A New Mechanism for Trusted Code Remote Execution

Liqiang Zhang
Computer School, State Key Lab of Software Engineer, Wuhan University, Hubei, China
whujking@126.com

Huanguo Zhang
Computer School, State Key Lab of Software Engineer, Wuhan University, Hubei, China
liss@whu.edu.cn

Xiantao Zhang
Computer School, State Key Lab of Software Engineer, Wuhan University, Hubei, China
xiantao@mail.whu.edu.cn

Lu Chen
Computer School, State Key Lab of Software Engineer, Wuhan University, Hubei, China
ieucl@163.com

Abstract

Current authentication model in collaboration computing use entity's identity to establish trust relationship, this kind of security mechanism is at great risk. The identity authentication only can guarantee that the interactive entity's identities are real, but know nothing about the entities' security status and behaviors. Therefore it also needs to carry on the authentication to the entity's platform status, simultaneously needs to introduce a dynamic method which can carry on the authentication to the entity's behavior. This paper integrates the identity authentication, platform authentication and behavior authentication based on Trusted Computing technology, remote attestation and trusted behavior, proposes a new mechanism for trusted code remote execution. This mechanism can solve the problem of codes executing remotely effectively.

1. Introduction

Current collaboration computing is mainly based on the client-server mode. If a client needs a computing result, it will submit the job codes to the server, and the server will return the result back after computing. The trust relationship between the client and server is based on the identity authentication. In order to protect both the client and server, they need to authenticate each other before job submitting. Once they authenticated, a trust relationship has been established between them, and then the job execution and result will be considered to be trusted.

This kind of security mechanism which establishing trust relationship just on identity authentication is at great risk. The identity authentication only can guarantee that the interactive entity's identities are real, but know nothing about the entities' security status and behaviors, so it is very hard to trust the returned result.

Therefore, "authentication" of an entity should have a broader meaning than it does currently. It should encompass not just cryptographically verifying its origin, but also include verifying or proving that its behavior conforms to a required security policy [1]. In order to hold the behavior characteristic of an entity, we adopt the definition from the Trusted Computing Group (TCG) [2]: that is "Trust is the expectation that a device will behave in a particular manner for a specific purpose". We need to design a new mechanism to ensure that an entity's identity is trusted, platform status is trusted and behavior is trusted.

We integrate the identity authentication, platform authentication and behavior authentication based on Trusted Computing technology, remote attestation and trusted behavior, propose a new mechanism for trusted code remote execution. There is very little applications applied remote attestation, and the trusted behavior is at the stage of research, no matter from the point of theory or technology. This paper is trying to combine these mechanisms to solve the problem of code execution remotely. The main contribution of this paper is to propose a new mechanism for trusted code remote execution. We also design remote behavior monitor mechanism to ensure trusted behavior of code execution remotely.

The rest of the paper is organized as follows: Section 2 overviews the background of Trusted Computing and property-based attestation. Section 3 discusses the mechanism in detail. Section 4 summaries the related work and we draw a conclusion in Section 5.

2. Background

0-7695-3073-7/07 $25.00 © 2007 IEEE
DOI 10.1109/CIS.Workshops.2007.126

In this section we briefly introduce the emerging Trusted Computing technology and property-based attestation.

2.1 Trusted Computing

Trusted Computing Group (TCG) is a vendor-neutral and not-for-profit organization for promoting industrial standards for Trusted Computing technologies [2]. It takes a distributed, system-wide approach to the establishment of trust and security.

The main solution of trusted computing is as follows: firstly building a root of trust, then building a trust chain which starting from root of trust to hardware platform, operating system, and applications [3]. The previous portion of code that is executed checking the integrity of the next component to be executed and passing trust, then trust can be extend into the whole computer system.

Trusted Platform Module (TPM) is the root of trust. It includes CPU, memories, I/O resources, crypto coprocessor, random number generator and embedded operating system etc. Its functions including secure storage, key generation, encryption and decryption. TPM is usually embedded on the main board and it is tamper-resistant.

Platform Configuration Register (PCR) is used to record the platform's configuration information which are executable software and configure files measured by hash function. PCR values are 160bits, reset at boot time and updated every time a new executable is loaded, with the hash of that executable added to the end of a hash chain. PCR are usable for trust measurement and report.

Trusted computing is very powerful for integrity measurement and report. Integrity measurement is to measure the integrity of the platform through Root of Trusted Measurement (RTM). The measurement results are stored in PCRs and will be compared with expected results. Integrity report is to sign the trusted status results (usually PCRs values) with platform certificate, and report to other entity through the Root of Trusted Report (RTR). Integrity measurement and report are foundation of remote attestation.

2.2 Property-based Attestation

Remote Attestation is a way for an entity to authenticate itself to a remote party, or for a remote party to verify the authentity of the entity [2]. The TCG specifications define mechanisms for a TPM-enabled platform to reliably report its current hardware and software configuration to a local or remote challenger. The remote attestation is a challenge and answer process. Firstly challenger requests the platform to attest to it. Then the platform asks its TPM to report current PCRs values and TPM will sign the PCRs values with an AIK (Attestation Identity Key), then the platform sends the signed PCRs values and TPM credential back to challenger. Challenger verifies the result then finishes the remote attestation process.

