Filter-Based Trusted Remote Attestation for Web Services

Peng Xinguang
Taiyuan University of Technology
College of Computer Science and Technology
Taiyuan, China
E-mail: sxgrant@126.com

Jia Wei
Taiyuan University of Technology
College of Computer Science and Technology
Taiyuan, China
E-mail: xingzhibi@163.com

Abstract—Remote attestation of the trusted platforms is one of the most important aspect of trusted computing specification. The approach of attesting program from higher semantic level has been proved more trustworthy than traditional static approaches. Semantic remote attestation based on virtual machine is a kind of representative measures of trusted attestation, but there are some shortcomings in attestation efficiency and given applications. A new method of trusted remote attestation for web service is proposed based on filter mechanism, which combines property-based attestation with trusted virtual machine. We construct trusted attestation environment by the open-source TPM Emulator and IAIK jTSS on Linux, and Tomcat as Web server. The prototype of remote attestation for web service based on filter was validated by establishing attestation trusted table. The results show that the remote attestation policy of filter mechanism improves attestation efficiency.

Keywords—trusted computing, remote attestation, web services, filter

I. INTRODUCTION

Trusted computing technology is already an important issue of information security, and trusted attestation is main part of Trusted Computing. TCG (Trusted Computing Group) released remote attestation specification, named as Privacy CA and DAA (Direct Anonymous Attestation) by successive order. They are based on AIK (Attestation Identity Key) and PCR (Platform Configuration Register) that reflect trusted platform sequence of event records.

The remote attestation technologies based on binary such as Privacy CA and DAA may take exposure on privacy information of remote platforms. Systems updating and backup are especially difficult in open and distribution applications. The remote attestation based on property gets over limitation of binary attestation to a certain extent but there are disadvantages of one-off and static attestation.

Frederic Stumpf proposed a Robust Integrity Reporting Protocol[1] that can attest the configuration information of remote platform effectively by improving the binary attestation. In fact, the binary attestation policy has been proved to be insufficiency. Remote semantic attestation based on language virtual machine[2] is proposed by Vivek Haldar, which can attest program behaviors on remote platform in semantic level. This dynamic attestation may overcome disadvantages of binary attestation proposal. Liqiang Zhang furthers this work by means of monitoring the access of virtual machine with TPM and keys[3]. It sharpens communication burden on servers because both servers and clients need language virtual machine for attestation.

Creditability and implementation of attestation are improved by means of introducing virtual technology into remote attestation. But the existing attestation policies take on disadvantage in attestation efficiency and specific application. An alternative attestation scheme based on filtering mechanism for web service is proposed, which combines strong points of property-based attestation with trusted virtual machine.

II. TRUSTED REMOTE ATTESTATION

A. TCG Introduction

TCG is an industry standards organization formed to develop, define, and promote open specifications for trusted computing and security technologies, including hardware building blocks and software interfaces, across multiple platforms, peripherals, and devices. It is set up by Intel, Microsoft, IBM and etc. nearly 200 companies. TCG specifications are designed to enable more secure computing environments without compromising functional integrity and with the primary goal of helping users to protect their information assets from compromise due to external software attack.

In the TCG specifications, the core of trusted computing platform technology is TPM (Trusted Platform Module) security chip. TPM has hardware protection function and internal access to the object in the TPM can only be provided by TPM interface. And these objects can only be secure access by its owner.

The root of trusted in platform computing environment every conversion, that trusted status can be maintained by way of transfer without being destroyed, then the platform is always trusted. A variety of operation will not undermine trusted platform under trusted environment, so the integrity of the platform itself to be assured, and endpoint security is guaranteed. This is the trust chain transfer mechanism.

