International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 9, September- 2014

Development of Cryptographic Toolkit in LINUX

A Practical Approach for Better Understanding

Anjlee Verma
School of Computer Engineering
Lovely Professional University
Phagwara, Punjab

***Abstract*— The call for security can never be ignored. Cryptography, now a days has become one of the basic essential courses in the professional field of information security. So there is always a deemed necessity to analyze, standardize and represent the algorithms of encryption in a way that they can be used for learning and training as efficiently and effectively as possible. And as this subject is comprehensive, as well as complementary for the students, it becomes significant to improve their practice and innovation ability. Therefore for betterment of all this, the analytical presentation of an educational tool on learning algorithms of cryptography is must and profitable. Proposed Cryptographic toolkit has been developed in QT in which 15 cryptographic techniques and 6 key generation algorithms have been integrated. Further an interface for analyzing these schemes is also provided. By using this kit, students can be able to visualize the working of different security algorithms and can learn to design effective techniques for securing data. Further, and experimental evaluation of 24 test students illustrates the advantages and potential of the proposed approach.**

Index Terms*— Cryptography, encryption, decryption, key generator.

I. INTRODUCTION

If a person says, “Hey I don’t need any security. I don’t have any secret, I don’t have anything to conceal,” Answer by saying, “OK, and then would you let me go through your medical files? What about your bank statement? Paychecks? Investments? Credit card bills? Salary statements? Will you share your and bank account number? Social Security Number? Credit card number? What is your ATM pin? Online banking password? Facebook password? Where do you spare your house key?” … The crux is that all of us have got some information which we want to keep hidden and confidential.[1] The main reason for so could sometimes be simply our natural tendency for confidentiality, no one feels comfortable if our financial details and other secret information is known to the whole world. The CIA triad (confidentiality, integrity and availability) is one of the core principles of Information security. [2] “The strength of any system is no greater than its weakest link”. In order to completely protect the information in its lifetime, protection mechanisms must be applied on each and every component of the information processing system. There are various mechanisms which are commonly adopted in order to provide requisite protection to our resources:

- First level of security can be provided by limiting access to the computer system or media. For example, 'logon' authentication can be implemented (e.g. via passwords).
- Different profiles or access control mechanisms according to the roles can be employed.
- Third level of security can be provided by restricting physical access (e.g. the resource containing confidential data can be kept aside preventing access to the computer itself).

All of these above approaches can be precious and effective, but these can be equally disadvantageous and can possess serious shortcomings. So, a more fundamental approach is provided for maintaining data security. This approach is also called Cryptography or Cryptology [3].

Above given traditional access control mechanisms can be bypassed (for example via hacking). And also their security can be violated if the information has to be sent, or if the information media has to be taken outside the secure environment. Also nothing can be done if a number of persons are sharing the same computer environment. So, here Cryptography is a technique which comes into the picture. It is designed in such a way that it protects your information in such conditions.

Cryptography (also known as cryptology) is a practice and study of hiding information. It is the technique of taking any piece of raw data, then scrambling it into gibberish mathematically, yet allowing for decoding back into the original plain data. In other words, it can be defined as the art of manipulating messages and making them secure. It deals with the processes of encryption and decryption. Cryptology can also be defined as the science of secret writing. It is made up of two halves; which are cryptography itself and the cryptanalysis. Cryptography includes the techniques for creating various systems, procedures or algorithms for secret writing whereas cryptanalysis consists of the techniques of breaking them.[4] Cryptology was well established in ancient times, amongst both Greeks and Romans and both of them used to practice different forms of cryptography.[5]

As we know that with the advancement of technology, Cryptology has become the branch of computer science that has gained so much of attention with its great utility nowadays. So there is always a deemed necessity to analyze, standardize and represent the algorithms of encryption so that they can be used for learning and training as efficiently and effectively as possible. This should also be kept in mind that good practices and examples must be accompanied with it. Therefore for betterment of all this, the analytical presentation of an educational tool on learning algorithms of cryptography is must and profitable.

