Secure USB Bypassing Tool

By

Jewan Bang, Byeongyeong Yoo and Sangjin Lee

From the proceedings of

The Digital Forensic Research Conference

DFRWS 2010 USA

Portland, OR (Aug 2nd - 4th)

available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/diin

Digital Investigation

Secure USB bypassing tool

*Jewan Bang, Byeongyeong Yoo, Sangjin Lee**

Center for Information Security Technologies, Korea University, Seoul, Republic of Korea

Keywords:
Computer Forensics
Digital evidence collection
USB flash drive
Authentication bypass tools

ABSTRACT

As storage capacity increases due to development of flash memory techniques, use of USB memory has increased. As use of USB memory increases, violations of privacy and company confidentiality and technical information leakage occur more often. In this context, use of USB memories that provide security functions to protect the data in them is increasing. Most USB memories are equipped with basic security functions offered by the USB flash drive controller (hereafter called the "USB controller"). However, USB-controller-based security functions have several vulnerabilities. This paper explains how security functions can be bypassed using USB controller commands and presents the design and implementation of a secure USB bypassing tool that bypass the USB security functions.

1. Introduction

The development of digital technology is making storage devices large in capacity, small in size, and low in unit price. Presently, small storage devices are widely used in daily life, and among these, NAND flash memory devices using the USB (universal serial bus) interface (hereafter called "USB memory") are the most widely used. USB memories offer a relatively large storage space (64 MB–64 GB) in a physical size smaller than a floppy disk.

However, because of their portability and large capacities, which are their greatest attractions, USB memories are being used in crimes like breaches of company confidentiality and leakage of technical information. In this context, more emphasis is now being given to the importance of USB memories in the course of obtaining evidence during digital forensic investigations (Carvey, 2007).

In an effort to protect against violations of individual privacy that can occur when USB memory is lost, it is becoming more common to protect data using the USB memory security function. The increase in use of USB security is effective in protecting individuals against leakage of private information, but creates difficulties in obtaining evidence during digital forensic investigations. The reason is that evidence data can be hidden using the security functions of USB memory, and even if the data are known to be hidden, evidence can be obtained only after the suspect's cooperation has been obtained or a security function bypassing process has been executed. Therefore, if the suspect exhibits an uncooperative attitude when evidence data need to be obtained from secure USB memory, it should be determined whether the security function has been used, and if it has, a process to bypass it is required.

Most USB memories are equipped with a security function using USB controller commands. Because the USB controller security function is easy to use, it is liable to be used for USB-memory-related crimes. However, some USB controller security functions introduce vulnerabilities during the user certification process. Through this process, user certification can be bypassed, and data can be accessed. This paper explains secure USB types, certification methods, and bypassing plans, and presents a design and implementation for a secure USB bypassing tool that can bypass the security function of a secure USB using USB controller commands.

* *Corresponding author*.
E-mail addresses: jwbang@korea.ac.kr (J. Bang), pinpanel@korea.ac.kr (B. Yoo), sangjin@korea.ac.kr (S. Lee).
1742-2876/$ – see front matter
doi:10.1016/j.diin.2010.05.014

2. USB security

A secure USB offers a function that blocks an uncertified user from accessing data on the USB and allows use of the data only from a designated PC. Moreover, a secure USB provides various functions like IP tracking that can check the location information of a secure USB user. In general, a secure USB is divided into a non-secure domain with storage space that can be accessed by anyone and a secure domain with storage space that can be accessed only by the designated user.

The security functions of a secure USB include methods to mount hardware for security functions, to certify a user or encode or store data, and to provide security functions using the USB controller commands.

Security hardware is mounted by installing an encoding chip inside and encoding data. In general, AES 128-bit encoding is used, and a strong security function is offered so that if a password is entered incorrectly with more than a certain frequency, the internal security circuit is destroyed or another certification method like fingerprint recognition is used. However, such a system has shortcomings like high price because of the separate hardware mounting.

Security software is a security product which is produced so that it may be used for multiple purposes regardless of USB manufacturer or controller. A user is certified or data are encoded or decoded through internal software commands rather than through the commands built into the USB controller.

