A Framework for Developing Visualization Tools for Learning Cryptography Algorithms

Yen-Hung Hu
Department of Computer Science
Hampton University
Hampton, Virginia, 23668
1-757-727-5552
yenhung.hu@hamptonu.edu

Robert Willis
Department of Computer Science
Hampton University
Hampton, Virginia, 23668
1-757-727-5552
robert.willis@hamptonu.edu

ABSTRACT
To enhance Computer Science / Computer Information System major students and other information technology (IT) professionals the understanding of cryptography algorithms, a tool, which visually demonstrates and explains key procedures of each problem and can be easily accessed anywhere through the Internet could be very helpful. The department of Computer Science at Hampton University realizes how important it could be to provide a visualization-enhanced learning environment for students who are preparing to engage themselves in information security profession. This environment should be able to present most complicated and profound cryptography algorithms in a visible approach as well as provide interactive capability for students to operate variant measurements of such algorithms and study their performances. The Information Assurance (IA) team in the department proposes to develop such a tool. It will not only meet the requirements for IA educational purposes but also be sufficient to conduct performance study researches of most well-known cryptography algorithms. In this paper, a framework that supports our objectives is presented. Rationale and analysis of the framework are also included.

Categories and Subject Descriptors
K.3.2 [Computers and Education]: Computer and Information Science Education

General Terms
Algorithms, Design, Security

Keywords
Cryptography, Information Assurance, Information Security

1. INTRODUCTION
Cryptography algorithms serve as one of the most important roles to protect current IT products. No matter what they are, from daily gadgets to high-tech equipments, cryptography algorithms ensure that the security of them are enforced and services for them are well provided and protected. Imagine that, how wonderful it is: when the secure socket layer (SSL) protocol and its successor the transport layer security (TSL) protocol [1] work closely with Internet browsers and systems to provide a safe zone where you can surf the Web without any hesitation; when authentication procedures successfully block someone who attempt to access your system without your permission. In these cases, cryptography algorithms implicitly affect our life. These cases also show that their success depends on performance against measurement criteria of the implemented cryptography algorithms. Several popular approaches (i.e., authentication, authorization, confidentiality, and integrity) have been chosen as the measurement criteria to assess performance of each cryptography algorithm and have been very successful to help to develop more secure algorithms. For instance, electronic codebook (ECB) block cipher mode has demonstrated failure to protect the confidentiality of the data since identical plaintext blocks will be encrypted to the same ciphertext block, which allows patterns in the plaintext to bleed through into the ciphertext [2] [17]. Therefore, a more complicated block cipher mode, cipher block chaining (CBC), has been developed to mitigate this issue. Another example is integrity violation. Message digest algorithm MD5 [3] has been used since 1992 and has been the major integrity checking program. Several countermeasures [4] [5] [6] [7] against this algorithm have been released and demonstrate that MD5 is not as secure as we thought. Although, there is no theoretical proof of better security, the combinations of multiple cryptography algorithms could be able to enhance the protection of integrity [8].

Although, they are important and of good use, cryptography algorithms are still not the ones that students are very willing to learn and practice. Complicated and profound theories and algorithms may retard students' progresses in learning cryptography algorithms. Other subjects such as programming languages have faced similar challenges. The difficulties of mastering early programming courses may scare many students away [9]. To draw students back to the computer science major, several researchers and educators have proposed projects of adopting computer games [9] [10] [11] [12] into the Computer Science curriculum. Research results in [12] show that the strategy of adopting computer games does work.

"... roughly 1/3 of our first year class has been attracted by the game courses ... K. Becker et. al.[12]"

This example points out a potential direction of teaching and learning cryptography algorithms. A tool, which could visually demonstrate and explain the key procedures of each cryptography algorithm and can be easily accessed anywhere through the Internet is very helpful.

This paper will describe the framework and architecture of our visualization tool. The rest of the paper is organized as follows: the next section discusses related work. The following section
describes framework and architecture of our visualization tool, while the last section concludes the paper.

