Journal of the Nigerian Association of Mathematical Physics
Volume 54 **(January** *2020 Issue*), **pp137 – 142**
© J. of NAMP

AN ENHANCED WEB BASED DATA PROTECTION SYSTEM FOR EDO STATE POLYTECHNIC USEN, USING A COMBINATIONAL APPROACH

Ihama E.I., Izogie L. E. and Iyamu Iziegbe

Department of computer Science and Information Technology, School of Applied Sciences, Edo State Polytechnic, Usen P.M.B 1104, Benin City, Nigeria.

Abstract

As technology is advancing, more data (information) are stored and used daily. This has also generated more security risk along the different network or computer systems, where these information (data) are stored or being transmitted across (different networks), which may not be properly secured. There is need for the information to be well secured while in storage and while being transmitted. We developed an enhanced web based data protection system to help secure our information while in storage or when being transmitted. This system was designed using combinational approach (an alphanumeric and a hashing technic), it will help prevent un-authorized users or intruders from gaining access to information systems maliciously. Information properly managed and secure, gives information system users confidence to process and store their information in the computer system. It also protects it while being transmitted via unsecured network or a public network (internet). The system was design using these approaches, the front-end was develop with PHP, Microsoft Visual Studio.net was integrated in the development environment and Microsoft SQL Server was used at the database backend.

Keywords: key, alphanumeric, combinational, malicious, data

Introduction

Cryptography is a procedure of changing and transmitting secret information in an encoded manner with the goal that only approve and plan users can acquire or take and have access to the information.

It is a Greek beginning word in which "crypto" signifies covered up and "graphy" signifies composing [1], so cryptography implies covered up or the mystery of composing. It is presented in sets of three, which are classification, non-disavowal, honesty and validness inside continuous information correspondence.

Cryptography is defined as the science of inscription in secret code and using calculation to encrypt and decrypt the information. Cryptography enables the user to store delicate information or convey it across insecure networks (like the Internet) so that it can only be read by the intended recipient, and not by unauthorized recipient.

According to [2], they confirm how to select constants for an improved version of SHA-1 enabling a saboteur to help an attacker later find collisions of a certain form. Both of these examples are also examples of choosing weak constants, a strategy discussed below. With this weakness, exploitability hinges on knowledge of the cryptanalytic attack. While at first glance, this may be a way to allow secrecy. History shows that other researchers often reproduce knowledge first developed behind closed doors.

In the works of [3], they explored SETUP attacks for instances of symmetric encryption, and rebrand SETUP attacks as algorithmic substitution attacks (ASAs). A SETUP attacks against symmetric encryption was established, to seek countermeasures. They argued that a symmetric encryption scheme is secured against sabotage if no attacker is established, or even given a secret trapdoor, it can differentiate between ciphertexts generated by the reliableposition algorithm against one generated by a backdoor version of it. The SETUP and ASA structures do not capture all relevant aspects of a cryptographic weakness, nor potential routes for a saboteur to achieve one.

Ease of use characterizes the computational and logistical difficulty of mounting an attack using the weakness. For example, the Dual EC backdoor can be tricky to exploit in certain situations [4]. There is a string of high profile vulnerabilities of TLS certificate checking: Apple's double goto bug in [5], a bug in OpenSSL in [6], and the results of a number of research papers showing certificate checking bugs in numerous applications [7]. While we have no reason to expect that any of these bugs were maliciously inserted, similar bugs maliciously inserted would be devastating in case of sabotage.

A Framework for understanding whitebox design weaknesses was developed in [8].This was aimed at understanding backdoors, in substitution attack settings, the saboteur arranges for a subverted algorithm to replace a correct one. These attacks are assumed to be

Corresponding Author**:** Ihama E.I., Email: ihama.ei@edopoly.edu.ng, Tel: +2347039404855

black-box. It means that the defenders only have an API access to the algorithm (whether correct or subverted). Some sabotage is directly built into public designs (e.g., EC DRBG), in these settings substitution attacks are inapplicable. This may also be referred to as `whitebox design weaknesses' as they can withstand knowledge and scrutiny of the design of cryptosystem.