The security functions which use USB controller commands, a method of producing security software using the SDK (software development kit) provided by the USB controller manufacturer, are the major topic of analysis in this paper. Data are protected by hiding the partition in the security domain or controlling access as well as by encoding data. Because most USB controllers support security functions and because software based on these functions is distributed for free, users can use them with no difficulty.

3. Secure USB certification vulnerability analysis

USB memory sends data to and receives data from a computer using a device endpoint and a pipe. A device endpoint, a place where data are created or consumed, may be considered as a buffer which stores sent or received data. The pipe serves as a channel to provide and receive data in packet form, in practice by connecting the PC with a device endpoint. The pipe is divided into two parts: the message pipe and the stream pipe. The message pipe sends previously defined commands in packet form, and the stream pipe sends general data (Axelson, 1999).

To detour around the USB security functions, packets transferred between USB memory and the computer are analyzed. Normally the security functions and the relevant packets are sent and received in plain text. In this case, through analysis of sent and received packets, the password can be obtained, and the security function can be cancelled. If the security function and relevant packets are sent and received in code instead of in plain text, obtaining the password may be more difficult, but nevertheless, through packet analysis, the security function can be cancelled.

Table 1 – Status of Market Shares of USB Controllers in 2007 (Unit: Million Dollars)(2007 USB, 2008).

Order	Manufacturer	Market share	Profit
1	Phison	35.5%	$32.3
2	Silicon Motion (SMI)	23.2%	$21.1
3	Sandisk	14.9%	$13.6
4	Skymedi	9.0%	$8.2
5	Sony	7.4%	$6.7
6	AlcorMicro	3.2%	$2.9
7	Toshiba	3.1%	$2.8
8	Others	3.7%	$3.4
	Total	100%	$91.1

To cancel the security function of a secure USB, which is regarded as a vulnerability to monitoring of USB memory packets, a particular command should be transferred. To transfer commands to a USB or to receive certain information, the DeviceIoControl() API should be used. After receiving the handle of the present USB through the CreateFile() API, data can be sent to and received from the USB using the DeviceIoControl() API.

The commands used in a USB can be divided into common commands used in all USBs and vendor-specific commands that are programmed by the USB controller manufacturer to execute particular functions. The common commands are related to the functions which various USBs commonly execute, like READ and WRITE, and the commands specified by the controller manufacturer are defined in addition for the applicable USB and separately implemented in addition to the common commands. For example, vendor-specific commands used in USB instruments are dependent on the embedded USB controller. Therefore, even if USBs are manufactured by different manufacturers, if the embedded controllers are the same, the same command will be used for the same function, and the security function can be detoured around using the same method.

4. Secure USB certification detouring technique

Among secure USBs using USB controllers, those using three types of controllers, USBest UT163, Skymedi SK6281, and SMI SM321 have certification detouring methods which have

Fig. 1 – USBest UT163 security configuration condition check and password obtaining response message.

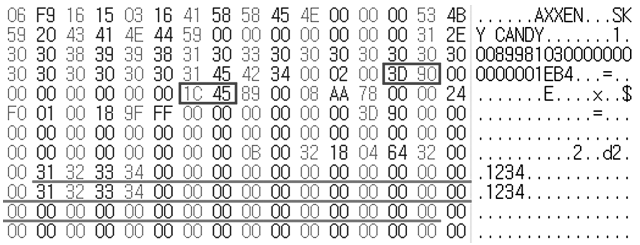

Fig. 2 – Skymedi SK6281 security configuration condition checking and password obtaining response message.

already been released (Lee et al., 2008). In addition to these three types of controllers, detouring methods for five other controllers, USBest UT165, Skymedi SK6211, AlcorMicro AU6983, Sandisk Cruzer Contour, and Toshiba, were researched. Table 1 indicates the market shares of several of these USB controllers:

The following section explains the password acquisition and security function detouring methods for each controller.

4.1. *USBest UT163*

The USBest UT163 controller can perform all of security configuration condition checking, password obtaining, and security certification detouring. Security configuration condition checking and password obtaining can look for the reply message for the command 0 × F8000000020000001000000 (12 bytes) after transferring to the USB. This response message has size 0 × 200. If the value of the dual offset 0 × 1E (two bytes) is 0 × 0101, a security domain exists, otherwise no security domain exists. The password and the password hint may be obtained at offsets 0 × 28 (8 bytes) and 0 × E0 (30 bytes) (Fig. 1).

