Visual Monitoring System of Multi-Hosts Behavior for Trustworthiness with Mobile Cloud

Eun-Ha Song*, Hyun-Woo Kim* and Young-Sik Jeong*

Abstract—Recently, security researches have been processed on the method to cover a broader range of hacking attacks at the low level in the perspective of hardware. This system security applies not only to individuals' computer systems but also to cloud environments. “Cloud” concerns operations on the web. Therefore it is exposed to a lot of risks and the security of its spaces where data is stored is vulnerable. Accordingly, in order to reduce threat factors to security, the TCG proposed a highly reliable platform based on a semiconductor-chip, the TPM. However, there have been no technologies up to date that enables a real-time visual monitoring of the security status of a PC that is operated based on the TPM. And the TPB has provided the function in a visual method to monitor system status and resources only for the system behavior of a single host. Therefore, this paper will propose a m-TMS (Mobile Trusted Monitoring System) that monitors the trusted state of a computing environment in which a TPM chip-based TPB is mounted and the current status of its system resources in a mobile device environment resulting from the development of network service technology. The m-TMS is provided to users so that system resources of CPU, RAM, and process, which are the monitoring objects in a computer system, may be monitored. Moreover, converting and detouring single entities like a PC or target addresses, which are attack pattern methods that pose a threat to the computer system security, are combined. The branch instruction trace function is monitored using a BiT Profiling tool through which processes attacked or those suspected of being attacked may be traced, thereby enabling users to actively respond.

Keywords—Hardware Hardening, TPM, TPB, Mobile Cloud, System Behavior Monitoring, BiT Profiling

1. INTRODUCTION

Recently, there has been an increasing threat to the security of computer systems in this knowledge-information-based society. In other words, as hacking attacks increase more and more in various forms, a software attack on the existing security weak points has been increasing exponentially. In order to prevent such an attack, it is necessary to install security patches for the operating system or software constantly. The vulnerability of software-based security in existing system security mechanisms has grown due to varying hacking attacks. As a result, TC technology has emerged in order to complement the weakness of system security through hardware-based security, in addition to software based security. TC technology is to provide reliabil-
Visual Monitoring System of Multi-Hosts Behavior for Trustworthiness with Mobile Cloud

ity so that computers may consistently behave in their intended ways. Nonetheless, there is no technology to monitor TCG (Trusted Computing Group)’s TPM (Trusted Platform Module)-based computer system security on a real-time basis in a mobile device environment [1-3].

Accordingly, this paper proposes easy, fast, and intuitive monitoring functions for the notification of abnormal conditions and trust evaluation. This study develops $m$-TMS, which evaluates and monitors system conditions in a mobile-device environment, and it may operate in any environment where network services are provided in order to provide high reliability on a client level. Mobile device-based $m$-TMS monitoring supports the following technologies in a cloud environment regarding the trusted state of a TPM-chip mounted TPB (Trusted Platform Board). In other words, users may properly respond through the single monitoring of targeted computer systems, the multiple monitoring of targeted multiple computers, and branch instruction trace function through a BiT (Branch Instruction Trace Profiling Tool) Profiling tool [4-6].

Single monitoring provides monitoring of the current status of resources such as CPU, memory, process, user, virtual memory, the trusted state of a targeted computer system, and the monitoring of a branch instruction trace function through the BiT Profiling technique. For multiple monitoring, the IP of a targeted computer system is added. All functions of single monitoring may be applied to the added IP and the location where the monitored computer system operates is notified. Under multiple monitoring, a constraint of providing single monitoring to users is overcome and multiple computer systems may be rapidly monitored, which increases time efficiency.

The idea of the BiT originated from the fact that malicious execution traces may identified on an instruction level despite an increase in the variety of methods to attack systems from outside and that instruction level behaviors under program implementation are influenced by system module operation. In other words, BiT may detect the location of logical errors early on through a branch instruction trace of the program, enabling the user to actively cope with them.