We advocate research and experimentation on new design approaches for cryptographic standards. One approach that has seemingly worked well for low-level primitives is that of public design competitions such as those conducted by NIST. This forces public review, allows a number of small design teams unfettered during design by large committees, and ultimately appears to have yielded strong primitives such as AES and SHA-3. Whether this can work for the more nuanced setting of higher-level cryptographic systems is an interesting open question. A key challenge we foresee will be specifying sound requirements. Even with committee designs, we might begin including in existing standardization processes an explicit review step for resilience to sabotage (perhaps helped by using the outcomes of the hoped-for models).

Software engineering for cryptography implementation of good standards securely, remains a monumental challenge. While some notable work has been done on implementation security, such as verified cryptographic implementations [9] and other tools for finding (benign) bugs or well-known problems [10], there seems too little relative to its importance.

In a nearer term, it is also relay by the following pragmatic suggestions from [11]. Firstly, vendors should make their encryption code public, including the specifications of protocols. This will allow others to examine the code for vulnerabilities. While it is true we won't know if the code we're seeing is the code that's actually used in the application, it may forces saboteurs to surreptitiously substitute implementations.

This raises the bar and forces the owner of the system to outright lie about what implementation that is being use. All this increases the number of people required for the sabotage conspiracy to work. The community should target creating independent compatible versions of cryptographic systems. This will help to check that if any individual is operating properly.

Cryptographic objectives

All encryption systems should ensure that it contains a means of adding and ability to transmit information across at network as highlighted in the objectives of cryptographic scheme [12]. These purposes are focused into five fundamental objectives, such as:

Privacy or Confidentiality: It is an element which guarantees that it is only the expected user that gets the information which is been transmitted.

Authentication: It is a means of checking the character of the transmitter and receiver before communicating with the cryptographic scheme.

Non-repudiation: It is a means use to ensure that the sender of a message is actually who sent the message and neither sender nor recipient could deny about message being sent by them.

Types of Cryptographic Scheme

Cryptographic systems are divided into two major types [13],these are the Symmetric Key Encryption and Asymmetric Key Encryption. The Symmetric Key: It is an encryption system where both the sender and the receiver share an indistinguishable key. It is also referred to as secret key encryption. This scheme could be implemented each utilizes square character system or stream character scheme. Square Cipher executes encryption check by-square of plain message though stream character executes encoding character-by-character [14].

Some examples of symmetric key encryption are AES
(Advanced Encryption Standard) and DES (Data Encryption Standard).

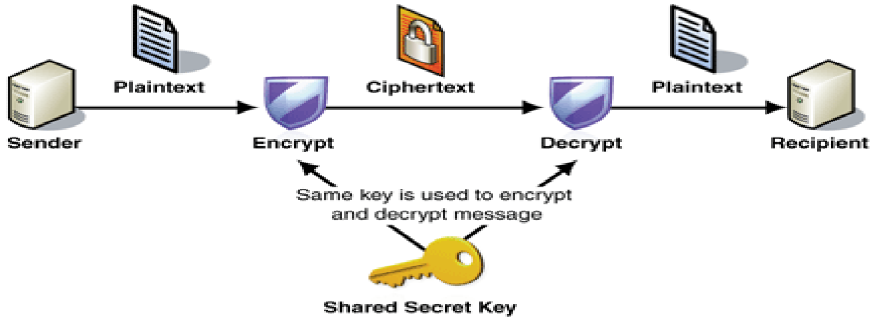


Fig1: Symmetric Key Encryption

Asymmetric Key Encryption: - In symmetric key, the key must be conveyed securely to all collectors and sender without being broken which appeared to be extreme errand. To beat the impediment of symmetric key sort, lopsided key encryption was created. Here, two keys are utilized for cryptography process [15]. Open key is utilized for encryption which is known to everybody and private key is utilized for decoding which is known distinctly to client. This disposes of the requirement for sharing keys. It is otherwise called open key encryption [16].

Some examples of asymmetric key encryption are DeffieHellman, DSA (Digital Signature Algorithm) and ElGamal etc.