The command for security certification detouring is issued as 0 × FC00000000000000100 (10 bytes), and the controller cancels the USB security function if it receives the applicable command.

4.2. *Skymedi SK6281*

The Skymedi SK6281 controller can perform all of security configuration condition checking, password obtaining, and security certification detouring. Security configuration condition checking and password obtaining can look for the response message after transferring the command 0 × D4000000001000000000 (10 bytes) to the USB. The response message has size 0 × 200. If the value of the dual offset 0 × 3D (2 bytes) and that of 0 × 46(2 bytes) are different, a security domain exists; if the values are the same, no security domain exists. The password and password hint can be obtained at offset 0 × 81 (16 bytes) and 0 × 91(30 bytes) (Fig. 2).

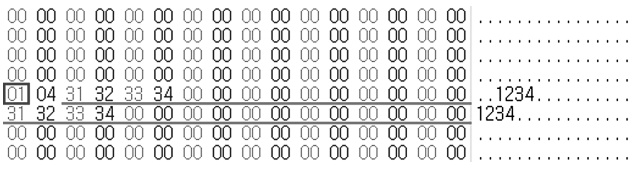

Fig. 3 – SMI SM321~SM325 security configuration condition checking and password obtaining response message.

Fig. 4 – Phison PS2136 security configuration condition checking response message.

When the security certification detouring command is issued as 0 × D8040000000000000000 (10 bytes), the controller cancels the USB security function if it receives the applicable command.

4.3. *SMI SM321 ~ SM325*

The SMI SM321 ~ SM325 controllers can perform all of security configuration condition checking, password obtaining, and security certification detouring. Security configuration condition checking and password obtaining can look for the response message after transferring the command 0 × F005000000000000000000100000000 (16 bytes) to the USB. The response message has size 0 × 200. If the value of dual offset 0 × 40 (1 byte) is not 0 × 00, a security domain exists, and if the value is 0 × 00, no security domain exists. The password and password hint can be obtained at offset 0 × 42 (12 bytes) and 0 × 50 (16 bytes) (Fig. 3).

When the security certification detouring command is issued as 0 × F11100000000000000000000100000000 (16 bytes), the controller cancels the USB security function if it receives the applicable command.

4.4. *Phison PS2136*

The Phison PS2136 controller can perform security configuration condition checking and security certification detouring only. Security configuration condition checking and password obtaining can look for the response message after transferring the command 0 × 0D0000000800000000000000 (12 bytes) to the USB. The response message has size 0 × 08. If the value of dual offset 0 × 00 (1 byte) is 0 × A0, a security domain exists, and if the value is different, no security domain exists (Fig. 4).

In the Phison PS2136 controller, security certification detouring goes through three stages. First, after transferring the command 0 × 0605494F464F00000000 (10 bytes) to the USB, the controller stores the value of offset 0 × 1FC (4 bytes) separately within the response message. Second, the controller replaces offset 0 × 1FC (4 bytes) by 0 × 00 within the command 0 × 06060100000000000000000 (12 bytes) and the response message from the first process and transfers them simultaneously. Third, the controller enters the value of offset 0 × 1FC (4 bytes) obtained during the first process in reverse order within the [ ] of 0 × 0D0000000802[00000000]000000000000 (16 bytes), the security certification detouring command (Fig. 5).

The Phison PS2136 controller, if the security function detours through certification, loses data from the general domain. Therefore, the data in the general domain should be duplicated in advance before security function certification detouring is performed.

Fig. 5 – Phison PS2136 security function detouring command.

Fig. 6 – USBest UT165 security configuration condition checking and password obtaining response message.

4.5. *USBest UT163 UT165*

The USBest UT163 controller can perform security configuration condition checking and password obtaining only. Security configuration condition checking and password obtaining can look for the response message after transferring the command 0 × F8000000040000000001000000 (12 bytes) to the USB. The response message has size 0 × 200. If the value of dual offset 0 × 1E (2 bytes) is 0 × 0103, a security domain exists, and if the value is different, no security domain exists. The password and password hint can be obtained at offset 0 × 28 (8 bytes) and 0 × E0(30 bytes) (Fig. 6).

