence of many people around, the project would reach to nothing. Special thanks to my supervisor, Mr. Saipunidzam Mahamad who has been understanding, supportive, patient and helping me a lot from the beginning to the end. To my family who has given support, be it physically, psychologically and financially in giving me the ongoing spirits to complete this project as a fulfillment to complete my degree in Information and Communication Technology from UTP. Thanks also to all my friends that had been very supportive and helpful during the development process of the application. Thanks for the ideas, critics, comment and suggestion to improve this project especially my colleagues that have shared the same experience in making each individual's Final Year Project a success.

## TABLE OF CONTENTS

<b>CERTIFICATION</b>	ii
<b>ABSTRACT</b>	iv
<b>ACKNOWLEDGEMENT</b>	v
<b>LIST OF FIGURES</b>	ix
<b>LIST OF TABLES</b>	x
<b>ABBREVIATIONS AND NOMENCLATURES</b>	xi
<b>CHAPTER 1: INTRODUCTION</b>	
1.1 Background of Study	1
1.2 Problem Statement	2
1.3 Objectives and Scope of Study	
1.3.1 Objectives	3
1.3.2 Scope of Study	3
1.4 Relevancy of Study	4
1.5 Feasibility of Study	4
<b>CHAPTER 2: LITERATURE REVIEW AND/OR THEORY</b>	
2.1 Background	5
2.2 Encryption and Decryption Process	6
2.3 Blowfish Algorithm	7
2.4 Strong Cryptography	8
<b>CHAPTER 3: METHODOLOGY/PROJECT WORK</b>	
3.1 Project Phase	
3.1.1 Requirement Analysis	10
3.1.2 Preliminary Design	11
3.1.3 Prototype Creation	11
3.1.4 Evaluation	12
3.1.5 Prototype Refinement	12
3.1.6 Final Delivery	13

3.2 Tools and Equipment	14
3.2.1 System Requirement	14
3.2.2 Software and Tools for Development	

## **CHAPTER 4: RESULT AND DISCUSSION**

4.1 System Background	15
4.2 System Functionalities	15
4.3 System Design	16
4.3.1 Use Case Diagram	16
4.3.2 Sequence Diagram	17
4.3.3 Class Diagram	18
4.3.4 System Workflow	19
4.4 The Application	
4.4.1 Main Page – User Login	20
4.4.2 Method or Algorithms	20
4.4.3 Encryption / Decryption	21
4.4.4 Email Service	22
4.4.5 Forum	22
4.4.6 Feedback	23
4.5 Evaluation	24
4.5.1 Effectiveness of the application	25
4.5.2 Ease of Interaction	25
4.5.3 System Reliability and Security	25
4.6 Discussion	26

## **CHAPTER 5: CONCLUSION AND RECOMMENDATION**

5.1 Conclusion	27
5.2 Recommendations	28

## **REFERENCES**

29

## **APPENDICES**

- Appendix A : Project Gantt Chart
- Appendix B : System's Guide
- Appendix C : System's Code



## LIST OF FIGURES

Figure	Page
Figure 1.1 : The cost involve in securing data at the user level are minimum	2
Figure 2.1 : The relationship between plaintext and ciphertext, and the encryption and decryption processes that transform them	6
Figure 2.2 : The Feistel Structure of Blowfish	7
Figure 3.1 : Prototyping Process	10
Figure 4.1 : Use Case Diagram for CleverSecure Online	17
Figure 4.2 : Sequence Diagram for Search and Encrypt File	18
Figure 4.3 : Class Diagram for CleverSecure Online System	18
Figure 4.4 : System's Workflow	19
Figure 4.5 : The screen shot of Main Page	20
Figure 4.6 : The Screen Shot of Method or Algorithm function	21
Figure 4.7: Screen Shot of Encryption Process	21
Figure 4.8: Screen Shot of Email Service	22
Figure 4.9: Screen Shot for Forum Function	23
Figure 4.10: Screen Shot of Feedback Function	23
Figure 4.11: Effectiveness of the application towards the user	24
Figure 4.12: Ease of interaction within the application	25

## **LIST OF TABLES**

<b>Table</b>	<b>Page</b>
Table 1 : List of System Requirements	14
Table 2 : List of Software and Tools for Development	14

## **ABBREVIATIONS AND NOMENCLATURES**

<b>OOP</b>	<b>Object-Oriented Programming</b>
<b>GUI</b>	<b>Graphical User Interface</b>
<b>PC</b>	<b>Personal Computer</b>
<b>OOP</b>	<b>Object Oriented Programming</b>
<b>COS</b>	<b>CleverSecure ONLINE System</b>
<b>UTP</b>	<b>Universiti Teknologi PETRONAS</b>
<b>PGP</b>	<b>Pretty Good Privacy</b>

# CHAPTER 1

## INTRODUCTION

### 1.1 Background of Study

Nowadays, the security is very important in communication technology. Encryption is now used in protecting information within many kinds of civilian systems, such as computers, networks (e.g. the internet e-commerce), mobile telephones, and bank automatic teller machines. Computer security is needed to protect the valuable data in our computer from unauthorized users. It's like "a house and the things in it" analogy, where the computer same as the house, apartment or condominium. For example, we routinely lock the doors and close the windows when leave, and don't give the keys to just anyone. Another alternative, some of people may install a security system to their house. The door, windows and security system is the things that provide the security to our house. Let's now apply on the computer, we can add a firewall, an anti-virus program, patches, and file encryption to improve the level of security on the computer.

Encryption comes together with the decryption process. Once we encrypt the file or data, we need to decrypt it to get back the original. This process is very important when a user need to send a private document or text to the third party. There are many encryption systems that allow user to do the process mentioned above. But the system is the applications that need a user to install it on their PC or laptop. So for the first time, I tried to develop the web based encryption service to promote the use of cryptography technique.

## 1.2 Problem Statement

The problem rose when the private file or document in the PC was accessed by someone and the email was attacked by the hacker. The network level security is not enough secure to make sure that the PC cannot be access by unauthorized user. So because of this problem, I have tried to develop a web service for encryption and decryption process to allow any online user to encrypt and decrypt the file easily without install any software on their PC. It's called CleverSecure ONLINE System (COS).

I hope with the web service it could help the users to protect their file for sending to other party and indirectly improve user awareness about the important of data security. The target user of the system is online user. Within this research, I found that the security of the information must be implementing on the network security level and desktop level. The cost of the implementation at desktop level is cheaper than network level and the difficulties to implement it also different. Please refer to Figure 1.1.

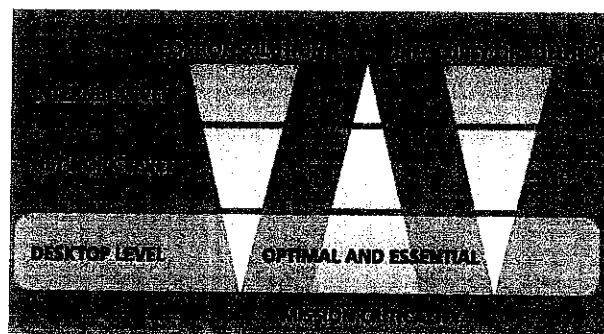


Figure 1.1: The cost involve in securing data at the user level are minimum

## **1.3 Objectives and Scope of Study**

### **1.3.1 Objectives**

Below are the objectives of the system:

- To develop web based file encryption service that can help user in protecting the valuable data or any private file while sending to the third party.
- To encourage user's awareness about the computer security by using the forum provided in the web page to discuss about the current issues.
- To implement the email service so that user can directly send the encrypted file after encryption process.
- To give the information to the user about the methods or algorithm available for encryption and other information about cryptography.

### **1.3.2 Scope of Study**

The project scopes are as follows:-

- Develop and generate the encrypted file using blowfish encryption method.
- Design solutions that ease the user to secure their file without installing the application to their computer. But they need the internet connection.
- Study on how to use the Blowfish algorithms during the encryption and decryption process. Besides know the advantages of the algorithms, how its work, their strength and weaknesses.
- Encrypt not only files but text messages. Security file service web site has special Text Encryption Assistant to encrypt important messages being sent via e-mail.
- Identify and choose suitable development steps among available software development model for identified system.

#### **1.4 Relevancy of Study**

The study is match to Information Technology (IT) field because it is concentrates with Web development and Internet Programming. The study will start with understanding the concept of encryption, the processes and analysis the level of strength of other existing encryption system. Then continue with the analyzing requirement and designing the solution. Finally, the study is about doing the programming and coding. Besides that it allows IT student to improve their programming skills in many different languages and allows students to apply the real situation in developing web based service.

#### **1.5 Feasibility Study**

The project is scheduled to finish within one year. By using the study on the method, concept of internet programming, the goal may be accomplish since the web development could be evaluate at each phase. The project cost is only cover the tools used and could limit the cost by using open source software. The knowledge in internet programming and mathematical operation of the algorithm chosen are required to finish the project.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1 Background

The purpose of the computer security is to devise ways to prevent the weaknesses from being exploited. To understand what preventive measures make the most sense, we consider what we mean when we say that a system is “secure”. When we talk about computer security, we mean that we are addressing three very important aspects of any computer related system: -

- Confidentiality ensures that computer-related assets are accessed only by authorized parties.
- Integrity means that assets can be modified only by authorized parties or only in authorized ways.
- Availability means that assets are accessible to authorized parties at appropriate times.

Other than computer security, another thing that we must when talk about security is Computer Crime. Computer crime is any crime involving a computer or aided by the use of one. Many studies attempted to determine the characteristics of computer criminals. By studying those who have already used computers to commit crimes, we may be able in future to spot likely criminals and prevent crimes from occurring.



## 2.2 Encryption and Decryption Process

Basically in computer security the word of “encryption” is synonym and can be used to refer to the cryptography. Encryption is defined as the transformation of data into a form in which it cannot be made sense of without the use of some key. Such transformed data is referred to as cipher text. Use of a key to reverse this process and return the data to its original (or plaintext) form is called decryption [4]. For purposes of security; the more difficult it is to decrypt the cipher text, the better.

On the other hand, if the algorithm is too complex, takes too long to do, or requires keys that are too large to store easily, it becomes impractical for use in a personal computer. Therefore, some balance must be reached between strength of the encryption (that is, how difficult it is for someone to discover the algorithm and the key) and ease of use. For practical purposes, the encryption need only be strong enough to protect the data for the amount of time the data might be useful to a person with malicious intent.

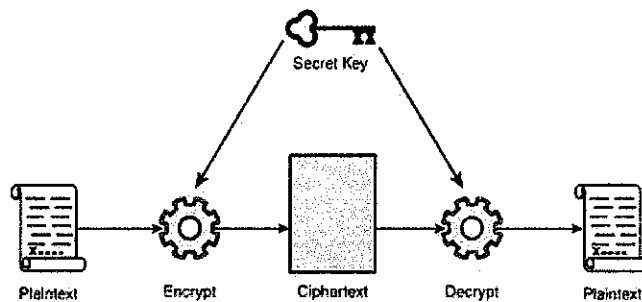


Figure 2.1: The relationship between plaintext and cipher text, and the encryption and decryption processes that transform them.

### 2.3 Blowfish Algorithm

Blowfish is a keyed, symmetric cryptographic block cipher designed by Bruce Schneier in 1993 and placed in the public domain. As a public domain cipher, Blowfish has been subject to a significant amount of cryptanalysis, and full Blowfish encryption has never been broken. Blowfish is also one of the fastest block ciphers in public use, making it ideal for an open uses product. Blowfish has a 64-bit block size and a key length of anywhere from 32 bits to 448 bits [1]. The Figure 2.2 below shows Blowfish's F-function.

The function splits the 32-bit input into four eight-bit quarters, and uses the quarters as input to the S-boxes. The outputs are added modulo 232 and XOR to produce the final 32-bit output. Since Blowfish is a Feistel network, it can be inverted simply by XOR P17 and P18 to the cipher text block, then using the P-entries in reverse order.

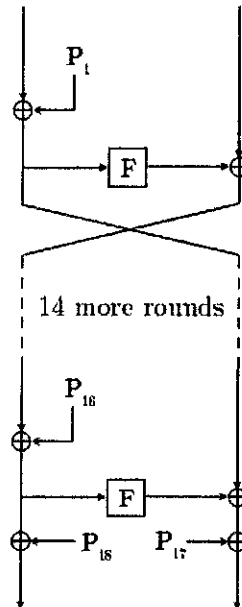


Figure 2.2: The Feistel Structure of Blowfish

## 2.4 Strong Cryptography

*"There are two kinds of cryptography in this world: cryptography that will stop your kid sister from reading your files, and cryptography that will stop major governments from reading your files. This book is about the latter."*

--Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*.

Cryptography can be *strong* or *weak*, as explained above. Cryptographic strength is measured in the time and resources it would require to recover the plaintext. The result of *strong cryptography* is ciphertext that is very difficult to decipher without possession of the appropriate decoding tool. How difficult? Given all of today's computing power and available time — even a billion computers doing a billion checks a second — it is not possible to decipher the result of strong cryptography before the end of the universe. [12]

No one has proven that the strongest encryption obtainable today will hold up under tomorrow's computing power. However, the strong cryptography employed by PGP is the best available today. Vigilance and conservatism will protect you better, however, than claims of impenetrability.

## **CHAPTER 3**

### **METHODOLOGY/PROJECT WORK**

#### **3.1 Project Phases**

There are many examples of software development models, but methodology that suitable for finishing a system within a limited time evolved from the iterative approach such as spiral model, prototyping model, reuse model and others. So in order to increase the system development speed and cut the time, the most suitable approach is the Prototyping Model. Prototyping model start with developing prototype (an early approximation of a final system or product) then it is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed [ 5] .

Prototypes help users get an idea of what the system will look like, and make it easier for user to make design decisions without waiting for the system to be built. Prototyping is most beneficial in systems that will have many interactions with the users. This model also works best in scenarios where not all of the project requirements are known in detail ahead of time. It is an iterative, trial-and-error process that takes place between the developers and the users [5]. Prototyping has several benefits: The designer and implementer can obtain feedback from the users early in the project. The users can give their feedback and compare the developed system with their specification. It also allows developer to estimate their time to finishing the system according to mile stones. There are several steps to develop a product using prototyping model.

