Usability Assessment on Symmetric Cryptography Learning Module

Irfan Syamsuddin
Department of Computer & Networking Engineering
Politeknik Negeri Ujung Pandang, Makassar, Indonesia
irfans@poliupg.ac.id

Arif Bramantoro
Faculty of Computing and Information Technology in Rabigh
King Abdulaziz University, Saudi Arabia
asoegihad@kau.edu.sa

Abstract **— Teaching information security requires lecturer to introduce several tools to encourage students performing hands on practices in better way in order to enhance their understanding. In this study, a novel learning module as a tool for student to perform symmetric cryptography exercises is introduced. It is called NgeXTEA, a novel learning module to describe how a simple yet powerful symmetric cryptography called Extended Tiny Encryption Algorithm (XTEA) works. The main objective of the study is to conduct assessment from usability point of view on NgeXTEA learning module using System Usability Scale (SUS) method. Finally, SUS based assessment indicated that the learning module is considered useful to enhance students' understanding and skills on the given topic.**

Key words -. Extended Tiny Encryption Algorithm (XTEA), Symmetric Cryptography, Learning Module, System Usability Scale

I. Introduction

Different types of organizations in business and government are facing serious cyber risks [1][2] due to the growing reliance on using information and communication technologies [3]. Considering such issues, ACM and IEEE have introduced the Information Assurance and Security to the Body of Knowledge to emphasize the urgent need learning and having adequate skill and knowledge regarding security and privacy issues in digital world to students of computer engineering and science [4].

The main objectives of the approach is to provide world standard guidance for educational institutions that offers information security related courses to structure the curriculum with up to date knowledge and skills to protect information and information systems and to guarantee Confidentiality, Integrity, and Availability (CIA) at all levels who strongly rely on computer and the Internet for daily tasks [2][5].

Although the importance of security methods and applications have been regarded as vital requirements in the computer engineering and science disciplines, in developing countries many computer engineering and science programs at higher education levels are still lacking of security expertise and lab resources to answer the requirements [6].

Therefore, higher education admits it as a gap that must be filled by continuous efforts to provide supporting tools and teaching materials for assisting students to gain adequate knowledge and skills in the area of information security [5][7].

Information Security course offered to the third year students majoring Computer & Network Engineering, at the School of Electrical Engineering of the State Polytechnic of Ujung Pandang, Indonesia. Similar to other courses that combine theory and practice, this course also requires a number of tools for laboratory exercises in order to help students understanding the given topic.

It usually requires hard efforts and time to teach cryptography topics such as Symmetric Cryptography and Asymmetric Cryptography until student gain understanding on the topics. Several problems associated in this case such as inadequate time for class, various mathematical backgrounds of the students, and limited tools to student for doing security practices [8]. Although we found particular security simulation tools, all of them could not support our teaching as they directly showing final result of encryption or decryption without adequate description on how the final results are obtained step by step. As a result, it will be difficult for students to understand mathematical calculation behind the cryptography algorithm [7][9].

In order to deal with the issue, a new learning module for Symmetric Cryptography topic is introduced. The learning module is called NgeXTEA which stands for "Ngerti XTEA" or "understanding XTEA" in Indonesian language. As might be infer from its name, NgeXTEA is based on Extended Tiny Encryption Algorithm (XTEA) as a basis to understand mathematical calculation of Symmetric Cryptography. NgeXTEA module comes with its easy to use Graphical User Interface (GUI) so it is a self-explainer learning tool for students to learn cryptography easier.

The main purpose of the study is to assess NgeXTEA learning module from usability perspective. Learning module such as NgeXTEA requires different views of evaluation in order to measure how useful it is according to the perspective of its users. Among many methodologies, System Usability Scale or SUS [10][18], a widely used technique to measure usability is employed as the methodology to conduct the study.

978-1-7281-5467-1/20/$31.00©2020IEEE

II. Literature Review

A. Extended Tiny Encryption Algorithm

XTEA encryption algorithm actually derived from TEA or Tiny Encryption Algorithm [11], which both developed by Wheeler and Needham in 1996 [12]. The strength of XTEA lies in its lightweight size comes from few lines of code that make it fast in operation [13] and its simple yet strong encryption mechanism among symmetric cryptography families [14].