Various theories and approaches have been adopted by the teachers in the universities for enlightening the minds of pupils with the concepts of the subjects. For example a technique consists of allocation of assignments, tasks, activities and responsibilities to the students along an established calendar with the aim of developing a software product. Then they go through all the activities and stages of the software engineering process aiming to obtain a commercial product. The methodology has been tested for 4 years and the results clearly show an improvement in the students learning and skills mandatory for their professional development.[6] Other approach is peer tutoring and collaborative learning.[7] One more approach suggests that use of pedagogical tools and IT tools play important role in teaching.[8] It has been noticed that while teaching cryptography, faculty mainly focuses on math theory ignoring nourishing students' practical capability. Students can better understand the concepts and algorithms by the application of open source software. It can also encourage their practical skills, creative and innovation capability. By collaborating cryptography and open source experimental teaching, positive results have been obtained. Students have not only gained more interest in learning, but it has also improved the quality of practitioners desired.[9]

II. OBJECTIVES

Cryptography has become vital for providing security to users, applications and data, yet the field still lacks skilled technocrats and professionals and the reason might be typical or conventional methods of teaching this branch of security from academic papers and text books are not engaging and also take considerable time.

This section includes the objectives of the proposed approach for learning cryptography. Which are as follows:

- Development of an educational toolkit for teaching cryptographic techniques to students in an interesting manner.
- To contribute in the field of education and security by uplifting experimental teaching as the students can be able to visualize the working of different algorithms and can learn to design effective cryptographic techniques for securing data.

III. RESEARCH METHODOLOGY

For the development of an educational toolkit and for analyzing the working and performance of cryptographic techniques, QT creator and LINUX platform have been chosen. QT is a cross-platform complete development framework with tools designed to streamline the creation of stunning native applications and amazing user interfaces for desktop, embedded and mobile platforms. It is part of the QT Project. QT Creator is a cross-platform C++ integrated development environment which is part of the SDK for the QT GUI Application development framework. QT Creator includes a code editor and integrates QT Designer for designing and building graphical user interfaces (GUIs) from QT widgets. The code editor in QT Creator supports syntax highlighting for various languages. In addition to that, the code editor can parse code in C++. QT Creator uses the C++ compiler from the GNU Compiler Collection on Linux. There are some tag lines which are famous about QT and are self-explanatory. Few of those are [10]:

- “Power. Beauty. Portability. Target Everything with QT.”
- “Improve Product Lifecycle and Corporate Productivity with QT”
- “If You Can Imagine It, You Can Build It With QT.”

IV. LAYOUT OF PROPOSED TOOLKIT

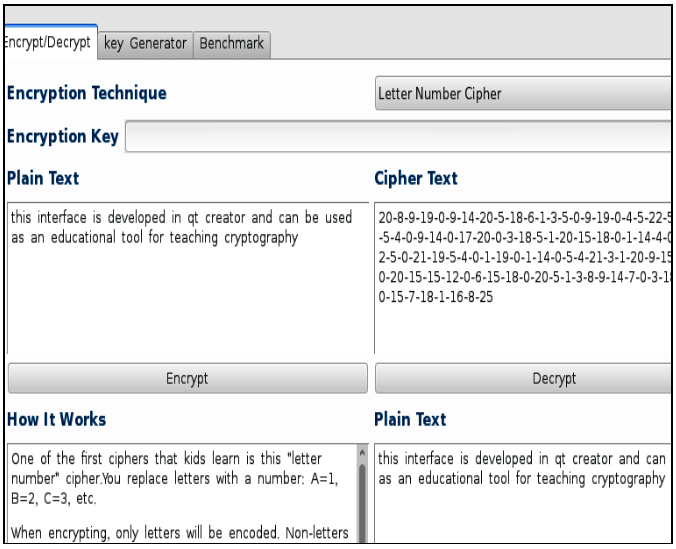
This section includes the toolkit elements and their significances. The designed proposed toolkit has 3 modules

a) Encrypt / Decrypt
b) Key Generator
c) Benchmark

a) Encrypt/Decrypt: This tab displays the working of 15 techniques of cryptography. These cryptographic techniques are:

- Affine Cipher
- Atbash Cipher
- Caeser Cipher
- Modified Caeser Cipher
- One Time Pad Cipher
- Columnar Cipher
- Columnar Key Cipher
- Baconian Cipher
- Polybius Square Cipher
- Vigenere Cipher
- Letter Number Cipher
- Rail Fence Cipher
- Mono Alphabetic Cipher
- Poly Alphabetic Cipher
- Rotate Cipher

Fig. 1 shows the graphical interface of Encrypt/Decrypt module.



