Trusted Boot Mechanisms in Physical and Virtual Environments

Sergei V. Mironov
Institute of Engineering Physics
Training Center
Serpukhov, Russia
smironovs@yandex.ru

Valentin L. Tsirlov
Information Security Department
Bauman Moscow State Technical
University
Moscow, Russia
v.tsirlov@bmstu.ru

Valery N. Baburin
Software Development Department
NPO Echelon
Moscow, Russia
mail@cnpo.ru

Abstract. This article discusses computer operating environment safe boot issues. The regulatory framework in the area of trusted boot of physical and virtual equipment is analyzed. Characteristics of three classes of trusted boot tools: the levels of basic input-output system, expansion card and boot record are discussed, examples are given. Special attention is paid to the difference between software and firmware modules of trusted boot. The scheme of trusted boot of virtual infrastructure with I and II types hypervisors is offered and explained.

Key words - trusted boot, trusted boot tool, trust chain, basic input/output system, boot sector, master boot record, expansion card, rootkit, bootkit, virtual infrastructure, hypervisor.

I. INTRODUCTION

There is an opinion that intruders choose BIOS as a subsystem for infection rarely because of the variety of manufacturers and different versions of old basic input/output systems (BIOS). However, experience shows that an ongoing struggle between antivirus manufacturers and malicious software developers makes the intruders to develop such programs that will be invisible to any program in the operating system, and therefore they will be implemented at the level of firmware [1, 2].

Apparently, to protect against this type of malicious software we must be sure that neither BIOS, nor the boot loader, nor the operating system has been modified, besides the actions to verify the authenticity of programs that run from the start of computer and until the OS loading shall be consistent. In other words, the administration should not be transmitted to the next firmware before we make sure that its code has not been modified. The idea for the tools that provide procedures for sequential verification of the booted code is called a trusted boot [3].

II. CLASSIC BOOT SCRIPT

"Classic" hardware boot script will be discussed for better understanding of a trusted boot.

BIOS is located in ROM chip on the motherboard of the computer (this chip is often called ROM BIOS) and is stored in a packed form. Unpacked BIOS boot loader primary initializes the chipset and unpacks the main part of BIOS to a special area of RAM (shadow memory) immediately after powering on the computer. Next, BIOS starts testing the system to verify its operability (POST-procedures). After completing the self-test procedure, the part of BIOS code that implements POST procedures is removed from RAM. The main task of BIOS part, left in RAM, is to find the active boot device.

The standard procedure of Bootstrap Loader, activated by the INT 19h interrupt, selects the Initial Program Loader (IPL), a block device that supports the sectors reading function. A list of bootable devices is stored in computer nonvolatile memory (CMOS), and the order of viewing of this list is one of the adjustable BIOS parameters. The procedure tries to boot the very first sector into RAM from this device, and if there is a loader signature AA55h at the address 0000:7DFE, then the control is transferred to it at the address 0000:7C00h. The 1st sector located on the 0 side, cylinder 0 of the drive from which the boot is made contains the loader, which loads the OS or its core. If the boot is performed from the hard disk, then the 1st sector there contains the master boot record (MBR). It is also loaded into the memory at the address 0000:7C00h. Next, if the signature AA55h is at the end of the sector, then the control is transferred to its beginning. The master boot loader copies its code and partition table to the address 0000:0600h and continues its further execution in a new area. The task of the main loader is to find the active partition, load its 1st sector into memory and transfer control to it, if it has a boot loader signature.

Expansion cards installed in expansion slots may have additional ROM BIOS (additional, or expansion, ROM). They are used by EGA/VGA/SVGA graphics adapters, hard disk controllers, SCSI controllers, network adapters with remote boot and other peripherals. The C8000h ÷ F4000h area is reserved in the memory space for these modules. At the final stage of execution (after the pointers loaded the interrupt vectors to their own handlers), POST scans the extension modules area in search of additional BIOS modules. The additional BIOS module should have a header aligned along a 2 Kb boundary, which indicates the signature of the module start (AA55h), its length in 512 byte blocks, the entry point of login procedure with Ret Far, the pointer to PCI data structure and to the extended header structure of ISA PnP cards. The initialization procedure

overrides the interrupt vectors serviced by BIOS, including INT 19h (Bootstrap), which allows obtaining boot control, for example from a local network.

III. Invisible n antivirus - rootkits and bootkits

What can happen if the malicious software interferes with the boot script described above?