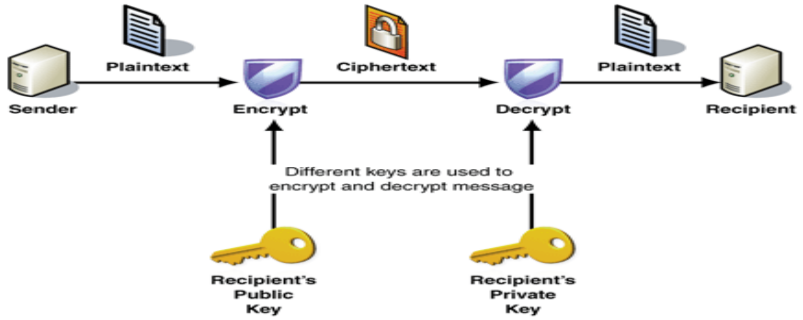


Fig2: Asymmetric Key Encryption

The Hashing Process

Hash functions, also called message digests and one-way encryption, are algorithms that, in some sense, use no key , Instead, a fixed-length hash value is computed based upon the plaintext that makes it impossible for either the contents or length of the plaintext to be recovered. Hash algorithms are typically used to provide a digital fingerprint of a file's content, often used to ensure that the file has not been altered by an intruder or virus. Hash functions are also commonly employed by many operating systems to encrypt passwords. Hash functions, then, provide a measure of the integrity of a file.

The key in public-key encryption is based on a hash value. This is a value that is computed from a base input number using a hashing algorithm. Essentially, the hash value is a summary of the original value. The important thing about a hash value is that it is nearly impossible to derive the original input number without knowing the data used to create the hash value.

Table 1.Hashing Process

Input number	**Hashing algorithm**	**Hash value**
10,667	**Input # x 143**	**1,525,381**

You can see how hard it would be to determine that the value 1,525,381 came from the multiplication of 10,667 and 143 from table 1 above. But if you knew that the multiplier was 143, then it would be very easy to calculate the value 10,667. Public-key encryption is actually much more complex than this example, but that's the basic idea.

The important thing about a hash value is that it is nearly impossible to derive the original input number without knowing the data used to create the hash value. In order to ensure that encrypted information is not easily decrypted by cryptanalyst, the system uses an alphanumeric N key system, which makes the system quite unique.

Cryptanalysis

Cryptanalysis is the process of defeating the work of cryptography. This word originated from Greece where kryptós stands for "hidden" and analýein means "to loosen" or "to untie" [17]. It is use to intrude or breach the cryptographic system with or without knowing the secret key of the process.

Cryptanalysis procedure is a process of crushing crafted system by cryptography. This word begun from Greece where kryptós means "covered up" and analýein signifies "to release" or "to loosen" [18]. It is use to interfere or rupture the cryptographic framework with or without knowing the mystery key of the procedure.

(a) Plain Text: It is the secret or private data to be protected while being transmitted.

(b) Cipher Text: It is the transmuted and transformed plain text which is not readable while merely looking at it.

It is gotten by applying an encryption algorithm and encryption key over the plaintext. It could feasibly be protected. On the off chance that it's not protected or being monitored, at that point any interloper can get to it effectively from the general population channel utilizing it while it is being transmitted. In any case, translating it without realizing the mystery key is an extreme errand.

(c) Encryption Algorithm: It is a scientific bit by bit process utilized for changing a plain content into its figure of content dependent on some encryption key. Various instances of such calculation are AES, DES, blowfish and snake etc. It is utilized next to sender [19].

(d) Decryption Algorithm: It is actually the turnaround numerical procedure utilized by using encryption calculation. It takes the content of the figure and decodes its key to deliver a unique plain content, it utilizing the next to receiver approach.

(f) Encryption Key: This key is a value that leads part of the cryptographic framework, it is either known distinctly to the sender or to both sender and collector. Safe guarding of this key is critical for making cryptographic framework fruitful. This key is applied inside encryption calculation to produce the figure messages out of the plain content[20].

(g) Decryption Key: This key is the worth known to collector and it might possibly be indistinguishable from encryption key. It is applied inside the decoding calculation to produce the plain content once again from the gotten figure content. An assortment that contains all conceivable decryption keys is known as Key Space.