The services provided by this cloud computing includes SaaS (Software as a Service) that provides applied software, PaaS (Platform as a Service) that provides software development

![Fig. 1. Relationship between $m$-TMS and Mobile Cloud System Environment for IaaS](image-url)
environment, and IaaS (Infrastructure as a Service) that provides the hardware resources of a computing system. IaaS is a service that provides storage devices through a cloud platform or computer capabilities through the Internet and it focuses highly on security. This service is provided through the web and therefore in case the reliability of security in computer storage devices and computer capabilities is lowered, all of its user’s information may be exposed. In this study, the user monitors on a real time basis the trusted state of IaaS, which uses a TPB-mounted computer system through m-TMS, and therefore may deal with malicious behaviors, thereby increasing system stability[7-8].

2. RELATED WORKS

The TCG is a not-for-profit organization that was formed in 2003 to develop, define, and promote open standards using reliability computing and security technologies based on hardware across multiple platforms or peripheral devices. The TCG that is proceeding with the entire standardization of trust computing defines TPM as a method to guarantee software-based reliability. A TPM provides storage domains that can protect data, keys, certificates, encryption engines such as RAS (named after its inventors, Ron Rivest, Adi Shamir, and Leonard Adleman), and Secure Hash Algorithm 1 (SHA-1). When each TPM chip is manufactured, unique keys—the endorsement key (EK) and storage root key (SRK)—are granted and they are kept within the chip and are not disclosed. By utilizing this trait of TPM, the weakness of software-based security methods may be complemented [9-10].

Basically, there exist three methods to monitor system behavior based on the TPM that includes virtual TPM, which is a software-based TPM emulator for Linux and monitoring systems [13-19]. A representative research of Virtual TPM is VMMs (Virtual Machine Monitors) [17], which are also known as hypervisors. VMMs allow multiple operating systems to simultaneously run on one machine. The basic responsibility of a VMM is to provide CPU time and memory and interrupts each VM. A VMM is a software layer underneath the operating system that meets these two basic requirements: (1) it provides a Virtual Machine (VM) abstraction that models and emulates a physical machine. (2) It provides isolation between virtual machines.

The objective of the TPM emulator is the implementation of a software-based emulator as well as an appropriate TCG device driver library [13, 15]. The emulator not only enable the implementation of flexible and low-cost test-beds and simulators but it also provides programmers of trusted systems with a powerful testing and debugging tool that can also be used for educational purposes. Because of its portability and interoperability, the TPM emulator runs on a variety of platforms (e.g., Linux, Mac OS X, and Windows) and is compatible with the most relevant software packages and interfaces. As it is entirely software-based, the TPM emulator can also be used to enhance virtual machines, thus enabling the execution of TPM-based software in a trustworthy virtualization environment. The TPM emulator also facilitates the evaluation of TPM extensions and firmware enhancements. In particular, it can be used to simulate new TPM commands and vendor extensions prior to their inclusion in a hardware specification or even prior to the real development process.

The idea of BiT originated from the fact that malicious execution traces may identified on an instruction level, despite an increase in the variety of methods for attacking systems from outside, and that instruction level behaviors under program implementation are influenced by sys-
tem module operation. The BiT measures runtime behaviors at the processor instruction level, thereby resolving a performance decrease in software-based security that can only cope with already known attack patterns or methods and that are incapable of adequate defense against new types of attacks. Unlike existing security methods through approaches to find the memory execution traces of control flow attacks, the BiT even performs the profiling of control data such as execution paths and branch targets and combines the converting and detouring of single entities such as a PC or target addresses. This makes it fairly hard to initiate damage to these items without modification [11-12].

The BiT monitors instruction traces and for processing selects a conditional branch in relation to each indirect branch instruction. Different standards for each indirect branch are defined for security purposes. The BPC (Branch Program Counter) is a linear address for indirect branch instructions. TPC (Target Program Counter) and EP (Execution Path) refer to the consecutive results (true or false) of conditional branches that lead to indirect branches indexed by BPC values. A program includes multiple indirect branches (multiple BPCs). These indirect branches follow different execution paths according to program execution. Therefore, the inclusion of multiple TPCs in each BPC is possible and multiple conditional branches (EP) result from indirect branch targets. The hierarchical structure is shown in Fig. 2.