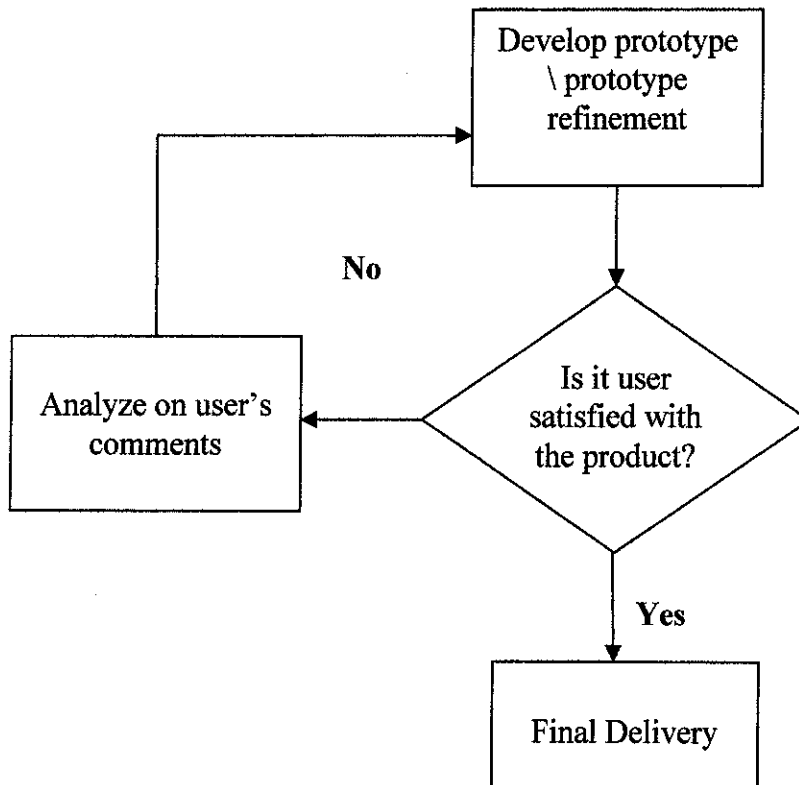


Figure 3.1: Prototyping Process

### 3.1.1 Requirement Analysis

In this phase, the system functional and non-functional requirements were collected from many different sources. The collection of requirements could be done by research on existing system and from written references. It also done by studying on existing non-computerized system and understand some usual operation in encryption/decryption process.

In order to gather the requirement of the system, several techniques used by doing the research on description of cryptography and make an observation to the existing system. The research on cryptography is conducted by surfing on the internets and finds the book or journal related to topic from the library. At this stage, we need to know about the encryption and decryption process, how it's very important in data security and understand the method/algorithm used in these process. This is the best techniques to identify the suitable method used for the process.

Besides that, during the requirement analysis, we need to observe on the existing system: - what their weaknesses are and how the system impact to the user's awareness on data security. User requirement also important to make sure that the system developed is satisfied their needs and desire.

### **3.1.2 Preliminary Design**

After the initial phase of requirement information is collected, or new information is gathered, it is rapidly integrated into a new or existing design so that it may fold into the prototype. In this phase, the system data flow will be recognized by identifying its input, process and output. Interface and database design will be involved. The system is designed by using Unified Modeling Language (UML). See Appendix A to see about the use case diagram.

### **3.1.3 Prototype Creation**

A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product. At this stage, the system's coding was developed. The phase is focus on programming. Some internal testing was done in this phase to ensure the prototype is running fine. During the Part A FYP presentation, the first prototype is developed to get the comment from users and supervisors. During the first output, only the main operation of the system was finished. Basically the purpose of the first prototype is to present the idea on what is the system ability.

The creation of the prototype will be repeating again and again by adding the new function and module in the system. The system also may be modified after prototype refinement phase.

### **3.1.4 Evaluation**

The users and supervisor thoroughly evaluate the first prototype, noting its strengths and weaknesses, what needs to be added, and what should to be removed. At this phase, the comments will be collect and analyze again to enhance the performance of the system in terms of function and others. Once

in two weeks, the prototype is presented to the supervisor to keep up the changes makes to the system.

The evaluation also occurred by other IT and IS lecturers department during Part A presentation. The suggestions were gathered and then were discussed with the supervisor in order to accept or not. The evaluation step was iterated as many times as necessary to the next prototype to see the comment and suggestion again.

### **3.1.5 Prototype Refinement**

The first prototype is modified, based on the comments supplied by the supervisor, and a second prototype of the new system is constructed. This is to ensure that the prototype is developed to be more effective and efficient based on the suggestion from supervisor. In this phase, the prototype was going back to the design and prototype modification again and then was assessed again. The preceding steps are iterated as many times as necessary, until the supervisor satisfied that the prototype represents the final product desired.

In this phase, the prototype also was added with the new functions or requirements and then will be integrated with the current prototype. The prototype and its documentation were refined each time after evaluation process. During the next evaluation, the new prototype was presented.

### **3.1.6 Final Delivery**

After many prototype refinements have been done and the prototype has been enhance and appears as a complete system, the prototype was delivered as a final product. The final delivery will be on the part B FYP presentation.

The system is not exactly following the structure and rules in prototyping model. Some features need to be adjusted to suit in the environment. First, unlike in prototyping model, the system will not be tested and assessed in real environment and with real users. However, this system only will be revised by supervisor. Therefore, the reliability and dependability will not be assessed but only its performance in functionalities. None user's participation involved. Second, because of the system is not implemented in a real environment, there is no routine maintenance is carried out on a continuing basis. But based on the prototyping models, the system needs to maintain and support to prevent large-scale failures and to minimize downtime. The prototype that will be developed in this phase may not fulfill the requirements in initial definition. The requirements that are not carrying out will be added in prototype refinement phase.



## 3.2 Tools and Equipments

During the development, the software and tools required in order to develop the interface, coding and database. It is important to make sure that it can be used easily and the chosen is based on the programming language used.

### 3.2.1 System Requirements

Table 3.1: List of System Requirements

Device / Software / Tools	Requirements
Operating System	Windows Vista™ Home Basic
Processor	Intel Celeron M. 1.6 GHz
Disk Space	80 GB
Memory	1 GB
Peripherals	Printer
Network	Fast Ethernet Adapter
Web Server	Apache
Database	MySQL server

### 3.2.2 Software and Tools for Development

Table 3.2: List of Software and Tools for Development

Software / Tools	Description
Adobe Photoshop CS2	Tool to be used to designing the web interface
Dreamweaver MX 2004 & Ms. Expression Web	Tool to be used for designing the interface and programming codes.
Logo Design Studio 2006	Tool to be used to design the logo for the website

## **CHAPTER 4**

### **RESULT AND DISCUSSION**

#### **4.1 System Background**

CleverSecure Online is a solution used for users that want to send their secure file or data to the third party. By using this system, users could do the operation of encryption and decryption without installing the software to their PC. Now they can find the service online and simply encrypt the file to send it to the others. In order to do the encryption process, the system automatically will generate the key and the method used is Blowfish algorithm. Other than that, the system will allow user to use the email service and make a discussion room, give the feedback and comment about the computer and data security.

#### **4.2 System Functionalities**

The system functionalities are group to 6 main functions which are Encryption, Decryption, Method, Email, Forum and Feedback. For the first page - Home, there is the login that allow user to enter the username, password and provide a brief explanation about the website. After users login to the web site, they can choose whether to make an encryption or decryption process. Before the users make a process done, they should know about the method used and method available for data security. Not all the method mentioned are available in this website.

Other than that, they can use the email service to send the encrypted file and involve in a discussion room which is the forum to discuss about the computer and data security. The user also can discuss about the technology today and ask any question about the computer security. The active member or administrator will answer their question as soon as possible. Then, for the feedback function, it allows user to make a suggestion or comment about the web.

### **4.3 System Design**

#### **4.3.1 Use Case Diagram**

In this section the Unified Modeling Language (UML) used to emphasize the functional requirement of the system from user's point of view. The use case diagram used to present a graphical overview of the functionality provided by a system in terms of actors, their goals and any dependencies between those use cases.

Figure 4.1 below shows the use case diagram for COS. The Actor or users can access the information about the algorithm used and available for data security. Then they also can simply search their file in directory and make an encryption process. Besides that, the user can decrypt file by searching their encrypted file in directory. The user can discuss and ask the question regarding the process using the forum. Forum is used to discuss about the problem arise and new technology in computer security today. From here, we can improve the user awareness about the knowledge of computer security. They also can give the feedback/suggestion and comment about the system functionalities.

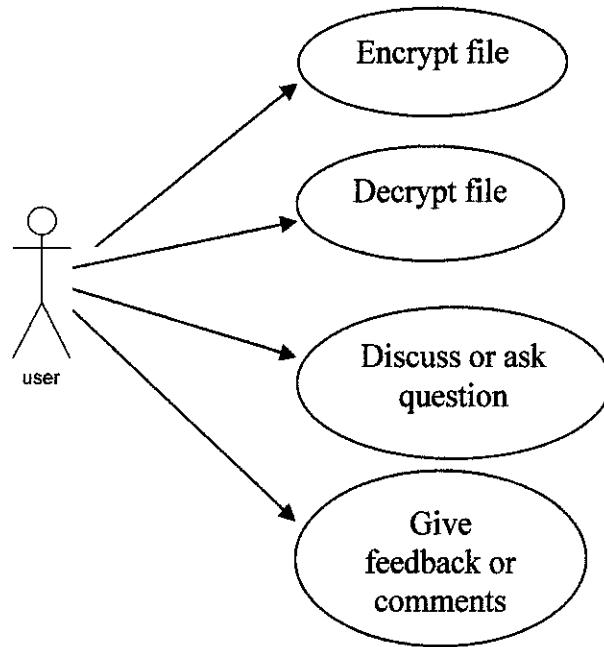


Figure 4.1: Use Case Diagram for CleverSecure Online

#### 4.3.2 Sequence Diagram

Figure 4.2 shows the sequence diagram for search file and encryption process. This diagram use message passing is to show the interaction between the objects. First, user will select the option through the main menu (User Interface) whether to encrypt or decrypt the file. If the file is selected, user will search the file using file dialog browser and the encryption process begins. User will choose the file location for encrypted file after encryption process finished.

Figure 4.3 shows the class diagram. It shows the interaction between the classes which are file, user and method. The class of users has their own userID and password to allow them to make an encryption/decryption process. The function of createUser() will be used to create a new user in the database. These user will choose the file location and the file will be encrypt using the available method.