Though remote attestation is a novel way to platform attestation, it still has some drawbacks such as privacy leak, lack of scalability, lack of openness and static report.[1][8].

Researchers have approved some new methods of attestation including property-based attestation [8] and semantic remote attestation [1]. In property-based attestation, not the binary information but the platform high level properties are verified. The platform has some property credentials which mean it has some special security properties. The credential is signed by TTP (Trusted Third Party). The TTP verifies the platform's hardware and software configuration and signs property credential for the platform. When the challenger asks for the attestation the platform will show the property credentials instead of binary information. This method not only completes the attestation but also doesn't leak privacy information of the platform.

3. Trusted Code Remote Execution Mechanism

3.1 Architecture

Trusted code remote execution mechanism is shown in the figure below:

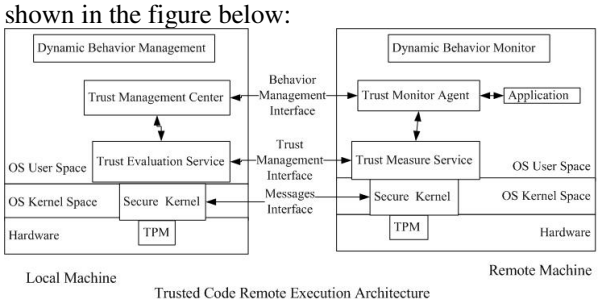

Figure1: Trusted Code Remote Execution mechanism

Local machine and remote machine are both computing platform with TPM, and have a secure kernel executing on the trusted platform. Current commercial operation systems are all open systems, their system space can easily modified so this is the main cause of many security accidents such as Buffer overflow, Virus, Trojans and Worms. It is necessary to use security enhanced kernel in the operating system kernel layer.

Trust Measurement Service (TMS) and Trust Evaluation Service (TES) are built on secure kernel.

Information exchanged between these services are managed by Trust Management Interface (TMI) and protected by low-layer Message Transmission Interface (MTI). The TMS of remote machine collects the platform trust values and sends this information to TES of local machine through TMI. TES will validate these values according to standard trust references and makes trust evaluation on remote machine.

After attestation of both identity and platform on remote machine, local machine notifies remote machine to install Trust Monitor Agent (TMA) and makes an attestation on the agent installation and execution status of remote machine. TMA could be designed and implemented either local machine or TTP. The manufacturer of this agent should sign the agent with a signature. TMA should be authenticated by TTP before dispatched; the authentication should including function, integrity and privacy factors, and be signed with a signature. This authentication process guarantees that TMA can be trusted on design, implementation and function. There also must be some mechanisms to guarantee that TMA is running correctly. This is usually done by Trust Management Center (TMC).

The TMC of local machine receives real-time messages from TMA, these messages including agent runtime information, platform status information of remote machine and all kinds of status information about the job codes execution. TMC analyzes the execution status of both trust monitor agent and job codes through the information received from TMA. This information is exchanged through Behavior Management Interface (BMI).

3.2 Trusted Identity and Trusted Platform

Trusted Computing can solve the problem of identity and platform authentication both on local machine and remote machine.

Each TPM has a pair of Endorsement Key (EK) stored in the shield location of TPM which is assigned by the manufacturer. The public key of EK is used for the unique identity of the TPM. Each platform with a TPM has a Platform Certificate with EK, which identity the platform uniquely. This unique identity fits the need of identity authentication very well. So local machine and remote machine could authenticate each other trustful through their Platform Certificates.

Remote attestation is a process for local machine to verify whether remote machine has exactly running environment for its job. Attestation includes hardware, software and configuration. Hardware and software attestation is to conform whether they are genuine and not modified. This is a very important process to detect the potential attacks before execution for many information revealing are caused by software modification. After hardware and software attestation we also need configuration attestation. Many accidents occurred is not for the hardware and software themselves but for the mistake configurations. Both the attestations can verify that remote machine has the exactly running environment.

Property-based attestation is based on the credentials signed by TTP. The TTP could be a VO organizer in gird or a private CA in Trusted Computing. Both local machine and remote machine trust the TTP. In the attestation process remote machine will show lots of property credentials to prove its abilities. Then local machine will verify these credentials with TTP who signed the credentials. If these credentials are genuine and up to date then local machine will trust remote machine.

3.3 Trusted Behavior

Trusted behavior means that local machine monitors the job execution behaviors on remote machine, makes sure that these behaviors match the expected behaviors. So we should characterize the normal job execution behaviors and monitor the real time behaviors happening on remote machine.

TMA is the core component for monitoring the job execution behaviors that every behavior of job execution should under its supervision. This sounds very like the concept of “Reference Monitor”. Specific monitoring method such as access control mechanism, behavior prediction mechanism can be used. The agent collects all kinds of behavior information about the job execution, makes judgment according normal behavior and returns the result back to trust management center in real time.