The following flowchart describes the whole steps of XTEA for both encryption and decryption [12] enriched by additional learning features to support end user requirements for learning module.

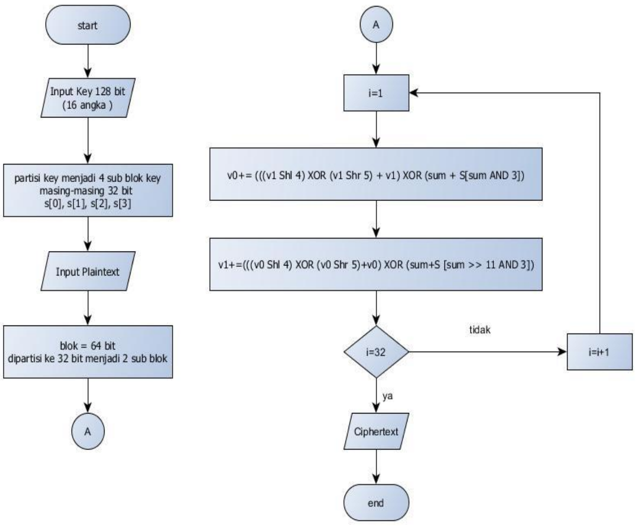

start
Input Key 128 bit (16 angka)
partisi key menjadi 4 sub blok key masing-masing 32 bit s[0], s[1], s[2], s[3]
Input Plaintext
blok = 64 bit dipartisi ke 32 bit menjadi 2 sub blok
A
A
i=1
v0+= (((v1 Shl 4) XOR (v1 Shr 5) + v1) XOR (sum + S[sum AND 3])
v1+=(((v0 Shl 4) XOR (v0 Shr 5)+v0) XOR (sum+S [sum >> 11 AND 3])
i=32
tidak
i=i+1
ya
Ciphertext
end

Fig. 1 XTEA encryption flowchart

.

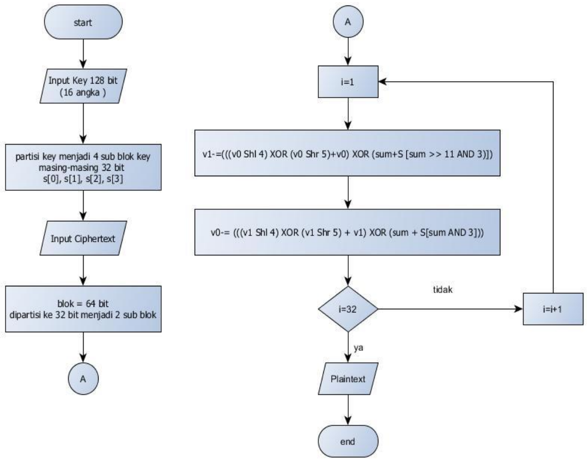

start
Input Key 128 bit (16 angka)
partisi key menjadi 4 sub blok key masing-masing 32 bit s[0], s[1], s[2], s[3]
Input Ciphertext
blok = 64 bit dipartisi ke 32 bit menjadi 2 sub blok
A
A
i=1
v1-=(((v0 Shl 4) XOR (v0 Shr 5)+v0) XOR (sum+S [sum >> 11 AND 3)])
v0-= (((v1 Shl 4) XOR (v1 Shr 5) + v1) XOR (sum + S[sum AND 3]))
i=32
tidak
i=i+1
ya
Plaintext
end

Fig. 2 XTEA decryption flowchart

In detail, the XTEA implements encryption using a 64-bit block split into two 32-bit halves, v0 and v1, which are input to the algorithmic routine that performs 32 rounds (Nr = 32) [9]. XTEA's key scheduling is modified to reflect different patterns for mixing the data and key continuously each round to add substantial confusion. Only four subkeys, each having a 32-bit length, are used, and basic addition and subtraction operations follow the modulo 232. The logic shifts are a logical left shift by four and a logical right shift by five, in addition to a simple 32-bit XOR logic operation [12].