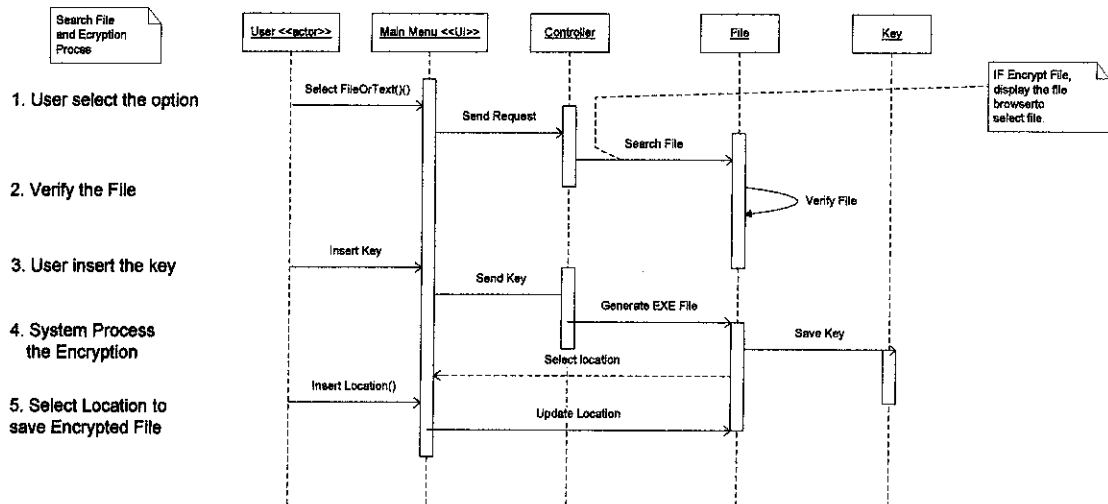


Figure 4.2: Sequence Diagram for Search and Encrypt File

### 4.3.3 Class Diagram

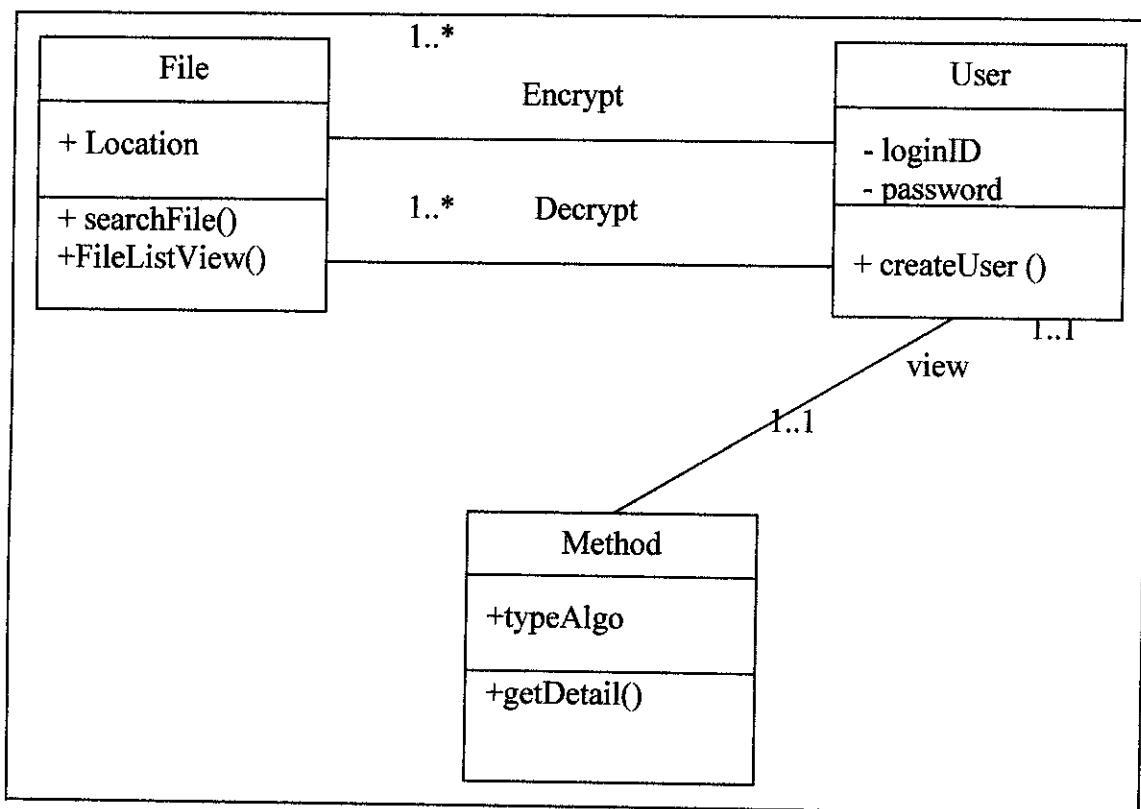


Figure 4.3: Class Diagram for CleverSecure Online System

#### 4.3.4 System's Workflow

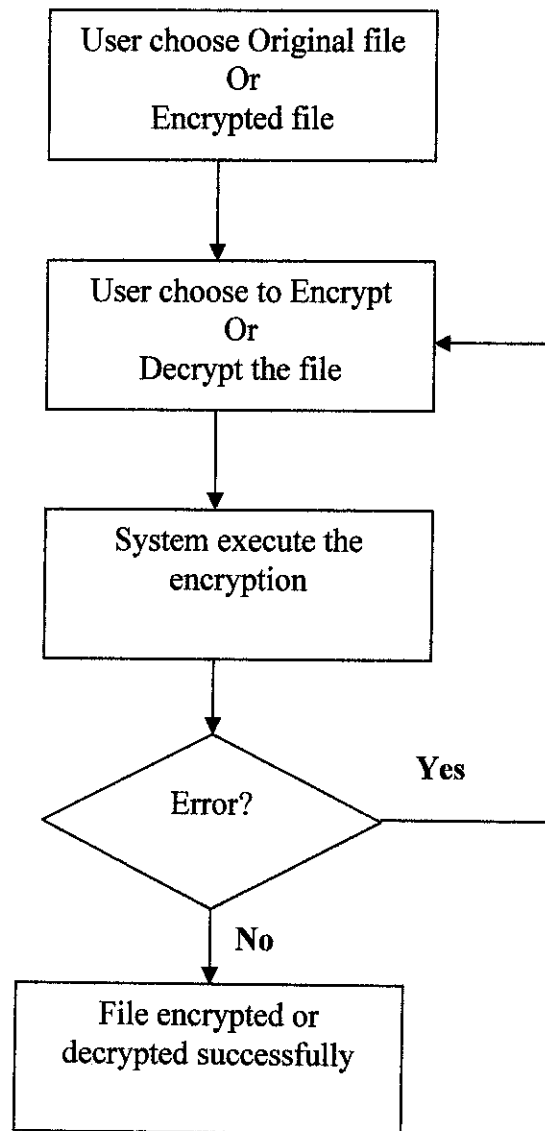


Figure 4.4: System's Workflow for COS

In this system workflow (See Figure 4.4.), the first task occurred would be the selection the file by the user whether the original file or encrypted file. After the selection, users also need to choose whether to encrypt or decrypt the file. Given the key by the users, the system will perform the encryption or decryption process. If there is an error, the system will prompt the users indicating that there is an error occurred.

## 4.4 The Application

The interface for CleverSecure Online System is designed using Macromedia Dreamweaver tools. Until now, it divided into 6 functions. The functions are:-

### 4.4.1 Main Page – User Login

A member can directly login by using their userID and password and for a new user; they can click on “click here” to go to the registration page. Figure 4.5 shows the Home page for CleverSecure Online. This page also allow user to get the information regarding the encryption and enable user to login as a member.

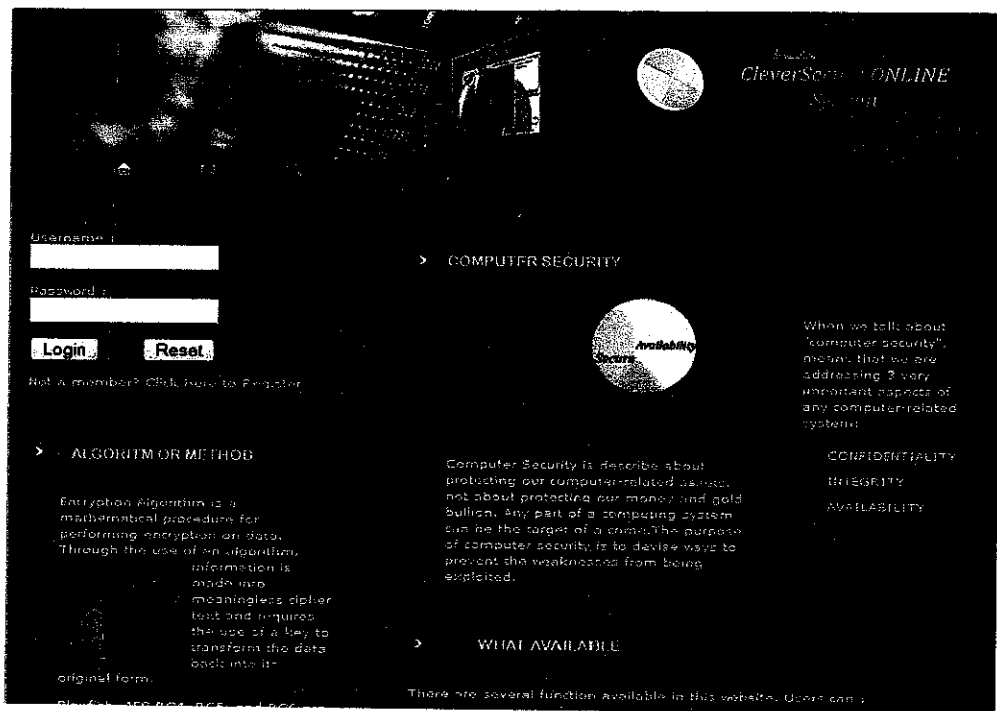


Figure 4.5: The screen shot of Main Page

### 4.4.2 Method or Algorithm

Within this function, user can get the information about the algorithm used for encryption on this online system and it also provide the information of other algorithm available for the data security. See Figure 4.6.

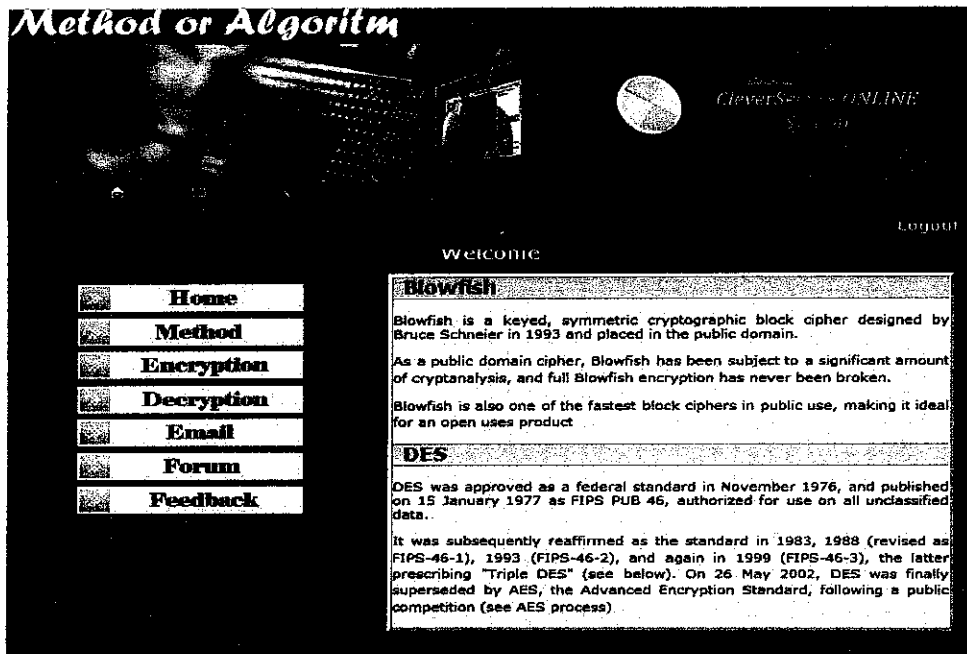


Figure 4.6: The Screen Shot of Method or Algorithm function

#### 4.4.3 Encryption / Decryption

Figure 4.7 shows the interface for encryption process. First of all, users need to browse the file and find the suitable path to locate the encrypted file. Same as the decryption, firstly users need to find the location of encrypted file so that the decryption process will be executed. Besides that, user can add some file for encryption process to encrypt them together. This feature is enable users make a process quickly.

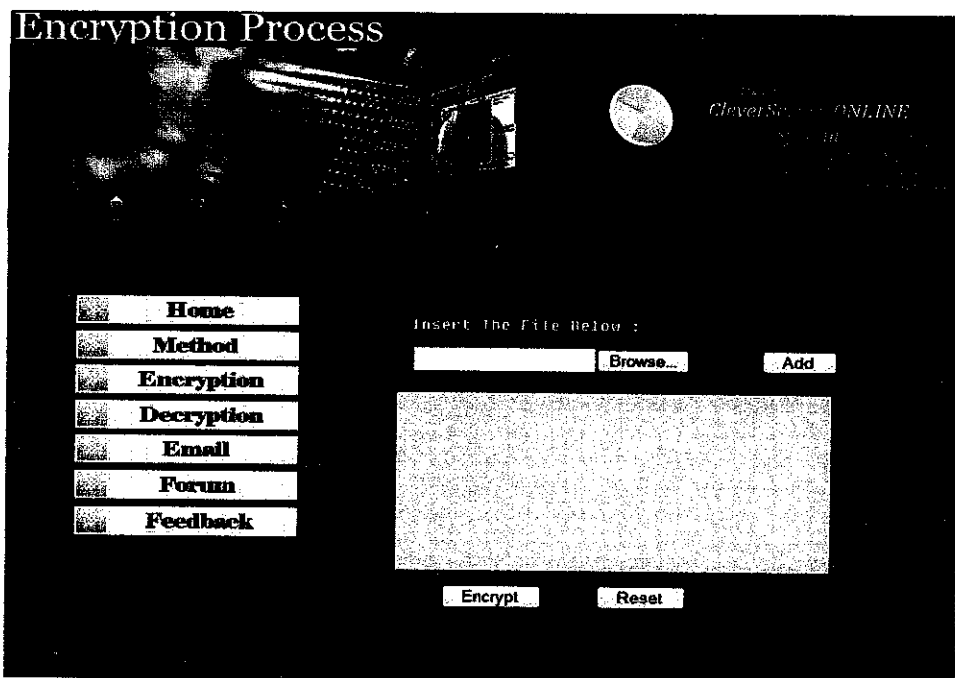


Figure 4.7: Screen Shot of Encryption Process



#### 4.4.4 Email Service

In this function, users can use the email service to send the encrypted file directly to the recipient. Here user also can send the file by attached it together with the message. For the message, its will not be encrypted and it only encrypt using network level during message transmission.

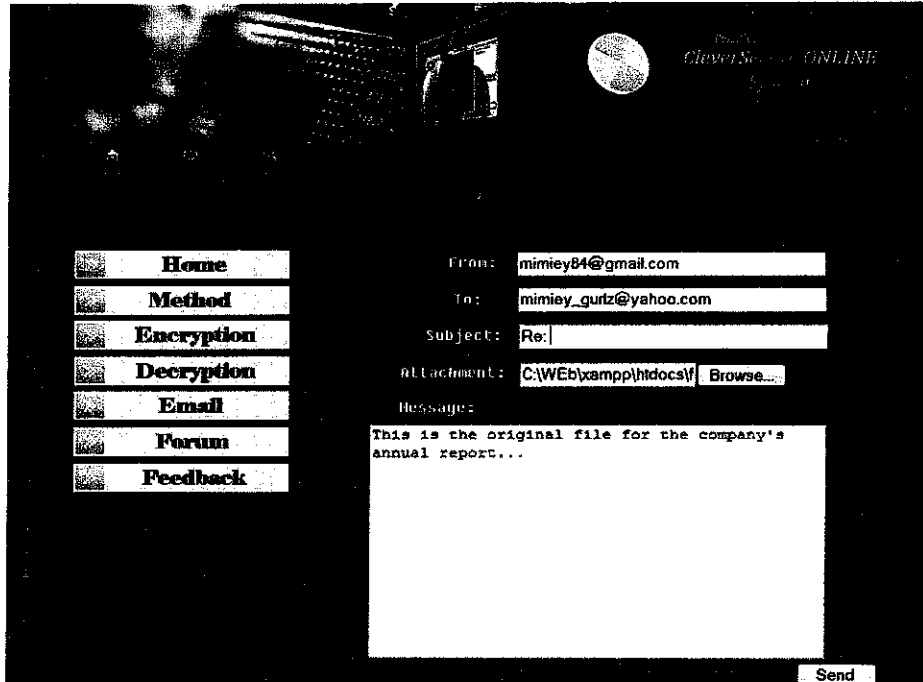


Figure 4.8: Screen Shot of Email Service

#### 4.4.5 Forum

For the forum function, the registered user can ask the question and discuss the current issues of computer security, available algorithm and any problem related with the function provided by this online system. The purpose of the forum provided by administrator is to encourage user awareness about the data security. It also used to expose the users with the available algorithm used and how the process works to discuss about it among the member. This forum is suitable for the users who are very interest with the computer security and computer science. Figure 4.9 shows the main page of forum after the registered user login.

Home | Help | Search | Admin | Profile | My Messages (0) | Members | Logout

---

**Hello mimi0484**  
 you have 0 messages, 0 are new.  
 ▶ Show unread posts since last visit.  
 ▶ Show new replies to your posts.