The first attempts to introduce malicious code into the firmware code became known back in 1999. This is a widely known virus CIH, or Chernobyl. Its impact on BIOS was destructive and the hardware did not boot at all, that cannot be called a successful implementation.

In 2006, a prototype of a rootkit called IceLord appeared, it infected BIOS quite correctly (meaning that after its implementation, the hardware was still working). And in 2011 it became known about the working rootkit Mebromi that is capable of modifying BIOS. Mebromi has mechanisms of hiding from traditional antiviruses, and it is impossible to get rid of it even after replacing the hard disc.

According to the reports on threats and trends of several companies over past years [1], bootkits have become one of the key technical trends.

Bootkit has something in common with boot virus, but it contains components that are introduced into the operating system before it is loaded. At the initial stage of its work, bootkit replaces the original loader and waits for the computer restart. The main task of bootkit is to intercept the INT 13h interrupt, by means of which file components of the operating system are read from the disk to the memory in order to replace operating system files with their components. Thus, bootkit can be unnoticed for any application system running in modified OS [4].

However, it should be noted that UEFI (Extensible Firmware Interface) has replaced BIOS in modern computers. A new specifications complex was developed to replace BIOS due to its significant limitations that restrained the development of computing systems. A key feature of UEFI is the SecureBoot mechanism, which verifies the loaded OS components with cryptographic methods - using a digital signature mechanism, the private key of which is written to motherboards chips.

The appearance of a well-unified UEFI simplified the development of legal and malicious software. At the end of 2014, at the "31st World Hackers Congress", Rafal Wojtczuk and Corey Kallenberg demonstrated an attack on UEFI, called Speed Race, associated with vulnerability in the implementation of UEFI. Among hardware platforms that are vulnerable were Dell Latitude and HP EliteBook [5] laptops.

In July 2015, after a scandalous leakage of source codes of spyware, developed by the Italian company "Hacking Team", it became known about the first UEFI-rootkit. "Hacking Team" employees developed a malicious software specially for Insyde, a UEFI company, which is very popular in laptops, but this software successfully operates on platforms with AMI BIOS. The peculiarity of malicious software, as well as Mebromi rootkit, is that reinstalling the OS, or formatting the hard disk, or even its replacement, will not help to free the computer from its impact [4, 5].

Thus, we see that the development of malicious software is quite dynamic, so SecureBoot technology should be considered only as a possible measure from the whole complex of mechanisms to ensure a safe boot, but not as panacea. One more mechanism for secure boot should be trusted boot tools.

IV. Trusted boot of physical environments

So what is a trusted boot and how does it help to fight against malicious actions?

According to the requirements of UPD.17 of FSTEC of Russia № 17 dated 11.02.2013, № 21 dated 18.02.2013 and № 31 dated 14.03.2014, trusted boot shall ensure:

- blocking of attempts of unauthorized boot of abnormal operating system (environment) or unavailability of information resources for reading or modification in case of abnormal operating system boot;
- user access control to the process of operating system boot;
- control of the integrity of software and hardware components of computer aids.

"Requirements for trusted boot tools" were approved by the order of FSTEC of Russia № 119 dated 27.09.2013, according to which this class of tools is divided into three types [6, 7]:

- trusted boot tools of the basic input-output system level;
- trusted boot tools of extension board level;
- trusted boot tools of boot record level.

The first group of tools – of BIOS level - this is usually multi-component software tools, one module of which is built directly into the microprogram of the motherboard. Examples of such tools of trusted boot are "MDZ-Eshelon" of CJSC "NPO Echelon" and "Altell Trust" produced by Altell. The PCI/PCI-E boards are not required for the operation of such tools, which simplifies and speeds up the commissioning of the ISS.

Trusted boot tools of extension board level are always software and hardware ones. This is the broadest segment of the trusted boot market. Examples are the PAC "Sobol" produced by "Kod bezopasnosty" company, DDO CAD "Accord-AMDZ", "Maksim-M1 NPO RusBITeh" and others.

Finally, the mechanism of operation of trusted boot tools of boot record level is based on the modification of boot sectors of logical partitions of hard disks. The contents of the boot sectors are encoded, this allows hiding information about logical partitions when the computer boot is unauthorized. Example is "Trusted Boot Loader" produced by "Kod bezopasnosty" company.

It also should be noted that the mandatory strengthening 1 (for the second class/security level of ISPD) and 2 (for the first class/security level of the ISPD) to UPD.17 measure indicates the need for trusted boot tools application of basic input/output system or expansion board level, i.e. the insufficiency of trusted boot tools application of the boot record level for the above classes/security levels.