Encryption Technique: Letter Number Cipher

Encryption Key

Plain Text

this interface is developed in qt creator and can be used as an educational tool for teaching cryptography

Cipher Text

20-8-9-19-0-9-14-20-5-18-6-1-3-5-0-9-19-0-4-5-22-5
-5-4-0-9-14-0-17-20-0-3-18-5-1-20-15-18-0-1-14-4-0
2-5-0-21-19-5-4-0-1-19-0-1-14-0-5-4-21-3-1-20-9-15
0-20-15-15-12-0-6-15-18-0-20-5-1-3-8-9-14-7-0-3-1
0-15-7-18-1-16-8-25

How It Works

One of the first ciphers that kids learn is this "letter number" cipher.You replace letters with a number: A=1, B=2, C=3, etc.

When encrypting, only letters will be encoded. Non-letters

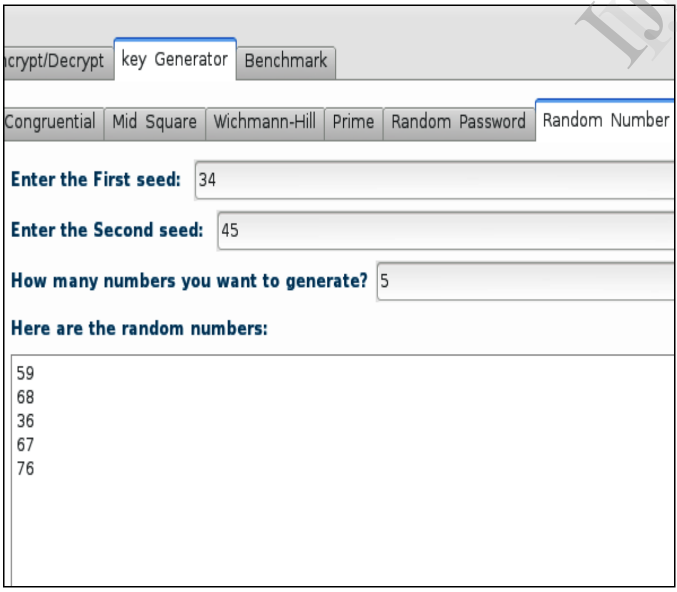
Plain Text

this interface is developed in qt creator and can as an educational tool for teaching cryptography

Fig. 1. GUI of Encrypt/Decrypt tab

This interface asks user to input plain text and the encryption key and then as user hits the encrypt button, text is encrypted and displayed in the corresponding text field. Also the basic idea followed by the algorithm is displayed in "how it works" box. Thus the working of algorithms along with the examples can be demonstrated by the teachers in the class to the students so that they can get a better understanding of cryptographic concepts.

b) Key Generator



Enter the First seed: 34

Enter the Second seed: 45

How many numbers you want to generate? 5

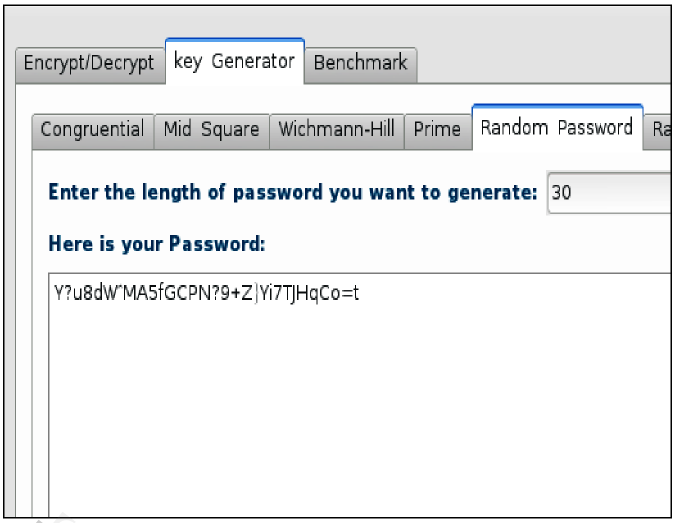
Here are the random numbers:

```
59
68
36
67
76
```

Fig. 2. GUI of key generator tab (i)

As we know that generating random bit sequences is an important problem in the field of cryptology. And many cryptographic systems are as stronger and secure as the unpredictable bit sequences are generated. Key generator tab (Fig. 2 and Fig. 3) includes 6 different schemes which can be used for generating random sequences. These techniques are:

- Congruential key generator
- Wichman-Hill algorithm
- Mid Square algorithm
- Random Number key generator
- Random password
- Linear Congruentail algorithm.