Cloud computing is a service in which a user borrows as many as necessary computing resources via the Internet. It includes SaaS, which provides software as a service, PaaS, which provides software development environment as a service, and IaaS, in which users as many as necessary rent virtual server instances from cloud data centers. IaaS does not sell hardware to server users but only provides them with hardware computing capabilities. IaaS has made it possible to transcend the limit of storage devices and capabilities. Nonetheless, IaaS is vulnerable in terms of security from the server managers’ perspectives. Accordingly, this study intends to combine a computer system that operates based on TPM-mounted TPB with m-TMS that monitors the current status of the system’s resources and its reliability. A representative IaaS is Amazon Web Services’ Simple Storage Service and Elastic Compute Cloud (S3 & EC2).

3. DESCRIPTION OF TPB AND TPBM

TPB is a hardware board for the extension of TPM functions, including the design of the hardware board and TPB interface. First, the hardware board specifies the hardware components, extended components of TPM, and digital circuit. The TPB interface specifies the hardware interface, extended interface, device driver design, and API design. There are many technologies in TPB including: safe high capacity storage, boot code for the expansion of system BIOS, a
high performance encryption co-processor, and additional services for trusted computing. The high performance encryption co-processor supports a damper area for key storage, high-speed operation for the encryption data algorithm, and a hash function.

For improving trustworthiness with the TPM, there are three different issues: (1) program behavior verification, (2) the trustworthy computing platform, and (3) hardware technologies for the trustworthy platform board. The goal of program behavior verification is to develop a program counter (PC) encoding compiler and monitoring system for system behavior. The PC encoding compiler is the trusted compiler for hardening the control flow of the program and is the visual monitoring system for system behavior. The purpose of the trustworthy computing platform improves the existing virtualization technology and hardens the operating system with respect to the TPM chip for trusted computing. The goal is to monitor the safety of booting, the operating system, and application programs. A trustworthy platform board is a good complement to hardware support technology for verifying the reliability. That is, it improves the primitive functions of TPM.

The trustworthy computing platform has the following two aims for enhancing the system reliability: (1) improving the existing virtualization technology and hardening the operating system with respect to the TPM chip for trusted computing, and (2) monitoring the safety of booting, the operating system, and application programs. The trusted boot loader supports the root of trust for measurement and extends trusted areas of the kernel to a computing platform that is based on hardware. Environments for the reliable execution of applications can include local attestation and trusted areas of the kernel for applications. Local attestation is a storage technology of databases for the integration of program code and verification. This platform uses encryption of the whole disk and provides disk access authority to authentication users only. It also specifies the design of the TPB software stack, TPB service class, and TPB device driver library API specification. As shown in Fig. 3, we make a comparison between the software layer hierarchy of TPM and TPB [6].

However, there have been no technologies until today that enable a real-time visual monitoring of the security status of a PC that is operated based on the TPM. And the existing TPB has provided the function simply to monitor the status in real time for the system behavior of a sin-

![Fig. 3. Hierarchy of TPM and TPB](image)
Visual Monitoring System of Multi-Hosts Behavior for Trustworthiness with Mobile Cloud

gle client. Therefore, in this paper, we developed the trusted platform board monitoring (TPBM) system that provided the real-time visual monitoring function that enabled effectively detecting and managing abnormal phenomenon on the TPB as well as providing security. The TPBM includes the function to monitor the trust status and resource status of specific multiple clients in a visible way, which is conducted in real time. In addition, the main purpose of the TPBM is to provide the function to monitor system behavior of remote multi-client systems that are operated to strengthen the weak points of the existing security and to ensure hardware security. First of all, the TPBM system is developed to monitor system status evaluation in order to secure high reliability at the level of a single client by providing the function of easy, quick, and intuitive monitoring for abnormal status transmission and reliability assessment. Furthermore, the main purpose is to provide the function to monitor not only the status of each system resource and process but also the behavior of multiple clients in the cloud computing environment. Also, another purpose is to utilize the TPBM to provide the real-time measurement results of the system behavior of multiple clients, which are subject to monitoring, at the level of instruction in a visual way. Since information on all of the users connected is the target for monitoring, the web-based TPBM was developed.