The selection of trusted boot tools should be highlighted. There is an unreasonable opinion in the Internet that "hardware modules of trusted boot have significant advantages over pure software tools" [8, 9]. Let's address to the mechanism of functioning of trusted boot tools (TBT) of BIOS level and expansion board level. To do this, let us recall the first part of this article about computer boot script. As it was mentioned above, when the POST procedures are completed, the extension module area is scanned for additional BIOS modules. At this stage, both types of TBT gain control redefining the Bootstrap (INT 19h) vector to themselves. Since TBT has gained control, regardless of the priority of the selection of the boot devices in BIOS Setup, trusted boot module will gain control. The difference is that the TBT loader of expansion board level is identified by the program written in the nonvolatile memory of the controller, and TBT loader of basic input/output system level is identified by the module built into BIOS microprogram. Figure 1 shows the main stages of computer boot, where the arrow marks the point at which the TBT of basic input-output system level and expansion board level start working.

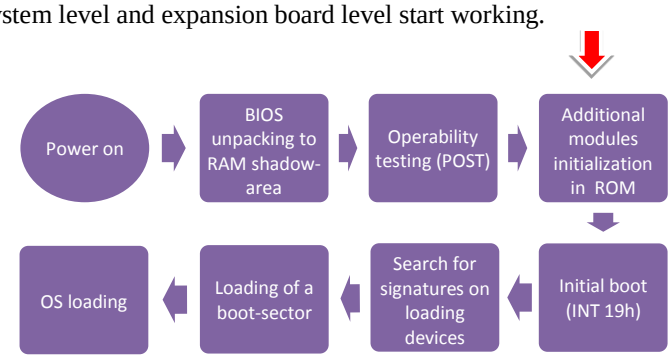

Figure 1 — Main stages of computer boot with BIOS microprogram

According to anti-virus reports of several companies more bootkits attack mobile platforms. It can be predicted that the popularity of software tools of trusted boot will only grow because PCI slots are missing there.

V. Trusted Boot of Virtual Environments

It's no secret that today the placement of users' workstations in virtualization systems becomes more popular. This allows creating a single point of control, establishing and administration of workstations. Let's address the issue of security of virtual environments and consider some typical actions of an intruder and the threats to which they lead:

unauthorized reading and modification of data processed in CA leads to integrity and confidentiality thread of information processed in AS;

unauthorized modification of authentication information, namely, substitution of certificates, password hashes and other key data may result in the addition of new privileged accounts to the system or in compromising existing ones;

introduction of software bugs and malicious software, in particular rootkits and bootkits, designed to mask objects, to control events in the system and to collect data.

Thus, on the one hand, the hypervisor that performs the functions of virtual machine monitor increases the service capabilities of computer and reduces its operational costs. But, on the other hand, the hypervisor can privately implement a software bug with uncontrolled capabilities that are the threat to information security.

In witness of this, the nature of the threat of a "thin hypervisor", which has the imaginative name Blue Pill [10] will be briefly discussed. The concept of Blue Pill is to capture a running instance of the operating system (the capture is performed when the OS starts) by a "thin" hypervisor and it virtualizes the rest of the computer. The operating system will still support the existing references to all devices and files, but almost everything, including hardware interrupts, data requests and even system time will be intercepted by hypervisor, which will send fake responses.

Due to the existence of virtual environment threats like those described earlier, the requirement to a trusted boot in the above-mentioned FSTEC orders refers not only to physical equipment, but also to virtualization environment (protective measure ZSV.5):

• trusted boot of virtualization servers, virtual machines (containers) and virtualization control servers should be provided in the information system according to protection measure "Provision of computer aids trusted boot".

• trusted boot shall provide blocking of attempts to unauthorized boot of hypervisor, host and guest operating systems.

• trusted boot of hypervisors is provided with trusted boot tools that function on virtualization servers.

• trusted boot of virtual machines (containers) is provided with the use of multi-component trusted boot tools, individual components of which function in hypervisors.

Let's consider the stages of a trusted virtual environment boot using the TBT of basic input/output system level [3, 11]. In this case we have a hypervisor of the first type, which is installed directly to the virtualization server as the system software. The module for trusted boot of basic input/output system level can be installed to the platform before the hypervisor is installed, and this component (that is, the component installed in the physical server chip) will control the security of hypervisor. Virtual trust module is built into virtual BIOS in each virtual machine, its work is aimed to control the integrity of virtual equipment, guest operating systems and files in them.