Some Security Methods

Some security measures to overcome or undo the attacks on networks are different technologies, used in recent times; some of the major techniques are given below [21].

(a) Authentication: All data and documents received must be authenticated if they are sent by a trusted sender or not. They must also be checked for unwanted breaching or alterations within data.

(b) Antivirus: Antivirus software must be installed and updated on regular time intervals, also network and systems checks must be conducted regularly.

(c) Firewalls: This software keeps tracks of inward and outward traffic of any system; it also informs the user about unpermitted access and usage.
(d) Access Control: Each user must have their particulars like, username and password, so that only intended users may login.
(e) Cryptography: It is a technique of encoding plaintext into cipher text before transmitting it over a channel to avoid the stealing of the confidential data.

The Encryption Process

The diagram below shows the encryption and decryption process using cryptography. The Readable text R is encrypted with an encryption algorithm system and an encryption key F. The resulting ciphertext,(C), is transmitted over the network. The receiver decrypts the ciphertext (C) with a decryption algorithm system and a decryption key G. The encryption and decryption algorithms are public information. However, at least one of the keys (F and G) is private information. The keys consist of a relatively short string of bytes (e.g., 128 bits). The longer the key, the more difficult to break the cipher.

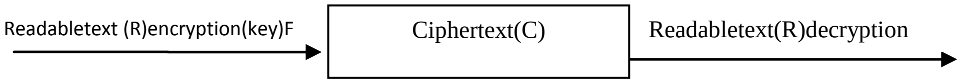


Fig. 3.The Process of Encryption.

This cryptography software is both online and offline based systems. That is they are applicable or useful in the protection of Internet files or systems within a network, and also protect an information systems both in storage and when transmitted across an unsecured network like the internet. The system is user friendly. It is design to provide features, which provides users with input screen such that a user can enter his/her access code to login to the system.

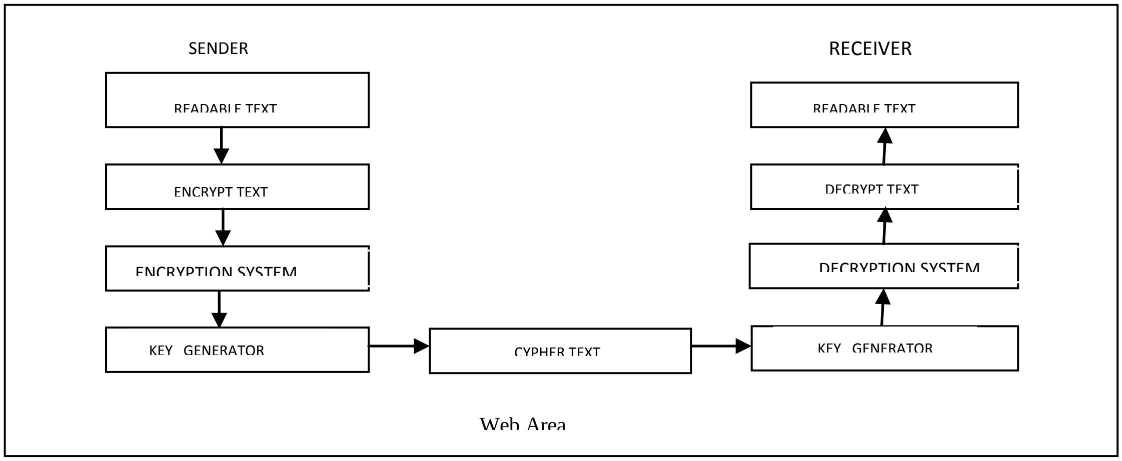


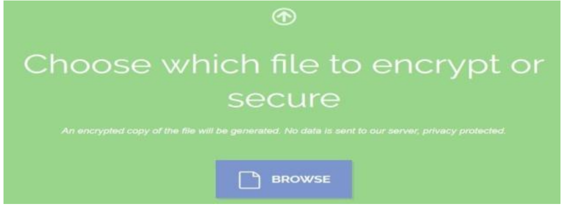
Fig 4: The Architecture of the Data Security System Using Cryptography Techniques

The Design Approach of the data protection system

The architecture above ensures that a sender uses an encryption system to encrypt their data (information).