The USBest UT165 controller transfers the password hash value that uses an independent algorithm along with the security cancellation command to cancel the USB security function. Because it is difficult to know the password hash value in advance, security function detouring is difficult. However, because the password can be found, the security function can be cancelled using the password.

4.6. *Skymedi SK6211*

The Skymedi SK6211 controller can do all of security configuration condition checking, password obtaining, and security certification detouring. Security configuration condition checking and password obtaining can look for the response message after transferring the command 0 × D4000000000100000000 (10 bytes) to the USB. The response message has size 0 × 200. If the values of dual offset 0 × 42 (2 bytes) and 0 × 64 (2 bytes) differ, a security domain exists, and if they have the same value, no security domain exists. The

Fig. 7 – Skymedi SK6211 security configuration condition checking and password obtaining response message.

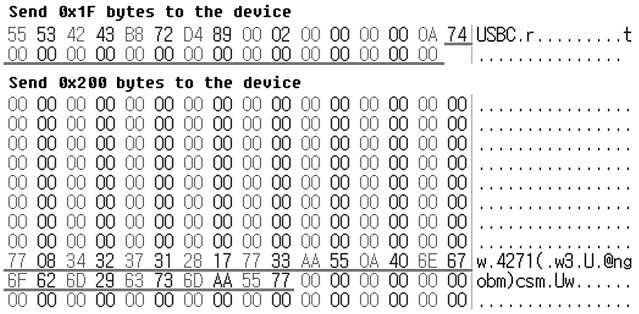

Fig. 8 – AlcorMicro AU6983 security function detouring command.

password and password hint may be obtained at offset 0 × 81 (16 bytes) and 0 × 91 (30 bytes) (Fig. 7).

When the security certification detouring command is issued as 0 × F1110000000000000000000100000000 (16 bytes), the controller cancels the USB security function if it receives the applicable command.

4.7. *AlcorMicro AU6983*

The AlcorMicro AU6983 controller can perform security configuration condition checking and security certification detouring only. Security configuration condition checking and security certification detouring can look for the response message after transferring the command 0 × 9C090000020000000000 (10 bytes) to the USB. The response message has size 0 × 200. If the entire response message has the value 0 × FF, a security domain does not exist, and if different values are present, a security domain exists.

The AlcorMicro AU6983 controller performs security bypassing in two stages. First, the controller obtains a 512-byte response message to check the security configuration condition. 0 × 80 (22 bytes) of the response message contain the password hash value used by the independent algorithm. Second, the controller transfers 512 bytes of the response message obtained in the first process simultaneously with 0 × 74000000000000000000 (10 bytes), the security certification detouring command (Fig. 8).

Because the AlcorMicro AU6983 controller uses the password hash value used by the independent algorithm as a password, it is difficult to obtain the password.

4.8. *Sandisk Cruzer Contour*

The Sandisk Cruzer Contour controller can perform security configuration condition checking and security certification detouring only. Security configuration condition checking can look for the response message after transferring the command 0 × FFA000000000000000000000000000000 (16 bytes) to the USB. The response message has size 0 × 10. If the values

Fig. 9 – Sandisk Cruzer Contour security configuration condition checking response message.

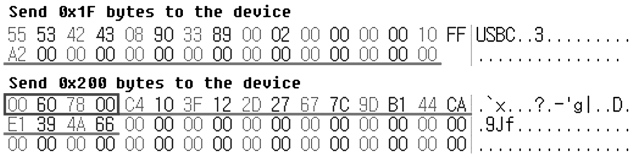

Fig. 10 – Sandisk Cruzer Contour security function detouring command transfer screen.

of the dual offsets 0 × 00 (4 bytes) and 0 × 04 (4 bytes) are the same, a security domain exists, and if they are different, no security domain exists (Fig. 9).

The Sandisk Cruzer Contour controller processes security bypassing through five stages. First, the controller transfers the command 0 × FF210000000000000000000000000000 (16 bytes) to the USB and obtains the value of offset 0 × 05(4 bytes) from the response message. Second, the controller substitutes the four-byte value obtained in the first process into the [ ] of 0 × [00000000]C4103F122D27677C9DB144CAE1394A66 (16 bytes) and saves it separately. Third, the controller transfers 0 × FFA2000000000000000000000000000000 (16 bytes), the security bypassing command, and the value saved in the second process simultaneously. Fourth, the controller transfers 0 × FF210000000000000000000000000000 (16 bytes), the last checking command (Fig. 10).

