International Journal of Emerging Engineering Research and Technology
Volume 2, Issue 4, July 2014, PP 78-84
ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

A Secure Data Transfer Algorithm for USB Mass Storage Devices to Protect Documents

Mr. A. N. Magdum
KIT's College of engineering
Kolhapur Maharashtra
aditya.magdum112@gmail.com

Dr. Y. M. Patil
KIT's College of engineering
Kolhapur Maharashtra
ymp2002@rediffmail.com

Abstract: *The Universal Serial Bus (USB) has become the most popular interface standard for hardware connection, and there has been a huge growth in the number of USB peripheral devices. External USB storage devices, in particular, are the most popular applications in market. Unfortunately, because USB affords high speed data transmission and is extremely convenient to use, many companies have prohibited the use of USB devices to prevent confidential data theft from computer systems via USB ports. However, this compromises the convenience of the USB connection. Therefore, finding a way to take in to account both the convenience of user and secure environment in company has become a significant issue. In this paper we propose a secure control algorithm which provides mutual authentication and key agreement between client and server to avoid this problem.*

Keywords: *Key agreement, Mutual authentication, Storage device, Universal Serial Bus (USB)*

1. INTRODUCTION

The rapid development of computer and information industry in recent years has produced various new peripheral devices that can connect to a business computer terminal, especially via a USB port, which has become a standard component of current computer hardware due to its convenience and ease of connectivity. The main advantage of the USB port is that it offers a single interface connection between the computer and various devices, such as hard disks, flash drives, printers, keyboards. Due to the evolution in computer science and technologies, information or private data for individuals and companies have been rapidly increasing in this modern society. Accordingly, there has been a requirement to plan in advance to prevent many problems related to the disclosure of the information. Meanwhile, the USB memories have become prevalent as personal storages for individuals, enterprises and governments thanks to their portability and accessibility. The portability of the memories sometimes can result in critical information leakage when they are lost, stolen or hacked because they usually store many kinds of important data. Due to this reason, various security-incorporated USB memories have been developed.

However, most of the commercial secure USB memories and their application software are proved through a sequence of analytical tests to have considerable vulnerabilities that result in exposing important information. These vulnerabilities can cause the secure USB memories to be easily compromised even though they are equipped with their own user and device authentication protocol

2. RESEARCH PROBLEM

The USB flash drive is a storage device that consists of NAND-type flash memory and integrated with USB interface. It is typically small, lightweight, portable and rewritable. But the various problems related security has been occurred as it is used widely. Nevertheless most USB flash drives do not include the security mechanism; an attacker can easily acquire the private information in the USB flash drive. USB has high transmission speeds and is so convenient, using one is like passing data through an unprotected gate, which expose confidential business data to the risk of theft. Therefore, effective control of these storage devices has become a significant issue for information security. If USB usage is comprehensively banned, then physical transmission between computers for various computer peripherals will be chaotic as before. If USB usage is not banned, then confidential data probably will be lost. Thus, balancing the convenience of the USB port with its lack of security is the most difficult USB management issue; confidential data must be secure, but access to USB devices must not be infringed upon. There is the need to find middle ground for this research.

In this paper, we have proposed a control algorithm which implements user authentication and key agreement to effectively govern file transmission via the USB port. Also to provide additional security we have proposed a system which will encrypt/ Decrypt the data to be transmitted via USB.

Problem definition:

Design and implement a two layered secure data transfer algorithm to securely store the documents in USB mass storage device.

3. RELATED WORK

There are few papers which gives detail security analysis of USB storage devices. [1] The protocol employs a remote authentication server to verify legal users and uses the Diffie-Hellmen key technique to protect the privacy of a file transmitted to a storage device. They have further proved that this protocol can resist some general attacks. In terms of protocol communication costs, realizing mutual authentication requires only two rounds of communication sessions. Therefore, this protocol provides an effective control protocol for USB storage devices which is both secure and efficient. Research in [6],[9] gives different attacks on USB devices and detailed a number of practical and theoretical attacks related to the mechanical, electrical, and software aspects of the USB keys. These attacks are not meant only for USB keys and could be expanded upon and attempted on other products. There are flaws in the existing USB hardware tokens on the market today, and users must recognize the security risks and benefits of each tool before it is recommended and implemented into their infrastructure. Some of these flaws can be worked around, but only after the weaknesses have been identified. It is important for designers of hardware devices, especially security products, to fully understand the threat model of their particular product before implementing a solution. In order to overcome the vulnerabilities mentioned in [6],[9] different methods of user authentication were proposed in research papers [2],[10], Which describes different authentication protocols for three different types of commercial secure USB memories. Even though these products tried to make themselves secure, the dedicated authentication protocols and the implemented software are not complete and many vulnerabilities can be utilized at some possible attacks. USB flash drive without any security function causes the exposure of private information. So new USB flash drive supported security function was invented to compensate for the problem. In this paper, authors analyze vulnerability of 6 famous secure USB flash drives, and demonstrate that password can be exposed on communication between the secure USB flash drive and PC. Also we show the vulnerability on the data recovery and the S/W bug of secure USB flash drive. Research in [6] suggests that after analyzing the vulnerabilities of secure USB flash drives. Authors found four vulnerabilities in the products. The first is the password exposure in communication between the secure USB flash drive and the security program. The second is that an unauthorized user is able to recover data in the secure USB flash drive after formatting. The third is the S/W bug of security program. The last one is designing hardware without packaging. To solve these problems, we propose the several solutions such as using hash function, wiping technology, secure coding technique.