Enter the length of password you want to generate: 30

Here is your Password:

Y?u8dW`MA5fGCPN?9+Z|Yi7TJHqCo=t

Fig. 3. GUI of Key Generator tab (ii)

c) Benchmark

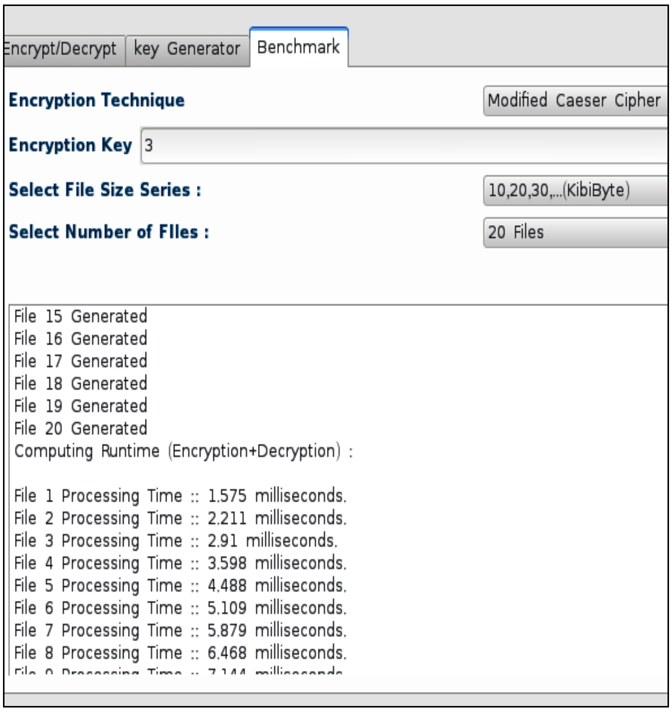
Fig. 4 shows the graphical user interface of benchmark tab. This interface allows users to evaluate and analyze various algorithms on the basis of text size and time. The proposed interface of this module firstly asks users to select the cryptographic scheme which they want to test. User is here provided with a drop-down list of all the names of algorithms which are integrated in encrypt-decrypt module. Then there is a tab "select file size series". This tab allows users to choose between three options. Which are:

- 10,20,30,40…. Kbs.
- 1,2,3,4……… Mbs.
- 10,20,30,40…..Mbs.

Then last tab helps users to select the number of files. Which could be any one of the following:

- 10 files.
- 20 files.
- 30 files.
- 40 files.
- 50 files.

User can also enter the encryption key in the respective field which is automatically enabled or disabled as per the selected cryptographic scheme. As user presses the "Benchmark start" button, mentioned number of files are created with random content and input as plain text to the algorithm chosen. A text field provided at the bottom of the interface displays the processing steps and time taken by the particular algorithm for encrypting and decrypting all the input files.



Encryption Technique: Modified Caeser Cipher

Encryption Key: 3

Select File Size Series : 10,20,30,...(KibiByte)

Select Number of FIles : 20 Files

```
File 15 Generated
File 16 Generated
File 17 Generated
File 18 Generated
File 19 Generated
File 20 Generated
Computing Runtime (Encryption+Decryption) :

File 1 Processing Time :: 1.575 milliseconds.
File 2 Processing Time :: 2.211 milliseconds.
File 3 Processing Time :: 2.91 milliseconds.
File 4 Processing Time :: 3.598 milliseconds.
File 5 Processing Time :: 4.488 milliseconds.
File 6 Processing Time :: 5.109 milliseconds.
File 7 Processing Time :: 5.879 milliseconds.
File 8 Processing Time :: 6.468 milliseconds.
File 9 Processing Time :: 7.144 milliseconds
```

Fig. 4. GUI of Benchmark tab

This would help students to visualize and analyze the working, speed and behavior of various algorithms according to the number of files and file size.

V. EVALUATION

Evaluation focuses on the testing of hypothesis that learning practical knowledge about security algorithms using this toolkit as pedagogy is more interesting, engaging and beneficial than gaining the same knowledge from books, academic papers and other study materials. Target sample for carrying out this project is 24 college students who have a technical background but not having any significant knowledge about cryptography. Students pursuing B. tech. and BCA are chosen for this purpose.