Because the Sandisk Cruzer Contour controller uses the password hash value that used by the independent algorithm as a hash, obtaining the password is difficult. In addition, if certification is detoured around using the security bypassing method previously described, the password is changed to an arbitrary password.

4.9. *Toshiba*

The Toshiba controller can perform security configuration condition checking and security bypassing only. Security configuration condition checking can look for the response message after transferring the command 0 × 1A003F000C0000000000 (10 bytes) to the USB. The response message has size 0 × 200. If the value of dual offset 0 × 25 (1 byte) is 0 × 00, a security domain exists, and if it is 0 × 01, no security domain exists (Fig. 11).

To detour around the security certification of the Toshiba controller, the current password should be modified to an arbitrary value. The Toshiba controller changes the password to the 20-byte hash value using an independent algorithm. In this paper, the certification detouring process will be explained using the value 0 × 8BDFA5B53A0F5485A8933C7961891F7D1894997B (20 bytes) which is

Fig. 11 – Toshiba security configuration condition checking response message.

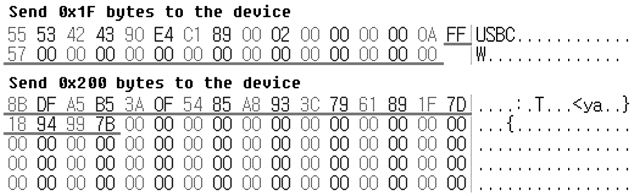

Fig. 12 – Toshiba security function detouring command transfer screen.

equivalent to '1234'. The certification detouring process has two stages. First, the process simultaneously transfers the password-modifying command 0 × FF570000000000000000 (10 bytes) and the value 0 × 8BDFA5B53A0F5485A8933C7961891F7D1894997B (20 bytes), the hash value equivalent to '1234'. This serves to change the current password to '1234'. Second, the process transfers simultaneously the security bypassing command 0 × FF540000000000000000 (10 bytes) and the hash value equivalent to '1234' (Fig. 12).

Whether the password is obtained, and whether the security certification can be detoured around (Table 2).

5. Design and implementation of secure usb certification detouring tools

In this research, a set of secure USB certification detouring tools was designed and implemented using the vulnerabilities of a secure USB and USB controller commands. A secure USB certification detouring tool can detour around the security function of a USB controller. Some of its specific functions include:

Table 2 – Whether the security function can be detoured around by various USB controllers.

Controller name	Security configuration condition checking	Whether password is obtained	Whether security certification can be detoured around
USBest UT163	O	O	O
Skymedi SK6281	O	O	O
SMI SM321 ~ SM325	O	O	O
Phison PS2136	O	X	O
USBest UT165	O	O	X
Skymedi SK6211	O	O	O
AlcorMicro AU6983	O	X	O
Sandisk Cruzer Contour	O	X	O
Toshiba	O	X	O

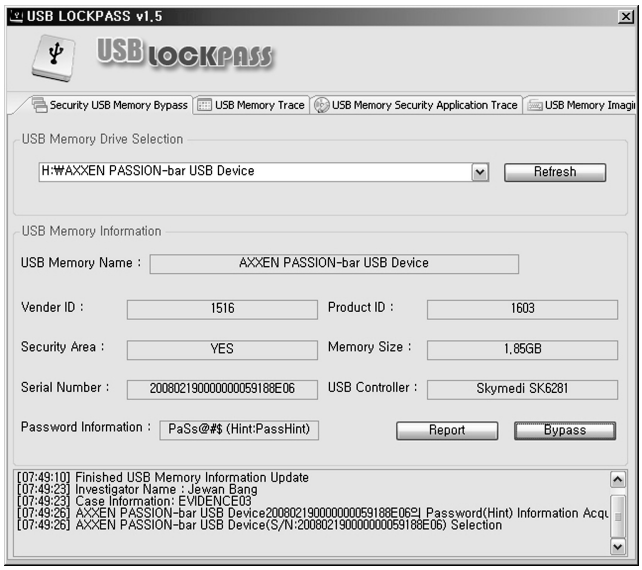

Fig. 13 – Certification detouring screen of the secure USB bypassing tool.