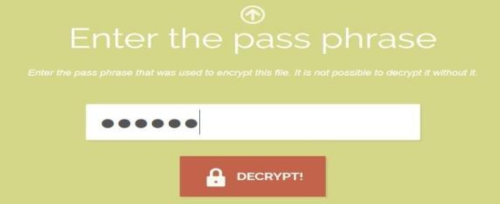
The Encryption Process: in the encryption process the readabletext is encrypted by the sender, by supplying the system with the key(alphanumeric), the processing task of the encryption process, is a combinatory approach, a transposition and substitution ciphers algorithms is adopted for the processing task of the encryption of the users information (data), which transposes the readabletext into a ciphertext by the encryption system and the key generator. This is done by supplying N (multiple) key/s of 256bits. These keys are use by the encryption system to encrypt the sender's information (data), this renders it into an unreadable form known as ciphertext. The Cyphertext: this is a transposed plaintext, using the encryption system and the key generator (which is a value that is computed from a base input number using a hashing algorithm).

Software Applications
Screen shot of the application process


Choose which file to encrypt or secure

An encrypted copy of the file will be generated. No data is sent to our server, privacy protected.

Fig5. Encryption process


Enter the pass phrase

Enter the pass phrase that was used to encrypt this file. It is not possible to decrypt it without it.

●●●●●●

Fig 6. Decryption process

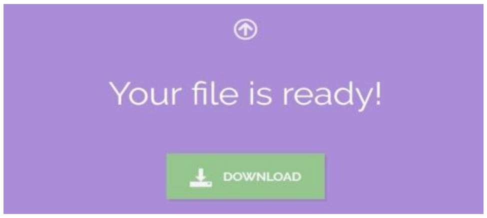
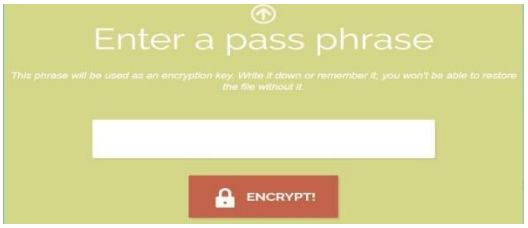
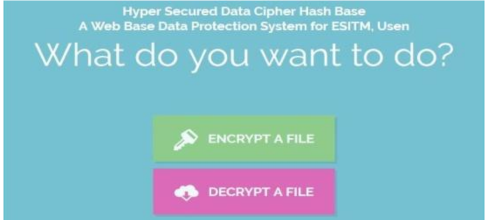

Your file is ready!

Fig 7. Downloading process


Enter a pass phrase

This phrase will be used as an encryption key. Write it down or remember it; you won't be able to restore the file without it.

Fig 8. Encrypting your file


Hyper Secured Data Cipher Hash Base
A Web Base Data Protection System for ESITM, Usen

What do you want to do?

Fig 9. Decrypting your file

CONCLUSION

In this paper, we looked at some cryptographic schemes, types and aims; how cryptography works and guarantees data (information) protection, how it ensures, prevents authorized users from transferring or gaining access to the information.

Information (data) security when properly sustained by diverse techniques like Cryptography, like watermarking, digital signatures, firewalls, access controls and steganography etc. The security nature of cryptographic systems makes it popular among data protection schemes. This makes cryptography as an evolving technique in safeguarding our confidential information. A strong data protection scheme system was designed and implemented for Edopoly to prevent authorized and malicious attack from unauthorized users from gaining access to our systems files and information systems, which can be costly to the institute and other information systems users, this systems provides both online and offline protection measures, by visiting this site: www.edopolyendencrypt.com.ng to download the software.

We seek all information systems users in the Institution to use this software in protecting their files and information systems from unauthorized users or intruders.

Reference

[1] Rajesh R Mane, (2015), "A Review on Cryptography Algorithms, Attacks and Encryption Tools", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 3, Issue 9.

[2] A. Albertini, J. P. Aumasson, M. Eichlseder, F. Mendel, and M. Schl□a_er. Malicious Hashing: Eves Variant of SHA-1.In Selected Areas in Cryptography (SAC) 2014.