The permutation functions are expressed by f(x) = (x<<4 Å x >> 5) + x, and subkey generation functions are expressed by sum + k (sum Ú 3), and sum + k (sum >> 11 Ú 3). Sum acts as a selector from the four subkeys k0, k1, k2, and k3 dependent on bits 0 and 1 of the sum or bits 11 and 12. The results of the permutation function and generated subkey generated are XORed and ADDed to v0 and v1. It is worth noted that the value of sum is initialized to zero prior to the start of the computation, and the value of delta is fixed to 0x9E3779B9 [12].

B. NgeXTEA Learning Module

The development of NgeXTEA is fully based on open source software. PHP and Javascript. Both encryption and decryption processes are written in PHP [15], while Javascripts [16] is used for developing GUI as well as to enhance user interactivity of NgeXTEA [17].

It begins by creating pseudo code of both encryption and decryption of XTEA as shown in table 1.

TABLE I
Pseudocode of XTEA

1	XTEA Encryption
	Sum = 0; delta = 0x9E3779B9 *for i = 0 to N do* *v0+ = ((v1 << 4) xor (v1 >> 5) + v1) xor (sum + key[sum & 3])* *sum+ = delta* *v1+ = ((v0 << 4) xor (v0 >> 5) + v0) (sum + key[sum >> 11 & 3])* *end for*
2	XTEA Decryption
	Sum = 0; delta = 0x9E3779B9 *for i = 0 to N do* *v1- = ((v0 << 4) xor (v1 >> 5) + v0) xor (sum + key[sum >> 11 & 3])* *sum+ = delta* *v0- = ((v1 << 4) xor (v1 >> 5) + v1) (sum + Key[sum &3])* *end for*

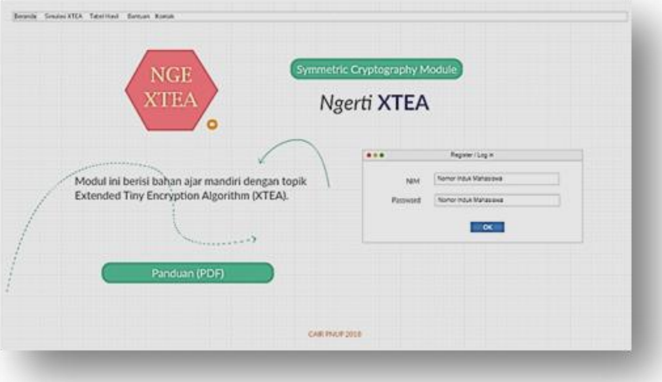
Once pseudo code is clearly verified, the next step goes to developing the software or coding in PHP. Figure 3 shows PHP code for encryption process of XTEA, while the decryption process in PHP code can be seen in figure 4.

Every round of XTEA processes (both encryption and decryption) is clearly explained with clear comment line. In addition, each round result always treated as variable which value is then showed through the GUI of NgeXTEA in order to let students comparing the simulation results with their manual calculations at every round for both encryption and decryption. This is considered as the main feature of NgeXTEA which makes it understandable and user friendly even for those who never learn cryptography before.

Once both encryption and decryption processes of XTEA successfully tested with different combination of alphabet and numeric inputs, the next step is developing the learning module called NgeXTEA. NgeXTEA stands for "Ngerti XTEA" which means understand XTEA in Indonesian language. NgeXTEA is an integrated teaching module for XTEA which is developed using PHP, Javascript and CSS.

The main user interface (UI) of NgeXTEA is represented in figure 3. All instructions, text menu and text buttons are currently only available in Indonesian language. NgeXTEA's simple UI design has an intuitive look and feel which clearly aid users get the idea of what the application is for.

In terms of usability aspects, we have provided NgEXTEA with easy to access menus that will prompt user to specific function according to user needs. All menus are provided with clear text and appropriate color.



Modul ini berisi bahan ajar mandiri dengan topik Extended Tiny Encryption Algorithm (XTEA).