Finally, the required system for overall protection of data proposed in [2] which include Diffie - Hellman key exchange [7], remote user authentication [3]. The research in [2] also provides security from general attacks. But there are still a few vulnerabilities which could be exposed. To effectively prevent the theft of information via USB storage devices, we need a control algorithm which will provide user authentication [10], key agreement [7]. Now as a part of additional security we proposed data encryption / decryption algorithm [11]. This addition of security not only makes the environment secure but also efficient.

4. MATERIALS AND METHODS / OUR APPROACH

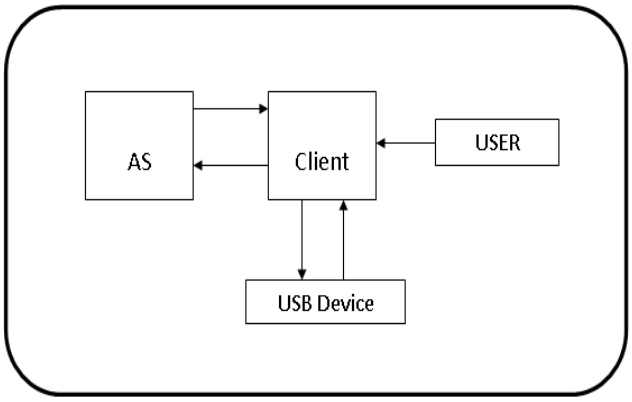
To manage a USB storage device effectively, any file transfer via a USB interface is restricted unless a user first passes a legal authentication procedure. A user wanting to transmit a file to a storage device via a USB interface must input a user name and password to verify legality. Then, the user is able to access the USB storage device.

In the user verification process, the user and the authentication server will generate a session key, which is used for encrypting all files transmitted to the storage device via the USB interface. Thus, all files stored on the USB storage device are encrypted. After the files are encrypted and transmitted,

USB access will be restricted until the next successful verification. If users want to decrypt the file on the USB storage device later, they must pass the same verification procedure and obtain the same agreement key to acquire the original file. This agreement key is established for each filename and user identity; different users or files generate different keys. In addition, the system will delete the agreement key temporarily saved on the user end after the file is encrypted/decrypted, thus ensuring key independence and system security.

This protocol has the following three characteristics:

1. Only users verified to be legal can access the USB storage device.
2. Even if a confidential file on a storage drive is stolen, the file cannot be decrypted without a key.
3. Even if a legal file owner wants to malevolently store confidential data on a storage device and distribute it to another person, the owner cannot obtain the corresponding agreement key for decryption as long as the authentication server suspends this user account. Consequently, the original file is secure.



Above is the system environment which includes client server pair, user and USB device. System will work according to following procedure,

1. Insert a USB device.
2. User will provide user ID and password.
3. Achieve mutual authentication with AS.
4. Acquire a key from AS for encryption.
5. Save an encrypted file in USB device.