Besides the real-time job execution behavior monitoring, it is necessary to score the execution platform trust value from historic records. This could be based on probability and statistics or fuzzy logic method. These methods are different from the classic access control mechanism and very useful for trust-based system..

3.4 Security Analysis

This mechanism is effective for following reasons:

Local machine and remote machine can use Platform Certificate for identity authentication which can make sure that both identities are trusted.

Property-based attestation can check whether remote machine has an appropriate job execution environment. This is done by property-based attestation. Remote machine has the TTP signed credentials that local machine can trust. The attestation process can check remote machine’s platform and configuration in both binary and property way.

TMA examines the job execution process. It monitors the job execution behaviors, makes sure that the job is executed in a normal way and the result is trusted.

Information exchanged between local machine and remote machine are encapsulated by MTI which is protected by cryptology.

4. Related Work

Trusted code remote execution means local machine should attest on remote machine and trace the execution process behavior, remote machine should have a genuine platform which it claims to be. Related methods are lacking in at least one of these areas.

Daonity [5] is a Trusted Computing enable emerging working grid security standard, to manifest how behavior conformity can help to improve grid security. Daonity is an open source project and adapted and modified from TrouSerS [6], and plugged into GT4. Daonity is at first stage that it doesn't solve the behavior security problems after the job migration.

Garfinkel have proposed the TerraVM [7] virtual machine monitor architecture to interface with trusted hardware. The goal of Terra VM is similar to Microsoft's Palladium architecture. They all have a high-assurance trusted micro-kernel running on hardware that provides strong isolation. But Terra is focusing on isolation between multi tasks on a single machine but we are focusing on remote code execution between two machines.

Vivek haldar etc. have proposed semantic remote attestation [1] architecture. They implement a new technique based on language-based virtual machines. The core idea is using a language-based virtual machine that executing platform independent codes. A key advantage of this method is that reasoning about the behavior of a program is not tie to a particular binary. The VM will add additional properties to the program.

Ravi Sandhu etc. have proposed a peer-to-peer access control architecture based on Trusted Computing [4]. The core idea is implementing trusted reference monitor on trusted hardware, monitoring and validating the integrity of application, executing access control policies. This idea shares some commons with ours. But their paper mainly focuses on access control mechanism of local platform, talks nothing about behavior monitoring and analyzing of codes execution on remote platform.

5. Conclusion

In a trusted remote execution, local machine should attest on remote machine whether it has a genuine platform which it claims to be before submitting the job and traces the whole execution process, remote machine should provide correct execution and under monitoring by local machine. Previous methods are lacking in at least one of these areas.

We propose a new method integrates trusted computing and trusted behavior perfectly. We integrate the identity authentication, platform authentication and behavior authentication based on Trusted Computing technology, remote attestation and trusted behavior. This mechanism can solve the problem effectively.

Acknowledgement

This paper is supported by the National Natural Science Foundation of China under Grant Nos. 60373087, 60473023, 90104005, 60673071; the National High-Tech Research and Development Plan of China under Grant No. 2006AA01Z442.

Reference

[1] Vivek haldar, Michael Franz, "Symmetric Behavior-Based Trust: A New Paradigm for Internet Computing.", Proceedings of the 2004 workshop on New security paradigms, ACM Press, NY, USA, 2004, pp.79-84

[2] Trusted Computing Group. TCG Specification Architecture Overview https://www.trustedcomputinggroup.org/ groups/TCG_1_0_Architecture_Overview.pdf.

[3] Huanguo Zhang, Jie Luo, Gang Jin, Fajiang Yu, Fei Yan. "A survey of Trusted Computing" (in Chinese), *Wuhan University Journal of Natural Science*, Wuhan University Press, Wuhan, China, 2006, pp.513-518

[4] Ravi Sandhu, Xinwen Zhang. "Peer-to-Peer Access Control Architecture Using Trusted Computing Technology", SACMAT05, ACM Press, NY, USA, 2005, pp.147-158

[5] Wenbo Mao, Fei Yan, Chunrun Chen, "Daonity: grid security with behaviour conformity from trusted computing," Proceedings of the first ACM workshop on Scalable trusted computing(STC06), ACM Press, NY, USA, 2006, pp.43-46

[6] TrouSerS. The open-source TCG Software Stack. http://trousers.sourceforge.net/

[7]T. Garfinkel, B. Pfaff, J. Chow, M. Rosenblum, and D. Boneh. "Terra: a virtual machine-based platform for trusted computing", ACM Symposium on Operating Systems Principles (ASOSP), ACM Press, NY, USA, 2003, pp.193-206

[8]A.-R. Sadeghi and C. Stuble. "Property-based attestation for computing platforms: caring about properties, not mechanisms", Proceedings of the 2004 workshop on New security paradigms, ACM Press, NY, USA, 2005, pp.67-77