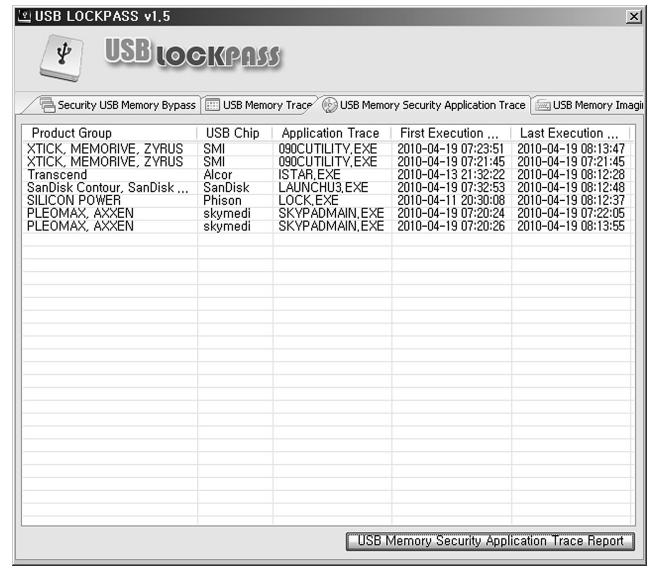

Fig. 15 – Trace analysis function of the secure USB bypassing tool's security cancellation tool.

- Secure USB security function canceling function
- Secure USB device information analysis function
- Secure USB password analysis function
- USB memory write protection function
- Function to analyze the information in the USB memory mounted in the intended system
- Function to analyze the traces of using the security canceling tool of the secure USB
- Disk image creation function
- Report composition function.

If the security function is active in an USB which has been obtained as evidence in a digital forensic investigation, the USB certification detouring tool can obtain data from this USB by detouring around the secure USB certification. In addition, this tool can create an image of the data obtained and compose a report on the data. In addition, the USB certification detouring tool can analyze secure USB device information and passwords and the information in the USB memory mounted in the target system using information saved in the registry(Carvey and Altheide, 2005). It also provides a USB memory write protection function to prevent corruption of the target USB memory during analysis. The USB memory's write protection function can activate this function by changing the configuration of the registry. If WriteProtectValue of "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\StorageDevicePolicies" is set to one, the write protection function is activated (Figs. 13 and 14).

Fig. 17 shows the structure of the USB bypassing tool. The tool is composed of three parts: the USB device recognition

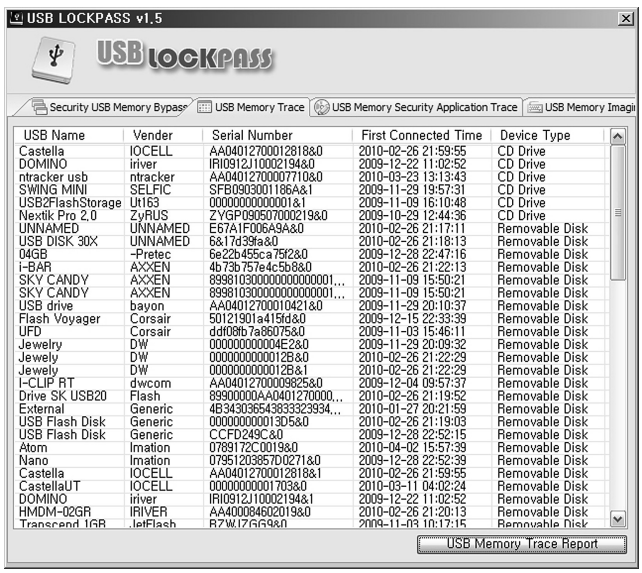

Fig. 14 – Trace screen of the secure USB certification bypassing tool.