Initially all the students were given a preliminary questionnaire about cryptographic algorithms. And their basic knowledge about security was checked. Then those students were divided into two groups- A control group and an experimental group. Control group was provided with the summarized reading material on all the security techniques covered in the toolkit. On the other hand every student of the experimental group was asked to study through the toolkit for the same amount of time. After that all the students were given a new questionnaire similar to first one containing some additional questions to know their interest in the particular subject. The whole process was carried out in the presence of 2 invigilators to maintain the veraciousness of the tests.

VI. RESULTS

This section presents the results obtained from the experiments. Fig. 5 shows the average student marks before and after the experiments. It can be easily depicted form the graph that before carrying out the experiment, average knowledge of control and experimental group about security algorithms was approximately same. There was not much difference. But graph after post-test tells that although control group in which students studied through notes and text books have gained knowledge but, experimental group was able to get a clearer understanding about the same concepts using the toolkit.

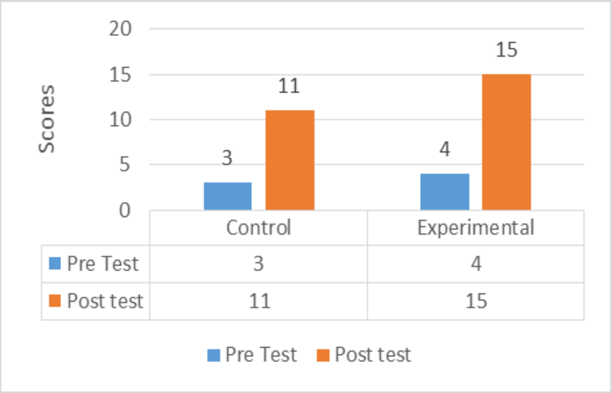


Fig. 5. Student marks on security test questions.

VII. CONCLUSION

Cryptography has a tremendous potential. However there can be challenges and reluctance in learning cryptographic techniques if traditional reading materials are used. Therefore in order to learn, teach, analyze, standardize and represent the algorithms of encryption as efficiently and effectively as possible this toolkit has been developed. This will not only help teachers to enlighten the minds of students with cryptographic schemes but also provide a natural way to get students to discover certain key mathematical concepts and techniques on their own. Students can thereby learn through the joy of discovery and can have acquisition of those practical skills and attitudes which are mandatory for their professional development.

VIII. FUTURE WORK

In the last phase of development of the toolkit, some more ideas were suggested and generated. In the future, this cryptographic toolkit can be fairly easily extended by integrating following modules:

- Frequency Analyzer: To calculate the number of occurrences of any alphabet, number and punctuation.
- Text Manipulator: To change something from lowercase to uppercase, Remove spaces or add spaces at every X characters.
- Cryptogram Jumbler: To display all the permutation combinations of a string entered.
- Implementation of some modern ciphers like RSA, DES, AES etc. so that this toolkit can be used for teaching advanced courses of cryptography too.

REFERENCES

[1] http://cryptozine.blogspot.com.es/2008/05/why cryptography.html

[2] http://security.blogoverflow.com/2012/08/confidentiality-integrity-availability- the-three-components-of-the-cia-triad/

[3] http://www.ciphersbyritter.com/LEARNING.HTM

[4] J. F. Dooley, "A Brief History of Cryptology and Cryptographic Algorithms," pp. 1-9, 2013.

[5] J. F. Dooley, "Cryptology Before 1500: A Bit of Magic," pp. 11-17, 2013.

[6] I. L. Ruiz and M. A. Gomez-Nieto, "Rolling: A new technique for practical teaching in computer science university degree," Education and Information technologies, pp. 49-77, 2012.

[7] A. Theodoropoulos, A. Antoniou and g. Lepouras, "Students teach students: Alternative teaching in Greek secondary education," April 2014.

[8] A. Iglesias, P. Martinez, R. Aler and F. Fernandez, "Learning teaching strategies in an Adaptive and Intelligent Educational System through Reinforcement Learning," pp. 89-106, August 2009.

[9] C. H. T. R. X. Zhang, "Research on Application of Open Source Software in Cryptographic Experimental Teaching," 2010.

[10] http://qtproject.org/wiki/Category:Tools::QtCreator