NIM: Nomor Induk Mahasiswa

Password: Nomor Induk Mahasiswa

Fig. 3 XTEA learning module

There are two simulations provided, called encryption and decryption which could be used by student. Student will be prompted to first mode of encryption by inserting "Plaintext" and "Key" into the given textboxes (in case of encryption). The results will be given step by step in order for students to gain understanding on how the encryption works. Student will see these in the next menu called "Tabel Hasil" showing the results of all rounds from "Plaintext" to "Ciphertext" [17]. Similar procedures applied to decryption process.

Simple but comprehensive guidance for users is also provided through a link of "Panduan (PDF)". It is a practical documentation in Indonesian language. Students may refer to the guideline (i.e. Panduan PDF) to follow the steps for conducting manual cryptanalysis whether by MS Excel or MS Calculator. Similar steps in descending way should be followed to conduct decryption process to the "Ciphertext" in order to gain the "Plaintext".

In the exercises, students will try three different style of plaintexts (text, number and combination of text and number) with different keys and compare the results of NgeXTEA module with its own manual calculation to create project report.

All findings might be reported by students individually to the lecturer via email textbox provided in "Kontak" menu [17].

Based on experiments on using NgeXTEA and following the given guidelines, in this case we need to evaluate the usability aspects of the learning module. The process of doing such assessment is performed by applying System Usability Scale methodology [10].

III. Methodology

System Usability Scale or SUS is used to perform assessment on NgeXTEA learning module. It was introduced by Brooke [10] as a simple but valid usability scale methodology that can be used for any assessments of systems usability.

Sauro [18] states that SUS has been applied in more than 600 case studies of various areas where usability evaluation is required. Therefore, SUS is considered as an industry standard since its introduction in 1986 [18[19].

According to SUS's inventor, usability testing might be appropriate to any kind of area as long as two aspects are maintained. Firstly, intended users of the system being assessed should clearly be defined. Secondly, all required tasks that the users intended to complete are well understood [10][19].

In this study, SUS survey was conducted in a laboratory based testing environment where users are asked to carry out specific tasks with NgeXTEA learning module in a controlled setting.

Jake-Schoffman et.al. [19] argue that for the purpose of assessing usability requirements in a particular population like in this study, laboratory based testing is appropriate.

Essentially, SUS consists of 10 survey questions with a 5-response Likert scale. The Likert formatting ranged from 1 to 5 for strongly disagree to strongly agree. The 10 survey questions are as follows with slightly modification according to the requirement in this study.

TABLE II
SUS survey questions

10 Questions of System Usability Scale				
Q1. I think that I would like to use NgeXTEA frequently.				
1	2	3	4	5
Q2. I found the system of NgeXTEA unnecessarily complex.				
1	2	3	4	5
Q3. I thought NgeXTEA was easy to use.				
1	2	3	4	5
Q4. I think that I would need the support of a technical person to be able to use NgeXTEA for the first time.				
1	2	3	4	5
Q5. I found the various functions in NgeXTEA were well integrated.				
1	2	3	4	5
Q6. I thought there was too much inconsistency in NgeXTEA.				
1	2	3	4	5
Q7. I would imagine that most students would learn to use NgeXTEA very quickly.				
1	2	3	4	5
Q8. I found NgeXTEA very cumbersome to use.				
1	2	3	4	5
Q9. I felt very confident using NgeXTEA.				
1	2	3	4	5
Q10. I needed to learn a lot of things before I could get going with NgeXTEA.				
1	2	3	4	5

Calculating the SUS value as explained by Sauro is quite simple. From these calculations the SUS value will be generated in the form of a percentage that represents all aspects of the overall use of the NgeXTEA system being studied. The process is as follows. For odd numbered items (eg 1, 3, 5, 7, and 9), reduce the participant response score by number 1. While for even numbered questions (eg 2, 4, 6, 8, 10) reduce the participant response score of 5.

Then calculating SUS value is done by summing the scores of each item, which has a range of 1 to 5 with 1 strongly disagreeing and 5 strongly agreeing. At this stage values will be obtained are in between 0 to 4, in which 4 represents the most positive response to each question.

Finally, after adding up all the individual scores, multiply by 2.5 to get the overall SUS score. This final value is the SUS score that can be mapped in the range of SUS values to determine the extent of the use of NgeXTEA according to respondents [10] [19].