[3] MihirBellare, Kenneth G Paterson, and Phillip Rogaway, (2014) Security of symmetric encryption against mass surveillance. In Advances in Cryptology{CRYPTO, Vol. 19, pp. 1.

[4] Stephen Checkoway, Matthew Fredrikson, Ruben Niederhagen, Matthew Green, Tanja Lange, Thomas Ristenpart, Daniel J. Bernstein, Jake Maskiewicz, and HovavSchacham. On the practical exploitability of Dual EC in TLS implementations. In USENIX Security Symposium, 2014.

[5] NIST National Vulnerabilities Database. Vulnerability summary for CVE, (2014) Available at http://web.nvd.nist.gov/ view/vuln/detail?vulnId=CVE-2014-1266.

[6] NIST National Vulnerabilities Database. Vulnerability summary for CVE, (2014). Available at http://web.nvd.nist.gov/ view/vuln/detail?vulnId=CVE-2014-0224.

[7] Tyler Nichols, Joe Pletcher, Braden Hollembaek, Adam Bates, Dave Tian, Abdulrahman Alkhelai_, and Kevin Butler, (2014)CertShim: Securing SSL certi_cateveri_cation through dynamic linking. In ACM Conference on Computer and Communications Security (CCS), ACM.

[8] MihirBellare, Kenneth G Paterson, and Phillip Rogaway, (2014) Security of symmetric encryption against mass surveillance. In Advances in Cryptology (CRYPTO), Vol.19, pp.1.

[9] Jose Bacelar Almeida, Manuel Barbosa, Gilles Barthe, and Francois Dupressoir, (2013) Certified compute raided cryptography: Efficient provably secure machine code from high-level implementations. ACM Conference on Computer and Communications Security (CCS), pp. 1217-1230.

[10] Lin-Shung Huang, Alex Rice, ErlingEllingsen, and Collin Jackson, (2014), Analyzing forged SSL certificates in the wild. IEEE Symposium on Security and Privacy. IEEE.

[11] Bruce Schneier, (2013), How to Design And Defend Against The Perfect Security Backdoor.https://www.schneier.com /essays/archives/2013/10/how_to_design_and_de.html

[12] N.Lalitha,P.Manimegalai,V.P.Muthukumar,M.Santha, (2014) "Efficient data hiding by using AES and advance Hill cipher algorithm ", International journal of research in computer applications and Robotics, volume 2.

[13] Vikasagarwal, (2014) "Analysis and Review of Encryption and Decryption for Secure Communication", International Journal of Scientific Engineering and Research IJSER), pp. 2347-3878, Vol. 2, Issue 2.

[14] PranabGarg and Jaswinder Singh Dilawari, (2012) "A Review Paper on Cryptography and Significance of Key Length", International Journal of Computer Science and Communication Engineering, IJCSCE Special issue on "Emerging Trends in Engineering" ICETIE.

[15] RituPahal, Vikas Kumar, (2013)," Efficient implementation of AES", International journal of advanced research in computer science and software engineering, Vol. 3, issue 7.

[16] DivyaSukhija, “A Review Paper on AES and DES Cryptographic Algorithms”, International Journal of Electronics and Computer Science Engineering, Vol. 3 pp. 354-359.

[17] https://en.wikipedia.org/wiki/Cryptanalysis.

[18] Daniel Bleichenbacher, (1998) Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS#1. In Advances in Cryptology{CRYPTO, pp. 1. Vol.12.

[19] Dag Arne Osvik, Adi Shamir, and EranTromer. Cache attacks and countermeasures: the case of AES. Cryptology (CT-RSA), pp. 1. pp.20. Springer, 2006.

[20] Ivan Damgard. (2000), Efficient concurrent zero-knowledge in the auxiliary string model. In Advances in Cryptology (EUROCRYPT), pp.418 -430.

[21] MihirBellare, ZvikaBrakerski, MoniNaor, Thomas Ristenpart, Gil Segev,HovavShacham, and Scott Yilek, (2009), Hedged public-key encryption: How to protect against bad randomness. Advances in Cryptology (ASIACRYPT), pages 232-249.