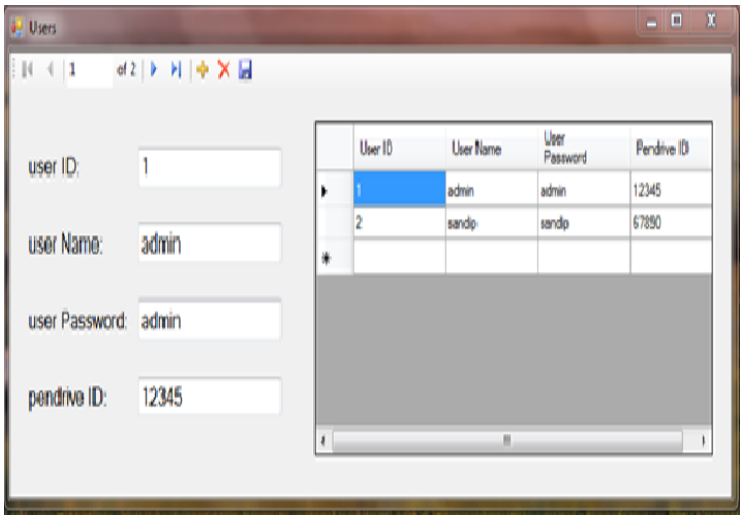
Even before accessing USB device user need to undergo the process of registration. Where, user needs to provide User ID and password. This set of ID and password will then communicated with authentication server (AS). Server will generate the mapping between ID – Password and USB device so; the process of registration for same USB devices will be avoided in future. In this registration phase different hash functions are used to provide better security.

Once the registration of the device is done it is ready to use for data transfer. While, doing so system will again undergo the process of verification. After verifying the authenticity of the user authentication server will generate a session key. This session key is generated using user name as well as the ID of the USB device so; the session key will be different for different files and/ or Users. The same key further used in the process of encryption and decryption. USB device stores the data in encrypted format. So it is very difficult to access the data in USB device without a session key. In order to generate a session key one has to undergo the process of verification. Further, advance encryption system (AES) is used to encrypt / Decrypt the data to be transferred to and from USB. This AES will provide more complexity to the protocol which is essential to avoid brute force attacks. Following are the different phases discussed further in detail,

4.1. Registration Phase

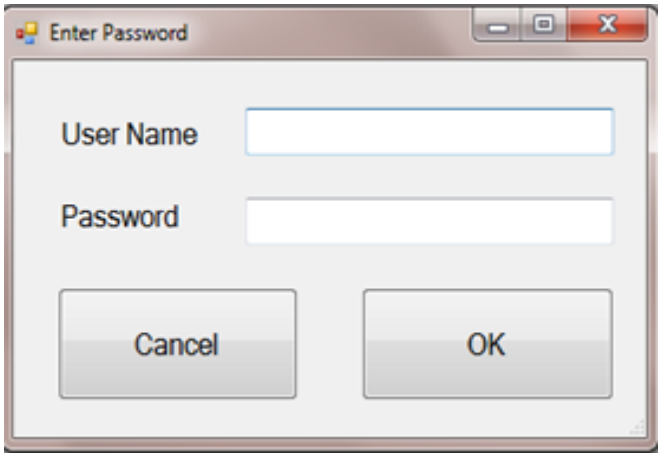
In this phase user needs to pass through registration before using the system. User needs to register user ID and password in the authentication server. To manage a USB storage device effectively, any

file via the USB interface is restricted unless a user first passes a legal authentication procedure. Following is the window where new user is registered.



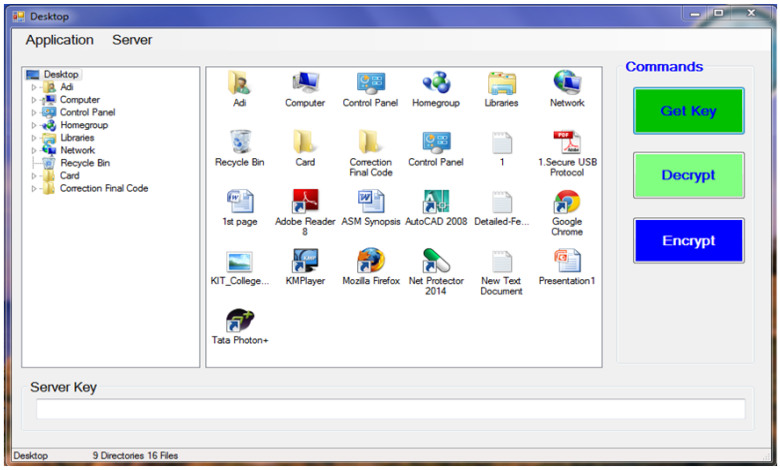
4.2. Verification and Key Exchange Phase

In verification process authentication server will verify that legality of the user. A user wanting to transmit a file to a storage device via USB interface must input username and password. Then, the user is able to access the USB storage device. After completing the registration phase, and when accessing the USB storage device, the user needs to achieve mutual authentication with the authentication server using the id and the password.