Sauro [18] presents a normalization technique to justify final SUS scores could be classified into A, B, C, D, E and F usability groups. Thus, interpreting SUS scores is easy and straightforward from strongly usable (group A) and useless (group F).

IV. Results and Analysis

In a controlled environment, 19 respondents fulfil the SUS survey. It is an open process where respondents may ask if they do not understand the purpose or scope of a particular question.

Then, all respondents submitted the survey and we conducted the calculation to obtain SUS score of each. The highest SUS scale is 100 while the lowest one is 65. The whole results are given in table 3.

These yield average SUS scale for NgeXTEA of 80.39 which falls into A group (see figure 4). This finding clearly indicates that students consider the NgeXTEA learning module is very useful tool to enhance their learning. Based on the finding, the module will be continuously applied in the coming years as a supporting learning module to enhance students' knowledge and skills in symmetric cryptography topic.

In the future, this study will be extended in two ways. Firstly, considering its small size and strength in securing information, XTEA might be applied for mobile cloud computing security such as secure chat for mobile government.

Secondly, NgeXTEA will be improved by making it available with several language options such as English and Arabic before releasing it publicly as open source software to make it more widely used.

V. Conclusion

A new learning module called NgeXTEA is developed and introduced in this study. In order to find out its usability as learning module, a usability assessment is required. The usability assessment is based on System Usability Scale method that consists of ten questions with specific calculation to obtain level of usability called SUS score. The process is conducted in a controlled environment in which students are the respondents.

It is finally revealed that SUS score of NgeXTEA is 80.39 which means the students acknowledge it as a useful learning tool to enhance their understanding on the given security topic of symmetric cryptography.

TABLE III
Results of SUS survey

Student	*S 1*	*S 2*	*S 3*	*S 4*	*S 5*	*S 6*	*S 7*	*S 8*	*S 9*	*S 10*	*Score*
A001	5	1	5	1	5	1	5	1	5	2	**97.5**
A002	3	1	4	3	5	2	4	1	3	1	**77.5**
A003	5	2	5	3	5	2	4	3	5	1	**82.5**
A004	3	1	3	3	3	1	5	2	4	2	**72.5**
A005	4	3	4	2	4	4	5	3	2	1	**65.0**
A006	4	1	2	2	3	2	5	1	5	1	**80.0**
A007	5	1	5	1	5	1	5	1	5	1	**100.0**
A008	4	3	2	2	3	2	2	1	5	1	**67.5**
A009	5	2	5	3	5	2	4	2	5	1	**85.0**
A010	4	1	2	2	5	2	2	1	5	1	**77.5**
A011	3	1	2	2	5	2	2	1	5	1	**75.0**
A012	5	2	5	3	5	2	4	1	5	2	**85.0**
A013	5	1	2	2	3	1	2	1	5	1	**77.5**
A014	4	3	3	2	4	2	2	1	5	1	**72.5**
A015	5	2	5	3	5	1	4	2	5	1	**87.5**
A016	4	1	4	2	3	2	2	1	5	1	**77.5**
A017	5	3	5	2	5	1	2	1	5	2	**82.5**
A018	4	2	5	3	5	1	4	3	5	1	**82.5**
A019	5	1	4	2	3	1	2	1	5	1	**82.5**
SUS Score : 80.39											

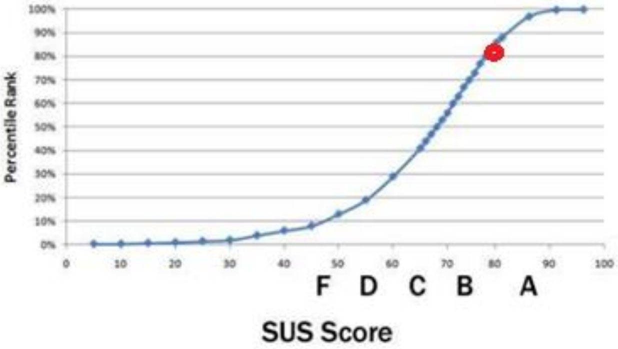


Fig. 4 Classification of SUS result

REFERENCES

[1] Syamsuddin, I., and Hwang, J. "A new fuzzy MCDM framework to evaluate e-government security strategy". *2010 4th IEEE. International Conference on Application of Information and Communication Technologies (AICT2010)*, pp. 1-5, 2010.