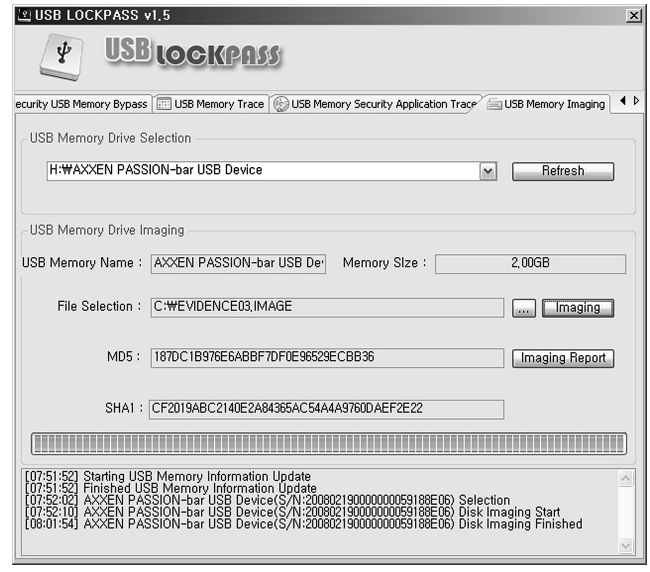

Fig. 16 – USB imaging screen of secure USB bypassing tool.

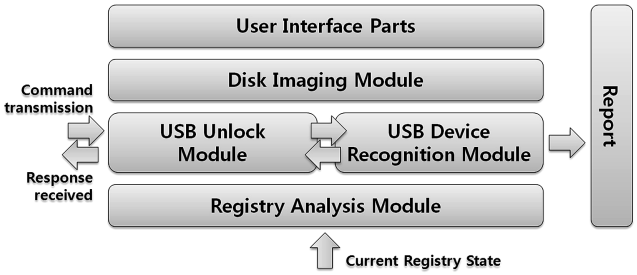

Fig. 17 – Structure of the secure USB bypassing tool.

module, the USB unlock module, and the registry analysis module.

The USB device recognition module analyzes and outputs basic information by recognizing the USB mounted in the target drive. The USB unlock module cancels the USB security functions based on the information transferred from the USB device recognition module and executes other operations such as password extraction. The registry analysis module reads the registry information of the target system and analyzes the information on any USB memory mounted in the system (Figs. 15–17).

6. Conclusions

In this paper, methods of bypassing USB security functions using USB controller commands have been explained, and the design and implementation of a secure USB bypassing tool that bypasses USB security functions has been described. As mentioned in the introduction, the use of USB memory has increased as USB memories have evolved toward larger capacities and small sizes. That means a greater importance for USB memory in the course of obtaining evidence in digital forensic investigations. Furthermore, if USB security functions are active in a USB obtained as evidence, the process of detouring around the security functions of a secure USB becomes more important to obtain evidence.

Therefore, in addition to the detouring method for USB controller security functions proposed in this paper, the security functions of new USB controllers to be launched in the future should be researched.

Acknowledgments

This research was supported by Bio R&D program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (20090084147).

REFERENCES

Carvey H. Windows forensic analysis. Norwell, MA, US: Syngress, ISBN 159749156X; 2007.

Axelson J. Usb complete: everything you need to develop custom usb peripherals. Lakeview Research; 1999.

Keungi Lee, Hyewon Lee, Changwook Park, Jewan Bang, Kwonyoup Kim, Sangjin Lee, USBPassOn: secure USB thumb drive forensic toolkit. In: The 2nd International conference on future generation communication and networking; Dec 2008.

2007 USB, Controller Market Shares(Revenue in Millions of Dollars), iSuppli Corp(Applied Market Intelligence); 2008.

Carvey Harlan, Altheide Cory. Tracking USB storage: analysis of windows artifiacts generated by USB storage devices. Digital Investigation Jun 2005;2(2).

Jewan Bang is a doctoral student in Graduate School of Information Management and Security at Korea University and a senior researcher of Digital Forensic Research Center in Korea University since 2009. He has performed projects related to Windows 7 forensics. His research interests are digital forensics, data recovery and reverse engineering.

Byeongyeong You received his B.S. degree in Computer Science from Sangmyung University, He is now studying master course in Graduate School of Information Management and Security, Korea University. He is currently working for Digital Forensic Research Center in Korea University. He has performed projects related to filesystem analysis and file carving. His research interests are digital forensics, filesystem analysis, file carving.

Sangjin Lee Received his Ph.D. degree from Korea University. He is now a Professor in Graduate School of Information Management and Security at Korea University and the head of Digital Forensic Research Center in Korea University since 2008. He has published many research papers in international journals and conferences. He has been serving as chairs, program committee members, or organizing committee chair for many domestic conferences and workshops. His research interests include digital forensic, steganography, cryptography and cryptanalysis.