2. RELATED WORK
In 1984, M. H. Brown et. al. [13] demonstrated their algorithm animation system BALSA, which provides graphical views and data structures of algorithms. Since then, many visualization tools for algorithms have been developed and provided to educators and researchers [14]. C. A Shaffer et. al. [16] have researched algorithm visualization tools that are available on the Internet. They have collected links to over 350 visualization tools. Most of them have been implemented as JAVA Applets, a few of them as JAVA applications. About 50% of the tools are made by individuals or small groups. Their results also show:

"A majority of the visualizations that we have encountered either cannot be made to work easily, or appear to have no pedagogical value ... C. A. Shaffer et. al. [16]"

Only one visualization tool [15] in [14] is relating to a cryptography algorithm. Susan Gerhart et. al. [15] have implemented cryptography visualization tools to demonstrate procedures of buffer overflow, S-DES, differential cryptanalysis, linear cryptanalysis, generic hash function, MD5, and SHA1. Even if only a few algorithms are covered, graphical and structural demonstrations and explanations of each procedure they implemented make their efforts worthy of recommendation.

3. THE VISUALIZATION TOOL
As mentioned, there are very few visualization tools available for cryptography algorithms. The purpose of this paper is to provide a framework of developing visualization tools for cryptography algorithms. From our efforts to survey the state of the field of cryptography algorithm visualizations, the perspective tool should be able to demonstrate the following characteristics:

- Accommodate most symmetric and asymmetric algorithms: The tool should cover most symmetric and asymmetric cryptography algorithms. By default, it should be very easy for users to learn most cryptography algorithms by only a few clicks. In our design, it should include at least the following algorithms: one-time pad, A5/1 [17], DES [17], Triple DES [17], AES [17], RC6 [17], ECB [2] [17], CBC [2] [17], CTR [2] [17], Knapsack, RSA [17], Diffie-Hellman [17], MD5, Sha-1, SSL, TSL, etc.
- Explain and demonstrate key procedures: The tool should be capable of demonstrating and explaining key procedures of each algorithm. This allows users to learn and understand mechanisms of such an algorithm. For example, for each round of DES, the tool should be able to demonstrate the details of key generation, expansion, substitution box (S-Box), permutation box (P-Box), and data and key for the next round.
- Interact with users: The tool should allow users to select parameters for each algorithm and let them decide which procedures should be highlighted and demonstrated. For example, in DES case, user should be able to control parameters such as key size, key value, input value, and round number and select which procedures should be displayed and explained in detail.
- Adopt new algorithms: Users can adopt new algorithms by using the predefined modules and can measure performances of the new algorithms. This provides the proposed tool the power of updating itself to meet the requirements of the rapidly evolving IT professions.

![Diagram of the Visualization Tool Framework](image)

---

**Figure 1. Framework of the visualization tool**

### 3.1 Framework
As shown in figure 1, our framework includes two categories: existing algorithms and new algorithms. When an existing algorithm is selected, a parameter configuration module will popup to ask users for inputs of parameters. After that, a demonstration and explanation module shows up for users to setup detailed displays of key procedures. At the end, users will also be able to adopt performance analysis modules for accessing performance of the selected algorithm.

When a new algorithm is selected, the tool will ask users to build it from the pre-defined modules. The predefined modules are objects that can be chained together towards the purposes of the users. For instance, a new algorithm could be formed by chaining several expansion boxes and substitution boxes together (see figure 2). After building the new algorithm, characteristics of the other four modules are the same as described in the previous case.
CONCLUSIONS AND FUTURE PLANS
We propose a framework for developing visualization tools for learning cryptography algorithms. Our framework allows users to use the existing algorithms or create new ones; allows users to setup parameters for each algorithm; allows users to customize outputs; and allows users to study performance comparisons. We believe that with this visualization tool, teaching and learning of cryptography algorithms will be more efficient and more attractive. Our next plan is to implement this framework through JAVA applets since they are flexible and can be embedded into Internet Browsers.

REFERENCES
[8] Gerardo Richarte, "MD5 To Be Considered Harmful Today", PacSec/Core05 2005, Tokyo, Japan.