4.3. Data Encryption / Decryption Phase

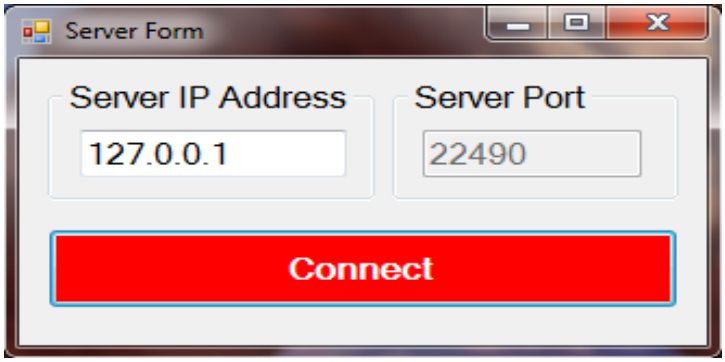
In this phase the session key generated in earlier phase is used to encrypt or decrypt the data to be transmitted via USB port.



Thus, all files stored on the USB devices are encrypted. After all files are encrypted and transmitted, USB access will restrict until the next successful verification. If user wants to decrypt the file on USB storage device later, they must pass the same verification procedure to obtain the same session key in order to acquire original file. This session key will be different for each user ID. To encrypt / decrypt the files, we use AES (Advanced encryption system) algorithm.

4.4. Client – Server Communication

The system environment is implemented using Client – Server communication. User has access to only client where as administrator has access to server. Client – Server communication also helps to monitor the entire process. Design of Client – Server communication form includes server IP address which will help client to connect to server. Also TCP port has to mention. Prerequisite of implementing client – server communication is to check both are in same network or connected to each other via web.



5. RESULTS

As defined in problem definition in chapter I, after execution of algorithm we are able to provide two layered security to the USB devices. In this section we are going to discuss the results of the algorithm. This algorithm is basically used to encrypt/ decrypt the files which are to be stored on the USB device. There are two types of files we consider while designing the algorithm text and image files. Algorithm works on both types of files. Further we are going to discuss results before and after execution of algorithm.

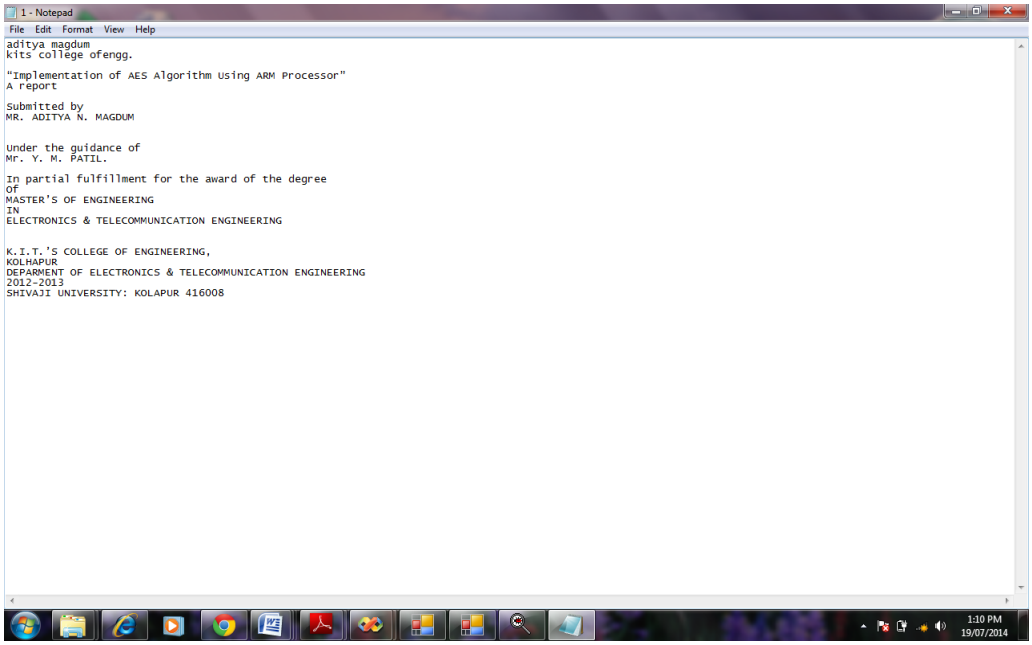


Fig. *File before Encryption*

Above figure shows a file (.txt) before encryption. The path of this file is selected on client and encryption is performed. Result of encryption is shown in fig. below which is a cipher text as we use the AES algorithm for encryption.