Total time logged in: 2 hours and 32 minut

---

**General Category**

Board name	Topics	Posts	Last post
<b>Announcement</b> Feel free to talk about anything and everything in this board.	2	2	Last post by mimi0484 in Begin with security. April 07, 2008, 01:31:37 pm

---

**Method or Algorithm**

Board name	Topics	Posts	Last post
<b>Blowfish</b> Designed by Bruce Schneier in 1993 and placed in the public domain.	0	0	
<b>AES</b> Symmetric block cipher that is intended to replace DES at the approved standard for a wide range of applications.	0	0	
<b>RC4</b> Cipher invented by Ron Rivest, co-inventor of the RSA Scheme	0	0	
<b>DES / 3DES</b> DES is a block cipher with 64-bit block size that uses 56-bit keys	0	0	
<b>RSA</b> proposed by three mathematicians, Ron Rivest, Adi Shamir and Len Adleman gave a concrete example of how such a method could be implemented.	0	0	

**MARK ALL MESSAGES AS READ**

Figure 4.9: Screen Shot for Forum Function

#### 4.4.6 Feedback

In this function, users can drop their complaint, compliment or suggestion to the admin so that this online system will be updated from time to time. For the 1<sup>st</sup> version, there are a lot of complaints from user. By using this function, users can easily send any comment or feedback directly to the administrator. See Figure 4.10.

**Feedback and Complaint Form**

This is a  COMPLAINT  COMPLIMENT  SUGGESTION

What is your  ?

**B I U ABC ↻ ↺ ↻ ↻**

What would you like us to do about it?

**B I U ABC ↻ ↺ ↻ ↻**

Date	Time	User's Name	Your E-mail
11/04/08	04:29:01	<input type="text"/>	<input type="text"/>

**Please rate our performance (Optional)**  
 Please take a moment to rate our service. Your feedback will help us evaluate our performance so that we can better serve you.

Please tick  one

	Excellent	Good	Average	Poor
1. Waiting time for encrypt file	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. User interface design	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure 4.10: Screen Shot of Feedback Function

## 4.5 Evaluation

After the system is completed, an evaluation is made. This evaluation is done during the interaction between ten numbers of users and an online system.

The system is evaluated on 3 sections:-

### 4.5.1 Effectiveness of the application

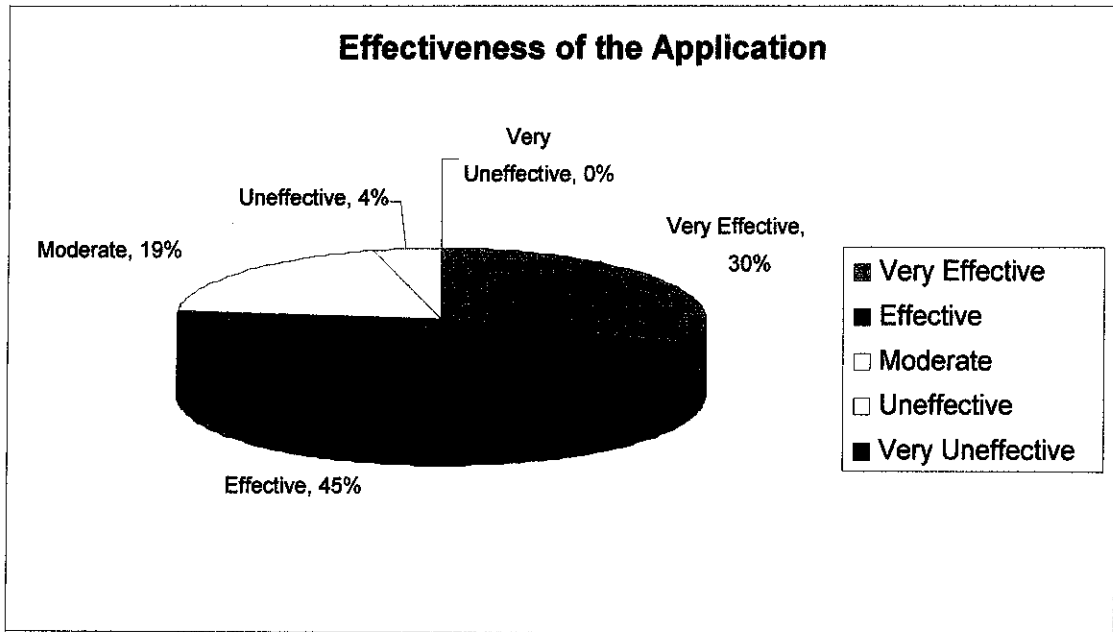


Figure 4.11: Effectiveness of the application towards the user

The first attribute of the evaluation is regarding the effectiveness of the interface for the Online System. The interfaces include the interactive representation interfaces for the main page frameset. From the evaluation session, the highest percentage of the respondents thought the application is *Effective* and said that a lot of things could be done to enhance the effectiveness of the application. On the other hand, 4% of the respondents thought that the application is *Not Effective*. This is due to their opinion that for a person new to such technology of the file security, it takes some time to get along with.

#### 4.5.2 Ease of interaction

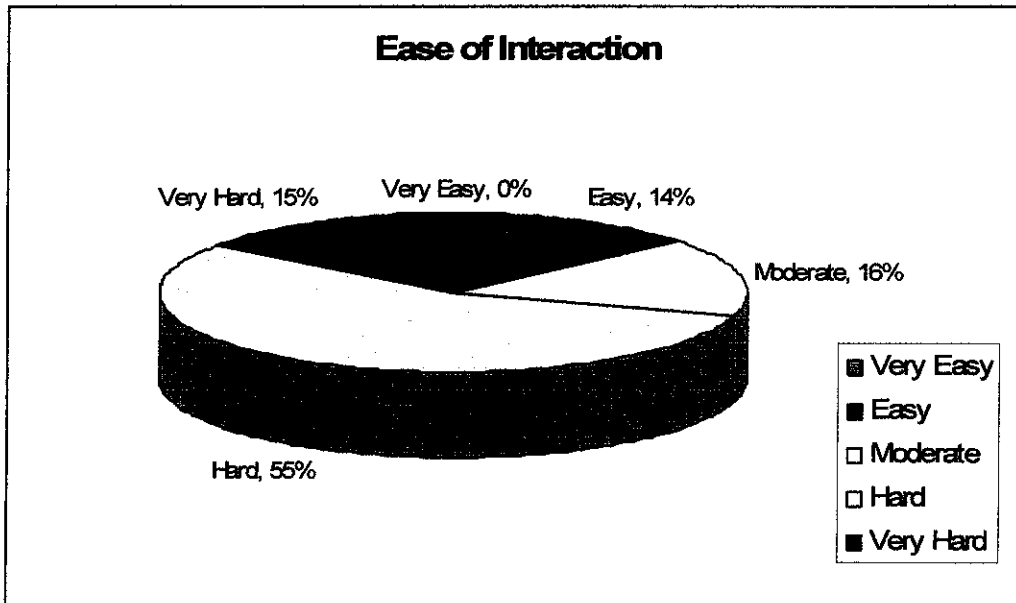


Figure 4.12: Ease of interaction within the application

The second attribute of evaluation is the ease of interaction in CleverSecure Online System. This attribute represents the interactive web design, ease of finding files and giving feedback or joining forums. The result in Figure 4.12 shows that 55% of the respondents thought that the interaction of the application is *Hard* mostly caused by the performance of the application and difficulties to view the intended.

#### 4.5.3 System Reliability and Security

A system cannot be deemed reliable if it is not secure, so the concepts of security and reliability are intrinsically linked. The security here refers to the encryption process which will produce the encrypted file with the strong security from intruders. A reliable system has been defined as one that is capable of operating without material error, fault, or failure during a specified period in a specified environment.

## 4.6 Discussion

The system is online to eliminate user task to install the program in their PC. So every time users want to make a process, they need to have the internet connection. Today the internet is easy to get and the latest one is the wireless technology. Everywhere you go, the information could be access via the internet and many features can be used without any problem. There are many application available that allow user to make an encryption and decryption process.

For example Mars Encapsulator Encryption system that enable user to create a self extracting file called Capsule in .EXE format. Some email systems do not support that type of file to send it to the others. The focus is only to decrypt file and data on the PC. Because of the problem, the online system is created to make easy for the user to use the email system to send the encrypted file.

The method used for this system is blowfish algorithm. The strong algorithm is measured by their complexity of the operation itself. The most popular algorithm is PGP. PGP combines some of the best features of both conventional and public key cryptography. When a user encrypts plaintext with PGP, PGP first compresses the plaintext. Data compression saves modem transmission time and disk space and, more importantly, strengthens cryptographic security [12]. Their operation is complex compare with the blowfish algorithm. But for the 1<sup>st</sup> version of this online system, I came out with the simple algorithm and will be enhanced in the future that allow user to encrypt their file using PGP.

## **CHAPTER 5**

### **CONCLUSION AND RECOMMENDATIONS**

#### **5.1 Conclusion**

In conclusion, the system is focusing on the transforming the file, folder and data which will be used by the user to protect the information from unauthorized access. The system is to make the user feel secure when they send their file to the third party. Indirectly it will encourage user to know more about the technology on computer security. Because of the technology today is changing rapidly, it is most important to the user to have the knowledge on that area.

The discussion room on the CleverSecure Online system enable user to discuss and ask any question regarding that topic. From there, we can get the knowledge and share the new information. As a member of the system, user will get the information about the method use and available in encryption process. Actually, users need to be exposed about the algorithm to get the basic idea what it is all about.

## **5.2 Recommendation**

The system still can be improved to be more a good online system. The possibility of internet connection is slow should be considered to make the system more reliable. It cannot be denied that file encryption also has its advantage. There are many hacker out there might want to access the encrypted file and try to break up the algorithm. So from time to time, the new algorithm should be added to make sure that the process is strong enough.

The quality of encryption and decryption process must be improved in order to encourage user to use it. Among recommendations are first; ensure the system user will allow the users to post their code of the method so that users can encrypt the file using new method. Indirectly this system also will attract the professional programmers that have highly interest on cryptography to develop the code for specific method. Second is to build a system that can encrypt the file from mobile as well.

## REFERENCES

- [1] SplashData, Inc. Blowfish Encryption. Retrieved August 30, 2007 from World Wide Web: <http://www.splashdata.com/splashid/blowfish.htm>
- [2] CRCnetBase Product, ITKnowledgeBASE. Retrieved September 3, 2007 from World Wide Web: [www.itknowledgebase.net/](http://www.itknowledgebase.net/)
- [3] Bruce Schneier, Blowfish Paper. Retrieved September 10, 2007 from World Wide Web: <http://www.schneier.com/paper-blowfish-fse.html>
- [4] Wikipedia, the free encyclopedia.. Retrieved August 27, 2007 from World Wide Web: <http://en.wikipedia.org>
- [5] What is Prototyping Model?, Retrieved March 18, 2008 from World Wide Web:  
[http://searchciomidmarket.techtarget.com/sDefinition/0,,sid183\\_gci755441,00.html](http://searchciomidmarket.techtarget.com/sDefinition/0,,sid183_gci755441,00.html)
- [6] Harold F. Tipton, CISSA, CISP. (February 2002). Computer at Risk -Ten Years After. Lifetime Achievement Award Recipient, 37-41.
- [7] Converse & Park With Morgan (2004), *PHP5 and MySQL Bible*, Wiley Publishing, Inc.
- [8] Encrypting data with Blowfish Algorithm. Retrieved October 28, 2007 from World Wide Web: <http://www.us.design-reuse.com/articles/article5922.html>
- [9] Pfleeger (2003), *Security in Computing*, Pearson Education, Inc.
- [10] William Stalling (2003), *Cryptography and Network Security*, Pearson Education, Inc.
- [11] Sommerville (2004), *Software Engineering 7*, Addison Wesley Publishers Limited
- [12] How PGP Works. Retrieved April 11, 2008 from World Wide Web:  
<http://www.pgpi.org/doc/pgpintro/>

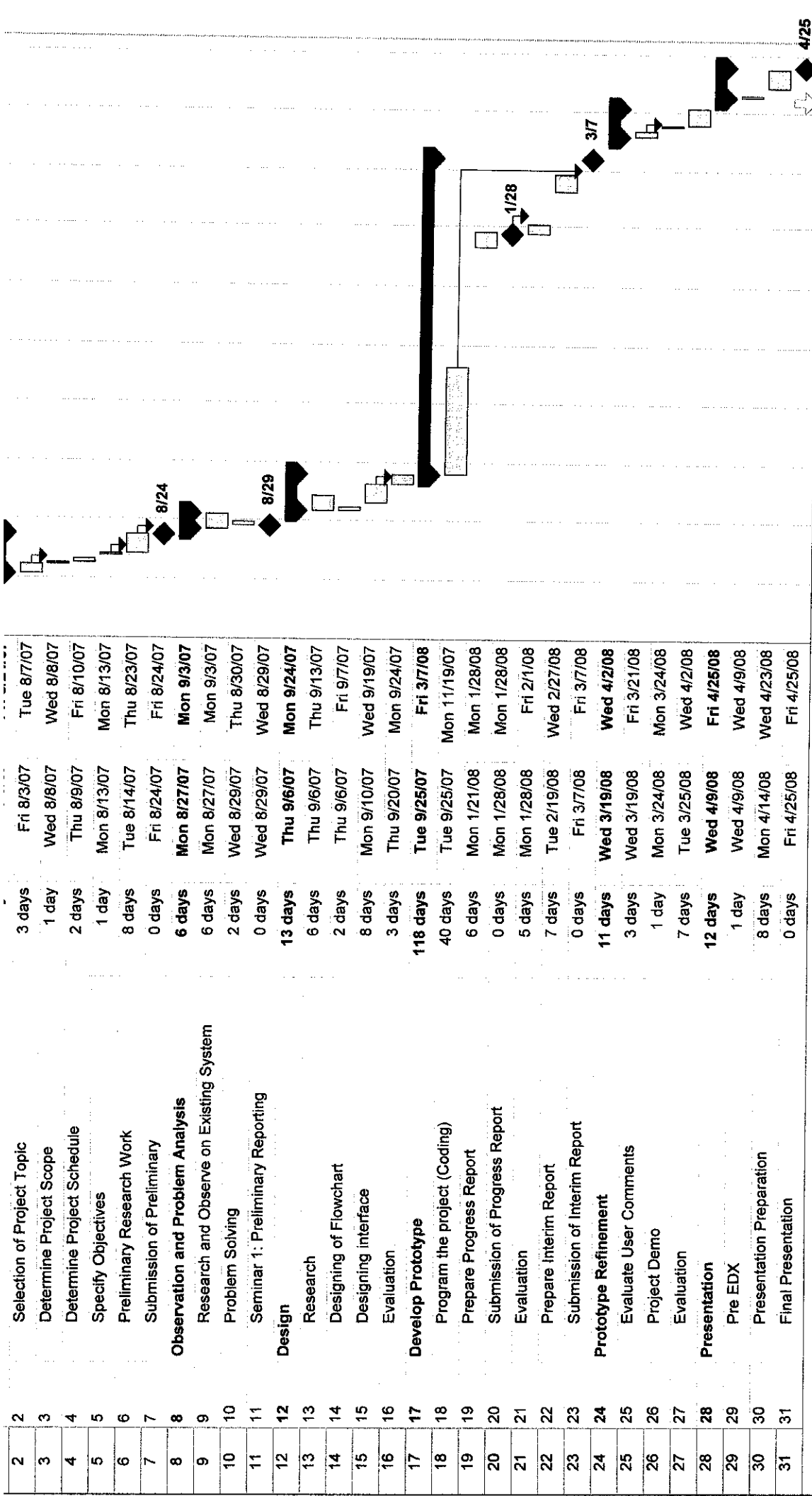


## APPENDICES

APPENDIX A	: Project Gantt chart
APPENDIX B	: System's Guide
APPENDIX C	: System's Code