[2] Syamsuddin, I., and Hwang, J. "The application of AHP to evaluate information security policy decision making". *International Journal of Simulation, Systems, Science and Technology*, 10, 46-50, 2009.

[3] Oulasvirta, L and Anttiroiko, A, "Adoption of comprehensive risk management in local government". *Local Government Studies* vol. 43 , pp. 451-474, 2017..

[4] ACM/IEEE, ACM/IEEE-CS Computer Science Curricula, https://www.acm.org/education/ CS2013-final-report.pdf, 2013.

[5] Olejar, D and Stanek, M., "Some Aspects of Cryptology Teaching", Proceedings of *the WISE1—IFIP WG 11.8 1st World Conference on Information Security Education,* Stockholm pp. 1–9, 1999.

[6] Elnajjar, A.E.A. and Abu Naser,S.S., DES-Tutor: “An Intelligent Tutoring System for Teaching DES”, *Information Security Algorithm*, vol. 2, pp. 69-73, 2017.

[7] Yuan,X., Vega,P., Qadah,Y., Archer,R., Yu,H. and Xu,J., “Visualization Tools for Teaching Computer Security”, *ACM Transactions on Computing Education*, vol. 9, pp.147–155, 2010.

[8] Adamovic, S., Sarac, M, Stamenkovic,D. and Radovanovic,D., “The Importance of the Using Software Tools for Learning Modern Cryptography”, *International Journal of Engineering Education*, vol. 34, pp. 256–262, 2018.

[9] Briliyant, O.C. and Baihaqi,A. “Implementation of RSA 2048-bit and AES 128-bit for Secure e-learning web-based application”. *11th International Conference on Telecommunication Systems Services and Applications (TSSA)*, pp. 25-31, 2017.

[10] Brooke, J., “*SUS: A ‘quick and dirty’ usability scale*”. In P. Jordan, B. Thomas, & B. Weerdmeester (Eds.), Usability Evaluation in Industry, London, UK: Taylor & Francis. pp.189–194, 1996.

[11] Wheeler, D. J. and Needham, R.M. TEA, “A Tiny Encryption Algorithm”. *International Workshop on Fast Software Encryption* pp. 363-366, 1994.

[12] Needham, R.M. and Wheeler, D.J. “*Tea Extensions*”. Report, Cambridge University, 1997.

[13] Ebrahim,M, Khan, S. and Khalid, U. “Symmetric Algorithm Survey: A Comparative Analysis”. *International Journal of Computer Applications*, vol. 61, pp. 12-19, 2013.

[14] Ballal,V. Kumar, K. Meghan, V. and Rai, S.R., “A Study and Comparison of Lightweight Cryptographic Algorithm”, *IOSR Journal of Electronics and Communication Engineering*, vol. 12, pp. 20-25, 2017.

[15] Baatard, G. “Teaching PHP with Security in Mind”. *Australian Information Security Management Conference*, vol. 21, pp. 21-23, 2007.

[16] Dipierro, M.. “The Rise of JavaScript”. *Computing in Science & Engineering*, vol. 20, pp. 9-10, 2018.

[17] Syamsuddin, I., “Evaluation of NgeXTEA A Cryptography Learning Module”, *Global Journal of Engineering Education*, vol. 20, no. 3, pp. 196-200, 2018.

[18] Sauro, J. “*Measuring Usability With The System Usability Scale (SUS)*”. Retrieved from: http://www.measuringu.com/sus.php 2011.

[19] Jake-Schoffman, D.E., Silfee, V.J., Waring, M.E., Boudreaux, E.D., Sadasivam, R.S., Mullen, S.P., Carey, J.L., Hayes, R.B., Ding, E.Y., Bennett, G.G. & Pagoto, S.L., “Methods for evaluating the content, usability, and efficacy of commercial mobile health apps”. *JMIR mHealth and uHealth*, vol 5, pp. e190, 2017.