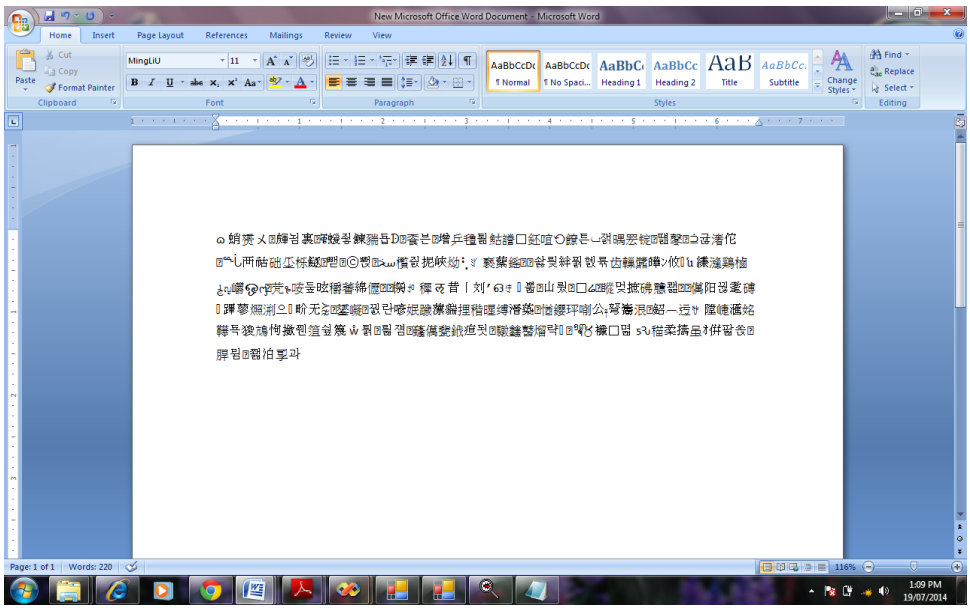


Fig. *File after Encryption*

6. CONCLUSION AND FURTHER WORK

In this paper, we have proposed a secure and efficient control algorithm to provide two layered security for USB devices. After analysing further we can prove that this algorithm can resist some general attacks. In terms of protocol communication costs, realizing mutual authentication requires only two rounds of communication sessions. Therefore, the proposed protocol provides an effective control protocol for USB storage devices which is both secure and efficient. We have designed this algorithm in visual studio. Further there is possibility to develop hardware which will suitable for the algorithm proposed in this paper.

ACKNOWLEDGMENTS

It gives me immense pleasure to acknowledge and thank many people who contributed in various ways for the successful completion of this paper. Words are inadequate to express my feelings while recording my deep sense of gratitude and respect to my guide Dr. Y. M. PATIL. The work presented in this paper could not have been accomplished without his inspiring guidance, constructive criticism and sustained encouragement.

REFERENCES

[1] Fuw-Yi Yang, Tzung-Da Wu, and Su-Hui Chiu, "A Secure Control Protocol for USB Mass Storage Devices", IEEE Transactions on Consumer Electronics, Vol. 56, No. 4, November 2010.

[2] Kyungroul Lee, Hyeungjun Yeuk. "Safe Authentication Protocol for Secure USB Memories", Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, volume: 1, number: 1, pp. 46-55, December 2010.

[3] Hyun sook Rhee, Jeong ok kwon, and Dong Hoon lee, "A remote user authentication scheme without using smart cards," computer standards & interfaces, Vol.31, No.1, pp. 6-13, 2009.

[4] Kangbin Yim, "A fix to the HCI specification to evade ID and password exposure by USB sniff," Proceedings of APIC-IST 2008, pp.191-194, December 2008.

[5] Tzung-her chen, wei-bin lee, "a New method for using hash function to solve remote user authentication," Computer & Electrical Engineering, Vol.34, No.1, pp. 53-62, 2008.

[6] Hanjae Jeong, "Vulnerability Analysis of Secure USB Flash Drives," Journal of the KIISC, vol. 17, No. 6, pp.99-118, December 2007.

[7] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, Vol. 22, No. 6, pp. 644-654,1976.

[8] C. P. Schnorr, "Efficient identification and signatures for smart cards," Journal of Cryptology, Vol. 4, pp. 161-174, 1991.

[9] Kingpin @stake, Inc.196 Broadway, Cambridge, MA 02139, USA "Attacks on and Countermeasures for USB Hardware Token Devices".

[10] Zhaohui Wang, Ryan Johnson, Angelos Stavrou "Attestation & Authentication for USB Communications".

[11] Aseem Jagadev, Vivek Senapati "ADVANCED ENCRYPTION STANDARD (AES) IMPLEMENTATION", may 2009.