## **APPENDIX A:**

# **PROJECT GANTT CHART**



2	Selection of Project Topic	3 days	Fri 8/3/07	Tue 8/7/07
3	Determine Project Scope	1 day	Wed 8/8/07	Wed 8/8/07
4	Determine Project Schedule	2 days	Thu 8/9/07	Fri 8/10/07
5	Specify Objectives	1 day	Mon 8/13/07	Mon 8/13/07
6	Preliminary Research Work	8 days	Tue 8/14/07	Thu 8/23/07
7	Submission of Preliminary	0 days	Fri 8/24/07	Fri 8/24/07
8	<b>Observation and Problem Analysis</b>	6 days	Mon 8/27/07	Mon 9/3/07
9	Research and Observe on Existing System	6 days	Mon 8/27/07	Mon 9/3/07
10	Problem Solving	2 days	Wed 8/29/07	Thu 8/30/07
11	Seminar 1: Preliminary Reporting	0 days	Wed 8/29/07	Wed 8/29/07
12	<b>Design</b>	13 days	Thu 9/6/07	Mon 9/24/07
13	Research	6 days	Thu 9/6/07	Thu 9/13/07
14	Designing of Flowchart	2 days	Thu 9/6/07	Fri 9/7/07
15	Designing interface	8 days	Mon 9/10/07	Wed 9/19/07
16	Evaluation	3 days	Thu 9/20/07	Mon 9/24/07
17	<b>Develop Prototype</b>	118 days	Tue 9/25/07	Fri 3/7/08
18	Program the project (Coding)	40 days	Tue 9/25/07	Mon 11/19/07
19	Prepare Progress Report	6 days	Mon 1/21/08	Mon 1/28/08
20	Submission of Progress Report	0 days	Mon 1/28/08	Mon 1/28/08
21	Evaluation	5 days	Mon 1/28/08	Fri 2/1/08
22	Prepare Interim Report	7 days	Tue 2/19/08	Wed 2/27/08
23	Submission of Interim Report	0 days	Fri 3/7/08	Fri 3/7/08
24	<b>Prototype Refinement</b>	11 days	Wed 3/19/08	Wed 4/2/08
25	Evaluate User Comments	3 days	Wed 3/19/08	Fri 3/21/08
26	Project Demo	1 day	Mon 3/24/08	Mon 3/24/08
27	Evaluation	7 days	Tue 3/25/08	Wed 4/2/08
28	<b>Presentation</b>	12 days	Wed 4/9/08	Fri 4/25/08
29	Pre EDX	1 day	Wed 4/9/08	Wed 4/9/08
30	Presentation Preparation	8 days	Mon 4/14/08	Wed 4/23/08
31	Final Presentation	0 days	Fri 4/25/08	Fri 4/25/08

Project: gant chart  
Date: Thu 5/8/08

Task: [ ] Milestone: [◆] External Tasks: [ ] External Milestone: [◆] Split: [ ] Summary: [ ] Progress: [ ] Project Summary: [ ] Deadline: [↑]

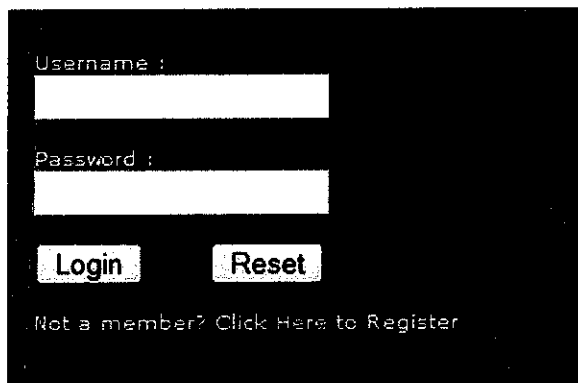
**APPENDIX B:**

**SYSTEM'S GUIDE**

## User Manual

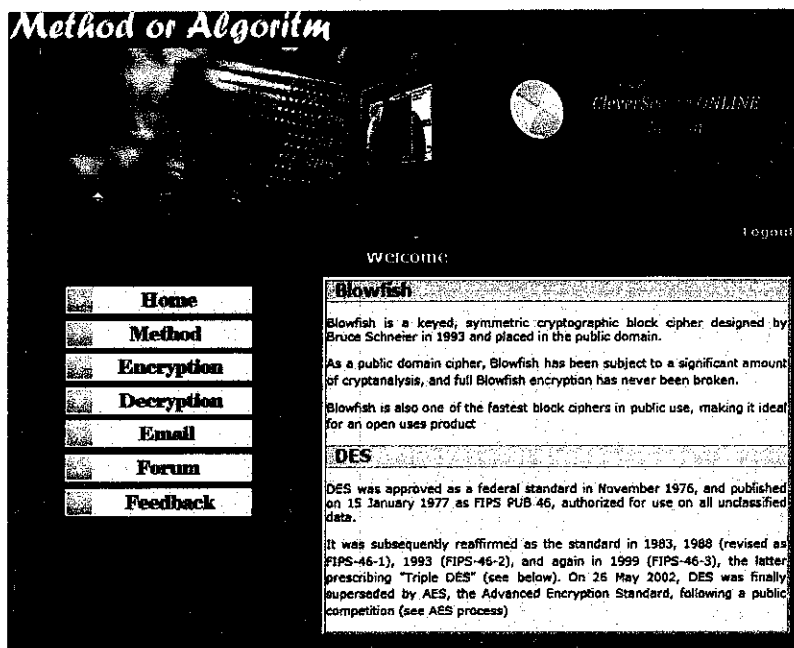
This guide describes how to use the CleverSecure ONLINE System through the internet connection.

1. Registered user need to enter their username and password (Figure Appendix A.1)
2. After that, the system will be redirect to the method function shows in Figure AppendixA.2.
3. Click the button on the left hand side to choose the function available.
4. Click on Encryption or Decryption button to execute the process respectively.  
See Figure AppendixA.3.



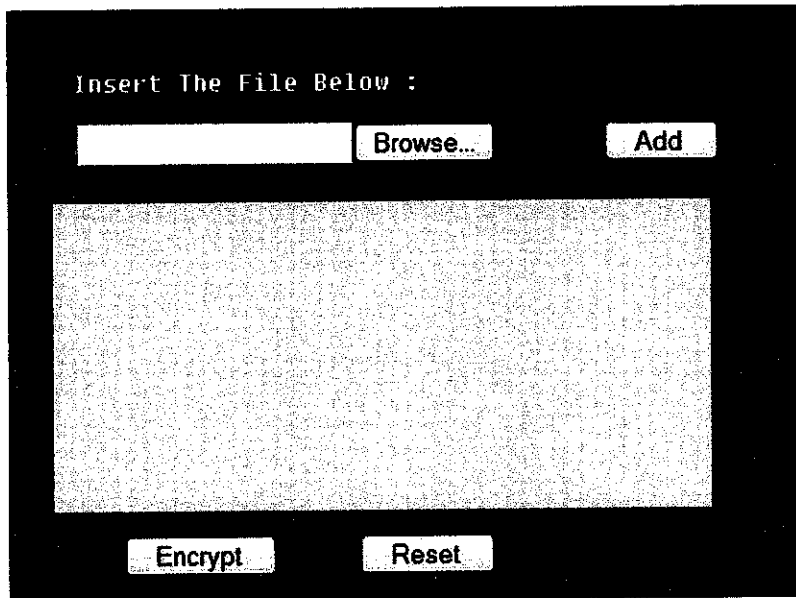
A screenshot of a user login form. It features two input fields: 'Username :' and 'Password :'. Below the password field are two buttons labeled 'Login' and 'Reset'. At the bottom, there is a link that says 'Not a member? Click Here to Register'.

Figure AppendixA.1: User Login



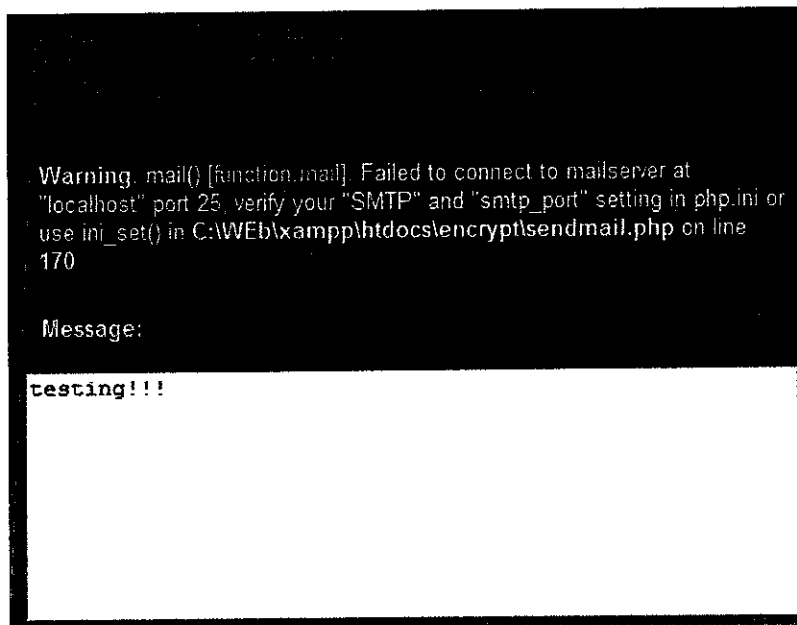
A screenshot of a web page titled 'Method or Algorithm'. The page has a dark background with a grid of buttons on the left side. The buttons are labeled: Home, Method, Encryption, Decryption, Email, Forum, and Feedback. The main content area on the right is titled 'Welcome' and contains information about Blowfish and DES. The Blowfish section describes it as a keyed, symmetric cryptographic block cipher designed by Bruce Schneier in 1993. The DES section describes it as a federal standard approved in November 1976. The page also features a 'Logout' link in the top right corner.

Figure AppendixA.2: Screen Shot of Method Page



AppendixA.3: Screen Shot of Encryption Function

5. Click Email button to send the encrypted file and message to recipient. Figure AppendixA.4 shows the confirmation message to send the email. The system will show the error message if there is no internet connection.



AppendixA.4: Email Confirmation

- Click on Forum to join the discussion room and ask any question regarding to the computer security. Figure AppendixA.5 shows the screen shot of how to post the topic in forum. Click on the “New Topic” to start. Click Reply to reply the question.

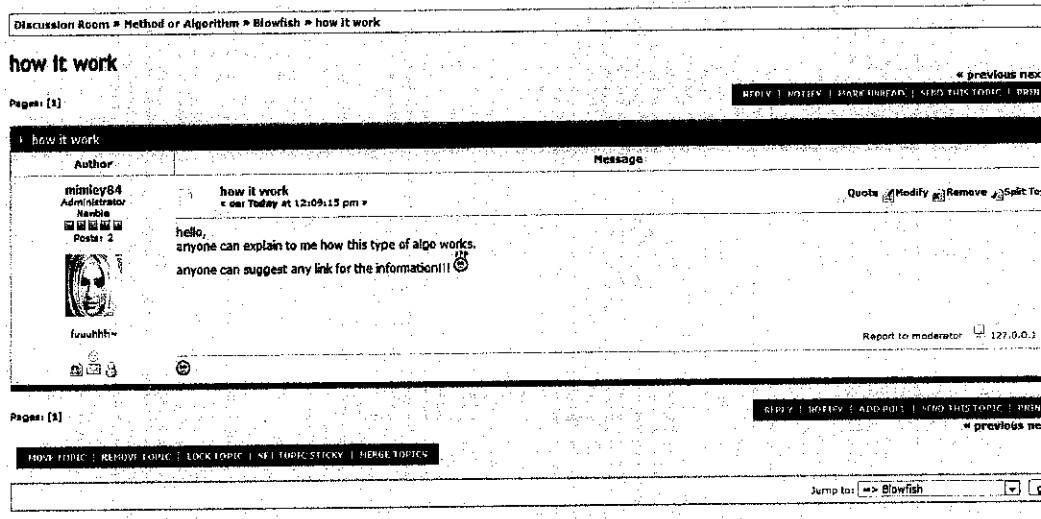


Figure AppendixA.5: How to post the topic

- Click on Feedback to make any complain or suggestion to CleverSecure ONLINE System. Fill the form whether to make a complaint, compliment or suggestion.
- Include the username and email address so because administrator will reply it to your email as soon as possible.
- Tick the checkbox to give the level of performance on this online system. Click Submit to send the form to administrator. See figure AppendixA.6.
- After using the system, click logout to end the session.

Feedback and Complaint Form

Home  COMPLAINT  COMPLIMENT  SUGGESTION

What is your

**B U A**

What would you like us to do about it?

**B U A**

Date	Time	User's Name	Your E-mail
11/04/08	04:29:01	<input type="text"/>	<input type="text"/>

**Please rate our performance (Optional)**  
Please take a moment to rate our service. Your feedback will help us evaluate our performance so that we can better serve you.