4. DESIGN OF m-TMS

m-TMS consists of three sides: TPB, TM-Interface, and Mobile-Device, and Fig. 4 shows its whole architecture.

On the TPB-side, computer system resources of the monitored target are regularly investigated and data on the most recent status of the resources are sent to the XML Converter. The XML Converter converts the data received from the system into XML data on BiT, integration, peripheral central resource (PCR), process, CPU, memory, and user.

The TM-Interface-side plays the role of a mediator that connects the TPB-side and the Mobile-Device-side. The Mobile-Device-side receives data on the current status of resources and trusted state of the TPB-side and provides monitoring through its mobile devices to users.

The Mobile-Device-side is composed of four internal components: User Interface, which receives the input of selected monitoring from users; Activity Update, which regularly delivers data on the selected monitoring and makes an analysis to reflect them to the Activity; Handler, which informs the analyzed data to the Activity Viewer; and Activity Viewer, which receives the notification of data update through the Handler and provides visualized information on the monitoring to users. The Mobile Device selects desired monitoring through User Interface regarding the relevant monitored computer system. Activity Viewer visualizes the information on the monitoring selected by User Interface. Activity Update brings data on the current status of computer system resources through the TM-Interface-side in relation to the selected monitoring and analyzes them in line with the Activity form. The handler notifies the Activity Viewer that such data collected and analyzed have been updated, and the Viewer is continuously updated in such a way.

In particular, in the case of multiple monitoring, multiple IP are input and the HttpURLConnection examines whether the relevant servers are operating. The conditions of the servers are received and when they are not operating, messages are delivered to the user. If a server is operating, the IP is added to the list and the location is received from the server and the physical
location of the computer system is provided to the user. Monitoring is done based on the IP. The added IP is shared among all Activities and all single monitoring functions are provided to them. Also, its IP, which should not be monitored, should be deleted in the Multi Activity list. At the same time the IP is deleted in each Activity List.

5. IMPLEMENTATION OF M-TMS

5.1 Monitoring of a Single Host

Fig. 5 explains the execution examples of Integrate Activity, CPU Activity, BiT Activity, Process Activity, User Activity, PCR Activity, Log Activity, RAM Activity based on an m-TMS emulator, and Android OS Mobile Phone for single host monitoring.

m-TMS consists of 8 different Activities: Activity (1) visually shows the trusted state of monitored computer system targets—RAM, CPU, user, process, and virtual memory—through the pentagon graph; Activity (2) is about the trusted state Log of the process; Activity (3) provides memory information; Activity (4) divides the trusted state of the current process into trusted, distrusted, and unknown and it provides the monitoring of each process; Activity (5) visualizes the branch instruction trace through the BiT Profiling tool; Activity (6) provides the...
monitoring of PCR values of TPM; Activity(7) provides comparative monitoring of basic information about the CPU and the utilization rates of each core through bar and line graphs; and Activity (8) provides information about user IDs, user group identifiers (GIDs), and the amount of resource utilization by each user in connection with information about sessions employed by users. The selected Activity takes the resource information data of a single host periodically by the TM-interface-side and analyzes data with respect to the type of Activity. Analyzed data should also visualize and monitor the selected Activity.

5.2 Monitoring of Multiple Hosts

For multiple monitoring by m-TMS, the relevant server’s IP is added. Activity (1) in Fig. 6 concerns the operation screen when multiple IPs are added. An IP is added to the relevant Activity by pressing the Add button, and the computer system server of the input IP determines whether execution is made and then delivers. The latitude and longitude of the address through Geocoder, which is provided by Android, are calculated and the derived value is synchronized with MapView, which enables the monitoring of the location. Under Activity (2) in Fig. 6, the IP may be selected from the list on the left side and this type of a selection may identify the utilization amount of the relevant system CPU and RAM resources. The locations of all systems under monitoring may be pinpointed through the All View mode. It supports the detailed address information if the overlay should be selected in MapView and the centric point of MapView moves to the center of the overlay simultaneously.