In the case of the second type hypervisor, its integrity and integrity of basic operating system is controlled by the trusted boot module built into BIOS code of virtualization server. The same trusted boot module, but installed to the virtual machine instead of the virtualization server can perform trusted boot of virtual equipment and guest operating systems.

VI. Conclusions

The variety of malicious software operating at BIOS/UEFI level indicates a serious need for protection from them. Moreover, an intruder can damage the system without using malicious software (for example, by booting from an external media). The task of trusted boot tools is to permit access only for authorized users, control the integrity of partitions and files, perform a trusted boot. Three types of trusted boot and the scheme of trusted boot of virtual environments using CA BIOS level are discussed in the article.

Nowadays most of the tools are software and hardware ones. The average price of trusted boot hardware module is dozens of thousands of dollars. However, the use of a hardware card is not always possible: a necessary slot on the motherboard can be missing or the board may be incompatible with BIOS version of the motherboard. Moreover, the installation of hardware part to a large number of platforms requires significant time resources. These reasons, together with the growing number of threats to mobile platforms, allow predicting that the CA of basic input/output system level will become more popular.

A number of patents have been obtained for the proposed solutions for information protection, including the protection in virtualization environment. This confirms the innovativeness, practical value and reliability of the study.

REFERENCES

[1] Barabanov A.V., Grishin M.I., Kubarev A.V. Modelirovanie ugroz bezopasnosti informatsii, svyazannykh s funktsionirovaniem skrytykh vredonosnykh komp'yuternykh program. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2014. N 4 (7). P. 41-48. (In Russ.)

[2] Vorobiev, E.G., Petrenko, S.A., Kovaleva, I.V., Abrosimov, I.K. Organization of the entrusted calculations in crucial objects of informatization under uncertainty. In Proceedings of the 20th IEEE International Conference on Soft Computing and Measurements (24-26 May 2017, St. Petersburg, Russia). SCM 2017, 2017, pp 299 - 300. DOI: https://www.doi.org/10.1109/SCM.2017.7970566.

[3] Wojtczuk R., Kallenberg C. Attacking UEFI Boot Script. 2015. URL: https://bromiumlabs.files.wordpress.com/2015/01/attacksonuefi_slides.pdf.

[4] C.Kallenberg, R.Wojtczuk. Speed Racer: Exploiting an Intel Flash Protection Race Condition: Komposter 2.0. 2015. – URL: http://composter.com.ua/documents/Exploiting_Flash_Protection_Race_Condition.pdf

[5] P.Lin. Hacking Team Uses UEFI BIOS Rootkit to Keep RCS 9 Agent in Target Systems: TrendLabs Security Intelligence Blog. 2015. URL: http://blog.trendmicro.com/trendlabs-security-intelligence/hacking-team-uses-uefi-bios-rootkit-to-keep-rcs-9-agent-in-target-systems

[6] Barabanov A., Markov A. Modern trends in the regulatory framework of the information security compliance assessment in Russia based on Common Criteria. In: ACM International Conference Proceeding Series 8. Ser. "Proceedings of the 8th International Conference on Security of Information and Networks, SIN 2015". 2015. P. 30-33. DOI: 10.1145/2799979.2799980.

[7] Barabanov A., Markov A., Tsirlov V. Procedure for Substantiated Development of Measures to Design Secure Software for Automated Process Control Systems. In Proceedings of the 12th International Siberian Conference on Control and Communications (Moscow, Russia, May 12-14, 2016). SIBCON 2016. IEEE, 7491660, 1-4. DOI: 10.1109/SIBCON.2016.7491660.

[8] Wojtczuk R., Rutkowska J. Attacking SMM Memory via Intel® CPU Cache Poisoning. URL: http://invisiblethingslab.com/resources/misc09/smm_cache_fun.pdf

[9] Hackers find a new place to hide rootkits. URL: http://www.infoworld.com/article/2653209/security/hackers-find-a-new-place-to-hide-rootkits.html

[10] Rutkowska J. Subverting Vista Kernel for Fun and Profit. Black Hat, 2006. 52 p.

[11] Avezova Ya.E., Fadin A.A. Voprosy obespecheniya doverennoy zagruzki v fizicheskikh i virtual'nykh sredakh. Voprosy kiberbezopasnosti [Cybersecurity issues]. 2016. N 1(14). P. 24-30. (In Russ.)