Please tick <input type="checkbox"/> one	Excellent	Good	Average	Poor
1. Waiting time for encrypt file	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. User interface design	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure AppendixA.6: Screen Shot of Feedback Function



**APPENDIX C:**

**SYSTEM'S CODE**

## Sending Email:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" >
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-1">
<link href="style.css" rel="stylesheet" type="text/css" />

<style type="text/css">
/*-----Text Styles-----*/
.ws6 {font-size: 8px;}
.ws7 {font-size: 9.3px;}
.ws8 {font-size: 11px;}
.ws9 {font-size: 12px;}
.ws10 {font-size: 13px;}
.ws11 {font-size: 15px;}
.ws12 {font-size: 16px;}
.ws14 {font-size: 19px;}
.ws16 {font-size: 21px;}
.ws18 {font-size: 24px;}
.ws20 {font-size: 27px;}
.ws22 {font-size: 29px;}
.ws24 {font-size: 32px;}
.ws26 {font-size: 35px;}
.ws28 {font-size: 37px;}
.ws36 {font-size: 48px;}
.ws48 {font-size: 64px;}
.ws72 {font-size: 96px;}
.wpmd {font-size: 13px;font-family: 'Arial';font-style: normal;font-weight: normal;}
/*-----Para Styles-----*/
DIV,UL,OL /* Left */
{
margin-top: 0px;
margin-bottom: 0px;
}
</style>

<style type="text/css">
div#container
{
position:relative;
width: 778px;
margin-top: 0px;
margin-left: auto;
margin-right: auto;
text-align:left;
}
body {text-align:center;margin:0}
.style1 {
color: #FFFFFF;
font-weight: bold;
}
</style>
</head>

<script language="JavaScript1.4" type="text/javascript">
<!--
function jsPlay(soundobj) {
var thissound= eval("document."+soundobj);
try {
thissound.Play();
}
catch (e) {
thissound.DoPlay();
}
}
//-->
</script>

<script language="javascript" type="text/javascript">
<!--
function MM_swapImgRestore() {
var i,x,a=document.MM_sr; for(i=0;a&&i<a.length&&(x=a[i])&&x.oSrc;i++) x.src=x.oSrc;
}
-->
```

```

function MM_preloadImages() {
  var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
  var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0; i<a.length; i++)
  if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image; d.MM_p[j++].src=a[i];}}
}

function MM_findObj(n, d) {
  var p,i,x;  if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
  d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
  if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
  for(i=0;!x&&d.layers&&i<d.layers.length;i++) x=MM_findObj(n,d.layers[i].document);
  if(!x && d.getElementById) x=d.getElementById(n); return x;
}

function MM_swapImage() {
  var i,j=0,x,a=MM_swapImage.arguments; document.MM_sr=new Array; for(i=0;i<(a.length-2);i+=3)
  if ((x=MM_findObj(a[i]))!=null){document.MM_sr[j++]=x; if(!x.oSrc) x.oSrc=x.src; x.src=a[i+2];}
}

!-->
</script>

<body>
<div id="wrapper">
      <div id="wrapper-i">
        <div id="header">
          <h1><a href="http://www.freewebsitetemplates.com">
            </a></h1>
          <div id="b-nav">
            >
          </div>
          <!-- end b-nav -->
          <div id="header-cap"></div>
        </div><!-- end header -->
        <div id="spacer"></div> <!-- end spacer -->
        <div id="body">
          <div id="body-i"><p><p><p>
            <form name="form1">
              <div id="text3" style="position:absolute; overflow:hidden; left:317px; top:262px;
width:435px; height:158px; z-index:7">
                <div class="wpmd">
                  <div>
                    <?php
//get data nedeed !
$from = $_POST[from];
$to = $_POST[to];
$subject = $_POST[subject];
$message = $_POST[message];
//replace \n with <br>
$message = str_replace("\n", "<br>",$message);
//report
echo "<b><font color=#8080FF> From: $from </b><br>";
echo "<b>To: $to </b><br>";
echo "<b>Subject: $subject</b><br><br></font>";
// Obtain file upload variables
$fileatt = $_FILES['fileatt']['tmp_name'];
$fileatt_type = $_FILES['fileatt']['type'];
$fileatt_name = $_FILES['fileatt']['name'];

$headers = "From: $from \n";

// if($_FILES['fileatt']['size'] > 0)
if (is_uploaded_file($fileatt)) {
// Read the file to be attached ('rb' = read binary)
$file = fopen($fileatt,'rb');
$data = fread($file,filesize($fileatt));
fclose($file);
// Generate a boundary string
$semi_rand = md5(time());
$mime_boundary = "====Multipart_Boundary_x{$semi_rand}x";
// Add the headers for a file attachment
$headers .= "MIME-Version: 1.0\n" .
"Content-Type: multipart/mixed;\n" .

```

```

" boundary=\{"$mime_boundary}\";
// Add a multipart boundary above the message
$message = "This is a multi-part message in MIME format.\n\n" .
"--{\$mime_boundary}\n" .
"Content-Type: text/html; charset=\{iso-8859-1}\n" .
"Content-Transfer-Encoding: 7bit\n\n" .
$message . "\n\n";

// Base64 encode the file data
$data = chunk_split(base64_encode($data));
// Add file attachment to the message
$message .= "--{\$mime_boundary}\n" .
"Content-Type: {\$fileatt_type};\n" .
" name=\{"$fileatt_name}\n" .
//"Content-Disposition: attachment;\n" .
//" filename=\{"$fileatt_name}\n" .
"Content-Transfer-Encoding: base64\n\n" .
$data . "\n\n" .
"--{\$mime_boundary}\n";
} else echo "File error! ";

//send the mail
if(mail($to, $subject, $message,$headers))echo "<b><font color=#FF0000>Message was send!<b></font>";
else echo "<b><font color=#FF0000>Message error!<b></font>";

```

?>

```

</div></div>
</div>
<p>
<textarea name="textarea1"
style="position:absolute;width:448px;height:142px;left:309px;top:460px;z-index:9" readonly><?=$_POST[message]>
</textarea>
</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
</form>

<div id="nav1" style="position:absolute; left:68px; top:258px; z-index:6; width: 180px;
height: 26px;"><a onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('nav10','images/menu_1i.gif',1)" href="index.php"></a></div>
<div id="nav1" style="position:absolute; left:68px; top:448px; z-index:6; width: 180px;
height: 26px;"><a onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('nav100','images/menu_7i.gif',1)" href="feedback.php"></a></div>
<div id="nav1" style="position:absolute; left:68px; top:416px; z-index:6; width: 180px;
height: 26px;"><a onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('nav200','images/menu_2i.gif',1)" href="forum/index.php"></a></div>
<div id="nav1" style="position:absolute; left:68px; top:290px; z-index:6; width:
180px; height: 26px;"><a onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('nav300','images/menu_3i.gif',1)" href="algorithm.php"></a></div>
<div id="nav1" style="position:absolute; left:68px; top:321px; z-index:6; width:
180px; height: 26px;"><a onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('nav400','images/menu_4i.gif',1)" href="encrypt.php"></a></div>
<div id="nav1" style="position:absolute; left:68px; top:353px; z-index:6; width:
height: 26px;"><a onMouseOut="MM_swapImgRestore()"
onMouseOver="MM_swapImage('nav500','images/menu_5i.gif',1)" href=""></a></div>
<div id="nav1" style="position:absolute; left:68px; top:384px; z-index:6; width:
180px; height: 26px;"><a onMouseOut="MM_swapImgRestore()"

```

```
onMouseOver="MM_swapImage('nav600','images/menu_6i.gif,1)" href="email.php"></a></div>
```

```
<div id="text4" style="position:absolute; overflow:hidden; left:318px; top:428px;
width:63px; height:18px; z-index:10">
```

```
<div class="wpmd">
<div class="style1">Message:</div>
</div>
</div>
```

```
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
```

```
</div>
```

```
</div>
```

```
</div><!--end wrapper-i-->
```

```
</div><!--end of wrapper-->
```

```
</body>
```

```
</html>
```

## Encryption and Decryption Process:

<?php

```
class Horde_Cipher_blowfish {

    /* Pi Array */
    public $p = array(
        0x243F6A88, 0x85A308D3, 0x13198A2E, 0x03707344,
        0xA4093822, 0x299F31D0, 0x082EFA98, 0xEC4E6C89,
        0x452821E6, 0x38D01377, 0xBE5466CF, 0x34E90C6C,
        0xC0AC29B7, 0xC97C50DD, 0x3F84D5B5, 0xB5470917,
        0x9216D5D9, 0x8979FB1B);

    /* S Boxes */
    public $s1 = array(
        0xD1310BA6, 0x98DFB5AC, 0x2FFD72DB, 0xD01ADFB7,
        0xB8E1AFED, 0x6A267E96, 0xBA7C9045, 0xF12C7F99,
        0x24A19947, 0xB3916CF7, 0x0801F2E2, 0x858EFC16,
        0x636920D8, 0x71574E69, 0xA458FEA3, 0xF4933D7E,
        0x0D955748F, 0x728EB658, 0x718BCD58, 0x8215AEE,
        0x7B54A41D, 0xC25A59B5, 0x9C30D539, 0x2AF26013,
        0xC5D1B023, 0x286085F0, 0xCA417918, 0xB8DB38EF,
        0x8E79DCB0, 0x603A180E, 0x6C9E0E8B, 0xB01E8A3E,
        0xD71577C1, 0xBD314B27, 0x78AF2FDA, 0x55605C60,
        0xE65525F3, 0xAA55AB94, 0x57489862, 0x63E81440,
        0x55CA396A, 0x2AAB10B6, 0xB4CC5C34, 0x1141E8CE,
        0xA15486AF, 0x7C72E993, 0xB3EE1411, 0x636FB2CA,
        0x2BA9C55D, 0x741831F6, 0xCE5C3E16, 0x9B87931E,
        0xAFD6BA33, 0x6C24CF5C, 0x7A325381, 0x28958677,
        0x3B8F4898, 0x6B4BB9AF, 0xC4BFE81B, 0x66282193,
        0x61D809CC, 0xFB21A991, 0x487CAC60, 0x5DEC8032,
        0xEF845D5D, 0xE98575B1, 0xDC262302, 0xEB651B88,
        0x23893E81, 0xD396ACC5, 0x0F6D6FF3, 0x83F44239,
        0x2E0B4482, 0xA4842004, 0x69C8F04A, 0x9E1F9B5E,
        0x21C66842, 0xF6E96C9A, 0x670C9C61, 0xABD388F0,
        0x6A51A0D2, 0xD8542F68, 0x960FA728, 0xAB5133A3,
        0x6EEF0B6C, 0x137A3BE4, 0xBA3BF050, 0x7EFB2A98,
        0xA1F1651D, 0x39AF0176, 0x66CA593E, 0x82430E88,
        0x8CEE8619, 0x456F9FB4, 0x7D84A5C3, 0x3B8B5EBE,
        0xE06F75D8, 0x85C12073, 0x401A449F, 0x56C16AA6,
        0x4ED3AA62, 0x363F7706, 0x1BFEDF72, 0x429B023D,
        0x37D0D724, 0xD00A1248, 0xDB0FEAD3, 0x49F1C09B,
        0x075372C9, 0x80991B7B, 0x25D479D8, 0xF6E8DEF7,
        0xE3FE501A, 0xB6794C3B, 0x976CE0BD, 0x04C006BA,
        0xC1A94FB6, 0x409F60C4, 0x5E5C9EC2, 0x196A2463,
        0x68FB6FAF, 0x3E6C53B5, 0x1339B2EB, 0x3B52EC6F,
        0x6DFC511F, 0x9B30952C, 0xCC814544, 0xAF5EBD09,
        0xBEE3D004, 0xDE334AFD, 0x660F2807, 0x192E4BB3,
        0xCOCBA857, 0x45C8740F, 0xD20B5F39, 0xB9D3FBDB,
        0x5579C0BD, 0x1A60320A, 0xD6A100C6, 0x402C7279,
        0x679F25FE, 0xFB1FA3CC, 0x8EA5E9F8, 0xDB3222F8,
        0x3C7516DF, 0xFD616B15, 0x2F501EC8, 0xAD0552AB,
        0x323DB5FA, 0xFD238760, 0x53317B48, 0x3E00DF82,
        0x9E5C57BB, 0xCA6F8CA0, 0x1A87562E, 0xDF1769DB,
        0xD542A8F6, 0x287E9FC3, 0xAC6732C6, 0x8C4F5573,
        0x695B27B0, 0xBBCA58C8, 0xE1FFA35D, 0xB8F011A0,
        0x10FA3D98, 0xFD2183B8, 0x4AFCB56C, 0x2DD1D35B,
        0x9A53E479, 0xB6F84565, 0xD28E49BC, 0x4BFB9790,
        0xE1DDF2DA, 0xA4CB7E33, 0x62FB1341, 0xCEE4C6E8,
        0xEF20CADA, 0x36774C01, 0xD07E9EFE, 0x2BF11FB4,
        0x95DBDA4D, 0xAE909198, 0xEAAD8E71, 0x6B93D5A0,
        0xD08ED1D0, 0xAFC725E0, 0x8E3C5B2F, 0x8E7594B7,
        0x8FF6E2FB, 0xF2122B64, 0x8888B812, 0x900DF01C,
        0x4FAD5EA0, 0x688FC31C, 0xD1CFF191, 0xB3A8C1AD,
        0x2F2F2218, 0xBE0E1777, 0xEA752DFE, 0x8B021FA1,
        0xE5A0CC0F, 0xB56F74E8, 0x18ACF3D6, 0xCE89E299,
        0xB4A84FE0, 0xFD13E0B7, 0x7CC43B81, 0xD2ADA8D9,
        0x165FA266, 0x80957705, 0x93CC7314, 0x211A1477,
        0xE6AD2065, 0x77B5FA86, 0xC75442F5, 0xFB9D35CF,
        0xEBCDAF0C, 0x7B3E89A0, 0xD6411BD3, 0xAE1E7E49,
        0x00250E2D, 0x2071B35E, 0x226800BB, 0x57B8E0AF,
        0x2464369B, 0xF009B91E, 0x5563911D, 0x59DFA6AA,
        0x78C14389, 0xD95A537F, 0x207D5BA2, 0x02E5B9C5,
        0x83260376, 0x6295CFA9, 0x11C81968, 0x4E734A41,
        0xB3472DCA, 0x7B14A94A, 0x1B510052, 0x9A532915,
```

0xD60F573F, 0xBC9BC6E4, 0x2B60A476, 0x81E67400,  
0x08BA6FB5, 0x571BE91F, 0xF296EC6B, 0x2A0DD915,  
0xB6636521, 0xE7B9F9B6, 0xFF34052E, 0xC5855664,  
0x53B02D5D, 0xA99F8FA1, 0x08BA4799, 0x6E85076A);