When the IP is added by a menu change through the Save button in Multi Activity, the same monitoring function in Fig. 6 is provided. The multi monitoring system gets the IP address from the user. It should select the IP of the single host, which adds dynamically to each Activity List. When the IP, which should not be monitored, should be deleted in the Multi Activity, it should be deleted in another Activity List simultaneously. Activity (3) in Fig. 6 shows the IP added by Multi Activity on the right side. This supports the monitoring and visualization of the CPU utilization rate, the RAM occupancy rate, the virtual memory occupancy rate, the process, and the...
user through the pentagon graph. When each Activity monitors two or more IPs, a button with which a selection may be made is dynamically generated on the Menu button. When an IP desired by the user is selected from the IP list, the relevant computer system’s resources and its trusted state are monitored. Therefore, m-TMS can monitor the system behavior of multi hosts with TPB.

5.3 BiT Profiling Monitoring

Fig. 7 shows the visual trace result of an Activity through BiT Profiling. The user selects the
process list; the name of the process; PID; utilization of CPU and memory; and the number of BPC, TPC, and EP. If the system should be attacked these can trace the branch instructions by BiT Profiling. It then shows the color red for those things and the monitored system behavior. The trace can be monitored by the BiT Profiling structure.

6. CONCLUSION

In order to provide a higher-level of reliability than existing software security methods, the m-TMS embodied in this study provides real-time monitoring of the information and resources of a computer system on which TPM chip-based TPB is mounted and visualizes the information and trusted state of the computer system regarding multiple IPs and not for a single IP. m-TMS provides monitoring to users by visualizing the current status of resources and the trusted state of IaaS in a cloud computing system. It enables them to cope with errors in the computer system. Meanwhile, the BiT Profiling technique provides the monitoring of branch instruction tracing. Each BPC, TPC, and EP are the result of each branch instruction tracing, and visualized information on the early locations of suspected logical errors is provided to a user through markings in red and via graphs. Accordingly, the user can actively respond. Moreover, the trusted state of IaaS through the TPB-mounted computer system in a cloud environment may be monitored through mobile devices, thereby enhancing security stability.

Future research will concern m-TMS’s functions and not just its monitoring functions for responding to attacks and its logging functions for notifying users of possible locations of logical errors caused by attacks, the possibility of attack on each process, and the occurrence of problems with a trusted state.

REFERENCES

Eun-Ha Song, Hyun-Woo Kim and Young-Sik Jeong


---

**Eun-Ha Song**

She received a B.S. degree in Statistics and an M.S. and Ph.D. degree in Computer Engineering from Wonkwang University in Iksan, Korea in 1997, 2000, and 2006, respectively. She is a part of the general studies and education team at Wonkwang University in Korea. Her research interests include grid computing, trusted computing, mobile and USN computing, and middleware.

---

**Hyun-Woo Kim**

He received his B.S in Computer Engineering from WonKwang Univ. in 2012. He has been studying for his MS course in the Department of Computer Engineering at WonKwang University. His interests are cloud computing and ubiquitous computing.
Young-Sik Jeong

He is a professor in the Department of Computer Engineering at Wonkwang University in Korea. His research interests include grid and cloud computing, mobile and ubiquitous sensor network (USN) computing, and USN middleware. He received his B.S. degree in Mathematics and his M.S. and Ph.D. degrees in Computer Science and Engineering from Korea University in Seoul, Korea in 1987, 1989, and 1993, respectively. Since 1993, he has been serving as an IEC/TC 100 Korean Technical Committee member, as the IEC/TC 108 Chairman of Korean Technical Committee, and as an ISO/IEC JTC1 SC25 Korean Technical Committee member. He is also a member of the IEEE.