```
public $s2 = array(  
0x4B7A70E9, 0xB5B32944, 0xDB75092E, 0xC4192623,  
0xAD6EA6B0, 0x49A7DF7D, 0x9CEE60B8, 0x8FEDB266,  
0xECAA8C71, 0x699A17FF, 0x5664526C, 0xC2B19EE1,  
0x193602A5, 0x75094C29, 0xA0591340, 0xE4183A3E,  
0x3F54989A, 0x5B429D65, 0x6B8FE4D6, 0x99F73FD6,  
0xA1D29C07, 0xEFE830F5, 0x4D2D38E6, 0xF0255DC1,  
0x4CDD2086, 0x8470EB26, 0x6382E9C6, 0x021ECC5E,  
0x09686B3F, 0x3EBAEFC9, 0x3C971814, 0x6B6A70A1,  
0x687F3584, 0x52A0E286, 0xB79C5305, 0xAA500737,  
0x3E07841C, 0x7FDEA5C, 0x8E7D44EC, 0x5716F2B8,  
0xB03ADA37, 0xF0500C0D, 0xF01C1F04, 0x0200B3FF,  
0xAE0CF51A, 0x3CB574B2, 0x25837A58, 0xDC0921BD,  
0xD19113F9, 0x7CA92FF6, 0x94324773, 0x22F54701,  
0x3AE5E581, 0x37C2DADC, 0xC8B57634, 0x9AF3DDA7,  
0xA9446146, 0x0FD0030E, 0xECC8C73E, 0xA4751E41,  
0xE238CD99, 0x3BEA0E2F, 0x3280BBA1, 0x183EB331,  
0x4E548B38, 0x4F6DB908, 0x6F420D03, 0xF60A04BF,  
0x2CB81290, 0x24977C79, 0x5679B072, 0xBCAF89AF,  
0xDE9A771F, 0xD9930810, 0xB38BAE12, 0xDCCF3F2E,  
0x5512721F, 0x2E6B7124, 0x501ADDE6, 0x9F84CD87,  
0x7A584718, 0x7408DA17, 0xBC9F9ABC, 0xE94B7D8C,  
0xEC7AEC3A, 0xDB851DFA, 0x63094366, 0xC464C3D2,  
0xEF1C1847, 0x3215D908, 0xDD433B37, 0x24C2BA16,  
0x12A14D43, 0x2A65C451, 0x50940002, 0x133AE4DD,  
0x71DDF89E, 0x10314E55, 0x81AC77D6, 0x5F11199B,  
0x043556F1, 0xD7A3C76B, 0x3C11183B, 0x5924A509,  
0xF28FE6ED, 0x97F1FBFA, 0x9EBABF2C, 0x1E153C6E,  
0x86E34570, 0xEAE96FB1, 0x860E5E0A, 0x5A3E2AB3,  
0x771FE71C, 0x4E3D06FA, 0x2965DCB9, 0x99E71D0F,  
0x803E89D6, 0x5266C825, 0x2E4CC978, 0x9C10B36A,  
0xC6150EBA, 0x94E2EA78, 0xA5FC3C53, 0x1E0A2DF4,  
0xF2F74EA7, 0x361D2B3D, 0x1939260F, 0x19C27960,  
0x5223A708, 0xF71312B6, 0xEBADFE6E, 0xEAC31F66,  
0xE3BC4595, 0xA67BC883, 0xB17F37D1, 0x018CFF28,  
0xC332DDEF, 0xBE6C5AA5, 0x65582185, 0x68AB9802,  
0xEECEA50F, 0xDB2F953B, 0x2AEF7DAD, 0x5B6E2F84,  
0x1521B628, 0x29076170, 0xECCDD475, 0x619F1510,  
0x13CCA830, 0xEB61BD96, 0x0334FE1E, 0xAA0363CF,  
0xB5735C90, 0x4C70A239, 0xD59E9E0B, 0xCBAADE14,  
0xEECC86BC, 0x60622CA7, 0x9CAB5CAB, 0xB2F3846E,  
0x648B1EAF, 0x19BDF0CA, 0xA02369B9, 0x655ABB50,  
0x40685A32, 0x3C2AB4B3, 0x319EE9D5, 0xC021B8F7,  
0x9B540B19, 0x875FA099, 0x95F7997E, 0x623D7DA8,  
0xF837889A, 0x97E32D77, 0x11ED935F, 0x16681281,  
0x0E358829, 0xC7E61FD6, 0x96DEDAF1, 0x7858BA99,  
0x57F584A5, 0x1B227263, 0x9B83C3FF, 0x1AC24696,  
0xCDB30AEB, 0x532E3054, 0x8FD948E4, 0x6DBC3128,  
0x58EBF2EF, 0x34C6FFEA, 0xFE28ED61, 0xEE7C3C73,  
0x5D4A14D9, 0xE864B7E3, 0x42105D14, 0x203E13E0,  
0x45EEE2B6, 0xA3AAABEA, 0xDB6C4F15, 0xFACB4FD0,  
0xC742F442, 0xEF6ABBB5, 0x654F3B1D, 0x41CD2105,  
0xD81E799E, 0x86854DC7, 0xE44B476A, 0x3D816250,  
0xCF62A1F2, 0x5B8D2646, 0xFC8883A0, 0xC1C7B6A3,  
0x7F1524C3, 0x69CB7492, 0x47848A0B, 0x5692B285,  
0x095BBF00, 0xAD19489D, 0x1462B174, 0x23820E00,  
0x58428D2A, 0x0C55F5EA, 0x1DADF43E, 0x233F7061,  
0x3372F092, 0x8D937E41, 0xD65FEFCF1, 0x6C223BDB,  
0x7CDE3759, 0xCBBE7460, 0x4085F2A7, 0xCE77326E,  
0xA6078084, 0x19F8509E, 0xE8EFD855, 0x61D99735,  
0xA969A7AA, 0xC50C06C2, 0x5A04ABFC, 0x800BCADC,  
0x9E447A2E, 0xC3453484, 0xFDD56705, 0x0E1E9EC9,  
0xDB73DBD3, 0x105588CD, 0x675FDA79, 0xE3674340,  
0xC5C43465, 0x713E38D8, 0x3D28F89E, 0xF16DFF20,  
0x153E21E7, 0x8FB03D4A, 0xE6E39F2B, 0xDB83ADF7);
```

```
public $s3 = array(  
0xE93D5A68, 0x948140F7, 0xF64C261C, 0x94692934,  
0x411520F7, 0x7602D4F7, 0xBCF46B2E, 0xD4A20068,  
0xD4082471, 0x3320F46A, 0x43B7D4B7, 0x500061AF,  
0x1E39F62E, 0x97244546, 0x14214F74, 0xBF8B8840,
```

0x4D95FC1D, 0x96B591AF, 0x70F4DDD3, 0x66A02F45,  
0xBFBC09EC, 0x03BD9785, 0x7FAC6DD0, 0x31CB8504,  
0x96EB27B3, 0x55FD3941, 0xDA2547E6, 0xABCA0A9A,  
0x28507825, 0x530429F4, 0x0A2C86DA, 0xE9B66DFB,  
0x68DC1462, 0xD7486900, 0x680EC0A4, 0x27A18DEE,  
0x4F3FFE2, 0xE887AD8C, 0xB58CE006, 0x7AF4D6B6,  
0xAACE1E7C, 0xD3375FEC, 0xCE78A399, 0x406B2A42,  
0x20FE9E35, 0xD9F385B9, 0xEE39D7AB, 0x3B124E8B,  
0x1DC9FAF7, 0x4B6D1856, 0x26A36631, 0xEAE397B2,  
0x3A6EFA74, 0xDD5B4332, 0x6841E7F7, 0xCA7820FB,  
0xFB0AF54E, 0xD8FEB397, 0x454056AC, 0xBA489527,  
0x55533A3A, 0x20838D87, 0xFE6BA9B7, 0xD096954B,  
0x55A867BC, 0xA1159A58, 0xCCA92963, 0x99E1DB33,  
0xA62A4A56, 0x3F3125F9, 0x5EF47E1C, 0x9029317C,  
0xFDF8E802, 0x04272F70, 0x80BB155C, 0x05282CE3,  
0x95C11548, 0xE4C66D22, 0x48C1133F, 0xC70F86DC,  
0x07F9C9EE, 0x41041F0F, 0x404779A4, 0x5D886E17,  
0x325F51EB, 0xD59BC0D1, 0xF2BCC18F, 0x41113564,  
0x257B7834, 0x602A9C60, 0xDFF8E8A3, 0x1F636C1B,  
0x0E12B4C2, 0x02E1329E, 0xAF664FD1, 0xCAD18115,  
0x6B2395E0, 0x333E92E1, 0x3B240B62, 0xEEBEB922,  
0x85B2A20E, 0xE6BA0D99, 0xDE720C8C, 0x2DA2F728,  
0xD0127845, 0x95B794FD, 0x647D0862, 0xE7CCF5F0,  
0x5449A36F, 0x877D48FA, 0xC39DFD27, 0xF33E8D1E,  
0x0A476341, 0x992EFF74, 0x3A6F6EAB, 0xF4F8FD37,  
0xA812DC60, 0xA1EBDDF8, 0x991BE14C, 0xDB6E6B0D,  
0xC67B5510, 0x6D672C37, 0x2765D43B, 0xDCD0E804,  
0xF1290DC7, 0xCC00FFA3, 0xB5390F92, 0x690FED0B,  
0x667B9FFB, 0xCEDB7D9C, 0xA091CF0B, 0xD9155EA3,  
0xBB132F88, 0x515BAD24, 0x7B9479BF, 0x763BD6EB,  
0x37392EB3, 0xCC115979, 0x8026E297, 0xF42E312D,  
0x6842ADA7, 0xC66A2B3B, 0x12754CCC, 0x782EF11C,  
0x6A124237, 0xB79251E7, 0x06A1BBE6, 0x4BFB6350,  
0x1A6B1018, 0x11CAEDFA, 0x3D25BDD8, 0xE2E1C3C9,  
0x44421659, 0x0A121386, 0xD90CEC6E, 0xD5ABEA2A,  
0x64AF674E, 0xDA86A85F, 0xBEBFE988, 0x64E4C3FE,  
0x9DBC8057, 0xF0F7C086, 0x60787BF8, 0x6003604D,  
0xD1FD8346, 0xF6381FB0, 0x7745AE04, 0xD73FCCC,  
0x83426B33, 0xF01EAB71, 0xB0804187, 0x3C005E5F,  
0x77A057BE, 0xBDE8AE24, 0x55464299, 0xBF582E61,  
0x4E58F48F, 0xF2DDFDA2, 0xF474EF38, 0x8789BDC2,  
0x5366F9C3, 0xC8B38E74, 0xB475F255, 0x46FCD9B9,  
0x7AEB2661, 0x8B1DDDF84, 0x846A0E79, 0x915F95E2,  
0x466E598E, 0x20B45770, 0x8CD55591, 0xC902DE4C,  
0xB90BACE1, 0xBB8205D0, 0x11A86248, 0x7574A99E,  
0xB77F19B6, 0xE0A9DC09, 0x662D09A1, 0xC4324633,  
0xE85A1F02, 0x09F0BE8C, 0x4A99A025, 0x1D6EFE10,  
0x1AB93D1D, 0x0BA5A4DF, 0xA186F20F, 0x2868F169,  
0xDCB7DA83, 0x573906FE, 0xA1E2CE9B, 0x4FCD7F52,  
0x50115E01, 0xA70683FA, 0xA002B5C4, 0x0DE6D027,  
0x9AF88C27, 0x773F8641, 0xC3604C06, 0x61A806B5,  
0xF0177A28, 0xC0F586E0, 0x006058AA, 0x30DC7D62,  
0x11E69ED7, 0x2338EA63, 0x53C2DD94, 0xC2C21634,  
0xBBCBEE56, 0x90BCB6DE, 0xEBFC7DA1, 0xCE591D76,  
0x6F05E409, 0x4B7C0188, 0x39720A3D, 0x7C927C24,  
0x86E3725F, 0x724D9DB9, 0x1AC15BB4, 0xD39EB8FC,  
0xED545578, 0x08FCA5B5, 0xD83D7CD3, 0x4DAD0FC4,  
0x1E50EF5E, 0xB161E6F8, 0xA28514D9, 0x6C51133C,  
0x6FD5C7E7, 0x56E14EC4, 0x362ABFCE, 0xDDC6C837,  
0xD79A3234, 0x92638212, 0x670EFA8E, 0x406000E0);

public \$s4 = array(

0x3A39CE37, 0xD3FAF5CF, 0xABC27737, 0x5AC52D1B,  
0x5CB0679E, 0x4FA33742, 0xD3822740, 0x99BC9BBE,  
0xD5118E9D, 0xBF0F7315, 0xD62D1C7E, 0xC700C47B,  
0xB78C1B6B, 0x21A19045, 0xB26EB1BE, 0x6A366EB4,  
0x5748AB2F, 0xBC946E79, 0xC6A376D2, 0x6549C2C8,  
0x530FF8EE, 0x468DDE7D, 0xD5730A1D, 0x4CD04DC6,  
0x2939BBDB, 0xA9BA4650, 0xAC9526E8, 0xBE5EE304,  
0xA1FAD5F0, 0x6A2D519A, 0x63EF8CE2, 0x9A86EE22,  
0xC089C2B8, 0x43242EF6, 0xA51E03AA, 0x9CF2D0A4,  
0x83C061BA, 0x9BE96A4D, 0x8FE51550, 0xBA645BD6,  
0x2826A2F9, 0xA73A3AE1, 0x4BA99586, 0xEF5562E9,  
0xC72FEFD3, 0xF752F7DA, 0x3F046F69, 0x77FA0A59,  
0x80E4A915, 0x87B08601, 0x9B09E6AD, 0x3B3EE593,  
0xE990FD5A, 0x9E34D797, 0x2CF0B7D9, 0x022B8B51,



```

0x96D5AC3A, 0x017DA67D, 0xD1CF3ED6, 0x7C7D2D28,
0x1F9F25CF, 0xADF2B89B, 0x5AD6B472, 0x5A88F54C,
0xE029AC71, 0xE019A5E6, 0x47B0ACFD, 0xED93FA9B,
0xE8D3C48D, 0x283B57CC, 0xF8D56629, 0x79132E28,
0x785F0191, 0xED756055, 0xF7960E44, 0xE3D35E8C,
0x15056DD4, 0x88F46DBA, 0x03A16125, 0x0564F0BD,
0xC3EB9E15, 0x3C9057A2, 0x97271AEC, 0xA93A072A,
0x1B3F6D9B, 0x1E6321F5, 0xF59C66FB, 0x26DCF319,
0x7533D928, 0xB155FDF5, 0x03563482, 0x8ABA3CBB,
0x28517711, 0xC20AD9F8, 0xABCC5167, 0xCCAD925F,
0x4DE81751, 0x3830DC8E, 0x379D5862, 0x9320F991,
0xEA7A90C2, 0xFB3E7BCE, 0x5121CE64, 0x774FBE32,
0xA8B6E37E, 0xC3293D46, 0x48DE5369, 0x6413E680,
0xA2AE0810, 0xDD6DB224, 0x69852DFD, 0x09072166,
0xB39A460A, 0x6445C0DD, 0x586CDECF, 0x1C20C8AE,
0x5BBEF7DD, 0x1B588D40, 0xCCD2017F, 0x6BB4E3BB,
0xDDA26A7E, 0x3A59FF45, 0x3E350A44, 0xBCB4CDD5,
0x72EACEA8, 0xFA6484BB, 0x8D6612AE, 0xBF3C6F47,
0xD29BE463, 0x542F5D9E, 0xAEC2771B, 0xF64E6370,
0x740E0D8D, 0xE75B1357, 0xF8721671, 0xAF537D5D,
0x4040CB08, 0x4EB4E2CC, 0x34D2466A, 0x0115AF84,
0xE1B00428, 0x95983A1D, 0x06B89FB4, 0xCE6EA048,
0x6F3F3B82, 0x3520AB82, 0x011A1D4B, 0x277227F8,
0x611560B1, 0xE7933FDC, 0xBB3A792B, 0x344525BD,
0xA08839E1, 0x51CE794B, 0x2F32C9B7, 0xA01FBAC9,
0xE01CC87E, 0xBCC7D1F6, 0xCF0111C3, 0xA1E8AAC7,
0x1A908749, 0xD44FBD9A, 0xD0DADECB, 0xD50ADA38,
0x0339C32A, 0xC6913667, 0x8DF9317C, 0xE0B12B4F,
0xF79E59B7, 0x43F5BB3A, 0xF2D519FF, 0x27D9459C,
0xBF97222C, 0x15E6FC2A, 0x0F91FC71, 0x9B941525,
0xFAE59361, 0xCEB69CEB, 0xC2A86459, 0x12BAA8D1,
0xB6C1075E, 0xE3056A0C, 0x10D25065, 0xCB03A442,
0xE0EC6E0E, 0x1698DB3B, 0x4C98A0BE, 0x3278E964,
0x9F1F9532, 0xE0D392DF, 0xD3A0342B, 0x8971F21E,
0x1B0A7441, 0x4BA3348C, 0xC5BE7120, 0xC37632D8,
0xDF359F8D, 0x9B992F2E, 0xE60B6F47, 0x0FE3F11D,
0xE54CDA54, 0x1EDAD891, 0xCE6279CF, 0xCD3E7E6F,
0x1618B166, 0xFD2C1D05, 0x848FD2C5, 0xF6FB2299,
0xF523F357, 0xA6327623, 0x93A83531, 0x56CCCD02,
0xACF08162, 0x5A75EBB5, 0x6E163697, 0x88D273CC,
0xDE966292, 0x81B949D0, 0x4C50901B, 0x71C65614,
0xE6C6C7BD, 0x327A140A, 0x45E1D006, 0xC3F27B9A,
0xC9AA53FD, 0x62A80F00, 0xBB25BFE2, 0x35BDD2F6,
0x71126905, 0xB2040222, 0xB6CBCF7C, 0xCD769C2B,
0x53113EC0, 0x1640E3D3, 0x38ABBD60, 0x2547ADF0,
0xBA38209C, 0xF746CE76, 0x77AFA1C5, 0x20756060,
0x85CBFE4E, 0x8AE88DD8, 0x7AAAF9B0, 0x4CF9AA7E,
0x1948C25C, 0x02FB8A8C, 0x01C36AE4, 0xD6EBE1F9,
0x90D4F869, 0xA65CDEA0, 0x3F09252D, 0xC208E69F,
0xB74E6132, 0xCE77E25B, 0x578FDFE3, 0x3AC372E6);

```

```
/* The number of rounds to do */
```

```
public $ _rounds = 16;
```

```
/* Constructor */
```

```
function Cipher_blowfish($params = null)
```

```
{
```

```
}
```

```
/**
```

```
* Set the key to be used for en/decryption
```

```
*
```

```
* @param String $key The key to use
```

```
*/
```

```
function setKey($key)
```

```
{
```

```
$key = $this->_formatKey($key);
```

```
$keyPos = $keyXor = 0;
```

```
$iMax = count($this->p);
```

```
$keyLen = count($key);
```

```
for ($i = 0; $i < $iMax; $i++) {
```

```
for ($t = 0; $t < 4; $t++) {
```

```
$keyXor = ($keyXor << 8) | (($key[$keyPos]) & 0xFF);
```

```
if (++$keyPos == $keyLen) {
```

```

$keyPos = 0;
}
}
$this->p[$i] = $this->p[$i] ^ $keyXor;
}

$encZero = array('L' => 0, 'R' => 0);
for ($i = 0; $i + 1 < $iMax; $i += 2) {
$encZero = $this->_encryptBlock($encZero['L'], $encZero['R']);
$this->p[$i] = $encZero['L'];
$this->p[$i + 1] = $encZero['R'];
}

$iMax = count($this->s1);
for ($i = 0; $i < $iMax; $i += 2) {
$encZero = $this->_encryptBlock($encZero['L'], $encZero['R']);
$this->s1[$i] = $encZero['L'];
$this->s1[$i + 1] = $encZero['R'];
}

$iMax = count($this->s2);
for ($i = 0; $i < $iMax; $i += 2) {
$encZero = $this->_encryptBlock($encZero['L'], $encZero['R']);
$this->s2[$i] = $encZero['L'];
$this->s2[$i + 1] = $encZero['R'];
}

$iMax = count($this->s3);
for ($i = 0; $i < $iMax; $i += 2) {
$encZero = $this->_encryptBlock($encZero['L'], $encZero['R']);
$this->s3[$i] = $encZero['L'];
$this->s3[$i + 1] = $encZero['R'];
}

$iMax = count($this->s4);
for ($i = 0; $i < $iMax; $i += 2) {
$encZero = $this->_encryptBlock($encZero['L'], $encZero['R']);
$this->s4[$i] = $encZero['L'];
$this->s4[$i + 1] = $encZero['R'];
}
}
}

```

```

/**
 * Return the size of the blocks that this cipher needs
 *
 * @return Integer The number of characters per block
 */
function getBlockSize()
{
return 8;
}

/**
 * Encrypt a block on data.
 *
 * @param String $block The data to encrypt
 * @param optional String $key The key to use
 *
 * @return String the encrypted output
 */
function encryptBlock($block, $key = null)
{
if (!is_null($key)) {
$this->setKey($key);
}

list($L, $R) = array_values(unpack('N*', $block));
$parts = $this->encryptBlock($L, $R);
return pack('NN', $parts['L'], $parts['R']);
}

/**
 * Encrypt a block on data.
 *
 * @param String $L The data to encrypt.
 * @param String $R The data to encrypt.
 *
 * @return String The encrypted output.
 */
function _encryptBlock($L, $R)
{
$L ^= $this->p[0];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[1]);
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[2];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[3];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[4];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[5];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[6];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[7];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[8];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[9];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[10];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[11];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[12];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[13];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[14];
$R ^= (((($this->s1[(($L >> 24) & 0xFF) + $this->s2[(($L >> 16) & 0xFF)] ^ $this->s3[(($L >> 8) & 0xFF)] + $this->s4[$L & 0xFF]) ^ $this->p[15];
$L ^= (((($this->s1[(($R >> 24) & 0xFF) + $this->s2[(($R >> 16) & 0xFF)] ^ $this->s3[(($R >> 8) & 0xFF)] + $this->s4[$R & 0xFF]) ^ $this->p[16];
$R ^= $this->p[17];
return array('L' => $R, 'R' => $L);
}

```

```

/**
 * Decrypt a block on data.
 *
 * @param String $block    The data to decrypt
 *
 * @param optional String $key The key to use
 *
 * @return String the decrypted output
 */

function decryptBlock($block, $key = null)
{
    if (!is_null($key)) {
        $this->setKey($key);
    }

    $unpack = unpack('N*', $block);

    if (!is_array($unpack))

        pla_error(
            sprintf('BLOWFISH: decryptBlock()<br>We expected unpack to produce an array, but instead it produced [%s]. T
            his function was entered with (%s,%s). If you think that this is a bug, then please tell the PLA developers how you got her
            e. You are using PLA [%s,%s]',

                serialize($unpack),rawurlencode($block),$key,pla_version(),phpversion()));

    list($L, $R) = array_values($unpack);

    $L ^= $this->p[17];

    $R ^= (((($this->s1[($L >> 24) & 0xFF] + $this->s2[($L >> 16) & 0xFF]) ^ $this->s3[($L >> 8) & 0xFF]) + $this-
    >s4[$L & 0xFF]) ^ $this->p[16];

    $L ^= (((($this->s1[($R >> 24) & 0xFF] + $this->s2[($R >> 16) & 0xFF]) ^ $this->s3[($R >> 8) & 0xFF]) + $this-
    >s4[$R & 0xFF]) ^ $this->p[15];

    $R ^= (((($this->s1[($L >> 24) & 0xFF] + $this->s2[($L >> 16) & 0xFF]) ^ $this->s3[($L >> 8) & 0xFF]) + $this-
    >s4[$L & 0xFF]) ^ $this->p[14];

    $L ^= (((($this->s1[($R >> 24) & 0xFF] + $this->s2[($R >> 16) & 0xFF]) ^ $this->s3[($R >> 8) & 0xFF]) + $this-
    >s4[$R & 0xFF]) ^ $this->p[13];

    $R ^= (((($this->s1[($L >> 24) & 0xFF] + $this->s2[($L >> 16) & 0xFF]) ^ $this->s3[($L >> 8) & 0xFF]) + $this-
    >s4[$L & 0xFF]) ^ $this->p[12];

    $L ^= (((($this->s1[($R >> 24) & 0xFF] + $this->s2[($R >> 16) & 0xFF]) ^ $this->s3[($R >> 8) & 0xFF]) + $this-
    >s4[$R & 0xFF]) ^ $this->p[11];

```

```
$R ^= ((($this->s1[( $L >> 24) & 0xFF] + $this->s2[( $L >> 16) & 0xFF]) ^ $this->s3[( $L >> 8) & 0xFF]) + $this->s4[( $L & 0xFF]) ^ $this->p[10];
```

```
$L ^= ((($this->s1[( $R >> 24) & 0xFF] + $this->s2[( $R >> 16) & 0xFF]) ^ $this->s3[( $R >> 8) & 0xFF]) + $this->s4[( $R & 0xFF]) ^ $this->p[9];
```

```
$R ^= ((($this->s1[( $L >> 24) & 0xFF] + $this->s2[( $L >> 16) & 0xFF]) ^ $this->s3[( $L >> 8) & 0xFF]) + $this->s4[( $L & 0xFF]) ^ $this->p[8];
```

```
$L ^= ((($this->s1[( $R >> 24) & 0xFF] + $this->s2[( $R >> 16) & 0xFF]) ^ $this->s3[( $R >> 8) & 0xFF]) + $this->s4[( $R & 0xFF]) ^ $this->p[7];
```

```
$R ^= ((($this->s1[( $L >> 24) & 0xFF] + $this->s2[( $L >> 16) & 0xFF]) ^ $this->s3[( $L >> 8) & 0xFF]) + $this->s4[( $L & 0xFF]) ^ $this->p[6];
```

```
$L ^= ((($this->s1[( $R >> 24) & 0xFF] + $this->s2[( $R >> 16) & 0xFF]) ^ $this->s3[( $R >> 8) & 0xFF]) + $this->s4[( $R & 0xFF]) ^ $this->p[5];
```

```
$R ^= ((($this->s1[( $L >> 24) & 0xFF] + $this->s2[( $L >> 16) & 0xFF]) ^ $this->s3[( $L >> 8) & 0xFF]) + $this->s4[( $L & 0xFF]) ^ $this->p[4];
```

```
$L ^= ((($this->s1[( $R >> 24) & 0xFF] + $this->s2[( $R >> 16) & 0xFF]) ^ $this->s3[( $R >> 8) & 0xFF]) + $this->s4[( $R & 0xFF]) ^ $this->p[3];
```

```
$R ^= ((($this->s1[( $L >> 24) & 0xFF] + $this->s2[( $L >> 16) & 0xFF]) ^ $this->s3[( $L >> 8) & 0xFF]) + $this->s4[( $L & 0xFF]) ^ $this->p[2];
```

```
$L ^= ((($this->s1[( $R >> 24) & 0xFF] + $this->s2[( $R >> 16) & 0xFF]) ^ $this->s3[( $R >> 8) & 0xFF]) + $this->s4[( $R & 0xFF]) ^ $this->p[1];
```

```
$decrypted = pack("NN", $R ^ $this->p[0], $L);
```

```
return $decrypted;
```

```
}
```

```
/**
```

```
* Converts a text key into an array.
```

```
* @return array The key.
```

```
*/
```

```
function formatKey($key)
```

```
{
```

```
return array_values(unpack("C*", $key));
```

```
}
```

```
}
```

```
?>
```