According as TCG specifications, when user-level process is established and the kernel will measure its executable code and sensitive data. These measure values are stored in the PCR registers and SML (Stored Measurement Log), and each PCR was defined only relative with some system-specific events of computing platforms. When every relevant event occurs, computing platforms will save the results of event log, and record the results of the Hash values in the event expanded to TPM in the corresponding PCR. The process of taking R to PCR[n] is as follows:
Clearly, the PCR value is associated with the event history, and events are recorded in the order. Therefore its value reflects the actual historical sequence of events. Each computing platform through a public key AIK (Attestation Identity Key) that embedded in TPM, identifies their own identity, and uses its private AIK key signature on the message content in order to prove their origin.

B. Binary Attestation

TCG released remote trusted attestation proposal, including Privacy CA (Privacy Certification Authority) and DAA (Direct Anonymous Attestation), and they all are binary attestation. These attestation proposals are based on reporting the current status to the remote platform. Each computing platform marks their own identity through TPM public key AIK, if SML (Stored Measurement Log) is also used to verify whether the platform is credible.

Privacy CA verifies themselves through TTP (Trusted Third Party). At first the platform sends its AIK to TTP, and then the platform verifies himself to remote platform through AIK and signatures AIK.

DAA did not need TTP anymore, and can generate different AIK for the same TPM. DAA is different from Privacy CA, as DAA introduces timer. But this binary attestation has some drawbacks as follow:

1. Poor scalability.
2. Privacy exposures.
3. Lack of openness.
4. Huge number of possible configuration.
5. Run-time assurance.

Load-time attestation provides no run-time assurance as there can be a big time difference between integrity measurement and integrity reporting. The platform could be compromised since it has been booted.

C. Property-based Attestation

In order to overcome the shortcomings of the binary attestation Sadeghi and others proposed PBA (Property-Based Attestation[4]). This proposal is a more effective and flexible solution for remote attestation. PBA attempts to solve information exposures of the platform and configuration, including system updates and backup problems. The property is platform behavior for special application; each property is corresponding to multiple platforms, thus avoiding specific platform configuration information of direct exposure. On PBA foundation, L. Chen proposed a protocol for property-based attestation[5]. Each property and its corresponding configuration information correspond to an attribute certificate (Property Configuration Certificates), which was managed and dispatch by trusted third CA. This attestation protocol still use a lot of attributes to achieve zero-knowledge proof verification, and consequently computational complexity is high.

The properties mentioned above describe the underlying object such as platform or application on the behavior of the specific needs of some aspects, e.g., security-related needs. Typically, the property has a different level of abstraction. For example, we can describe the platform with the platform property to protect privacy. The platform has a built-in regulation of the measure with privacy, or platform can guarantee isolation, e.g., the strict separation of each processor. Platform also can provide MLS multi-layer security (Multi-Level Security) and so on. Whether the set of attributes is reasonable, and it is mainly decided by the use cases and needs. The advantage of PBA is that different platforms can provide the same property and meet the same requirements, although they use different components and different configurations. But security properties have to be defined before attestation.

D. Traditional Attestation Limitation

Either binary or property-based attestation can not guarantee that the remote platform’s program is security in the semantic layer; as a result program behavior is not security. Furthermore traditional remote attestations are one-off attestation, and can not attest the run-time status and dynamical results of remote platform.

III. ATTESTATION BASED ON VIRTUAL MACHINE

In order to attest the programs of remote platform in semantic level, and overcome defects of traditional attestation with static and one-off attestation, Seshadri proposed Pioneer system[6]. Pioneer system proposed a software attestation, and can attest remote program behavior. Its core component is checksum function that runs in the remote system. Pioneer can detect potential attacks by examining the response time. If the response time is in a given range the response is credible.

Vivek Haldar changed the checksum function that is the core part of Pioneer into trusted virtual machine, and the server and client both have language virtual machine, attest program behaviors in language virtual machine. The attestation results based on language virtual machine are more reliable than Pioneer system.

It must be mentioned that Terra architecture [7] that was proposed by T. Garfinkel here. The architecture builds different robustness trusted virtual machine on TVMM (Trusted Virtual Machine Monitor), and allows different applications running on different virtual machines. The virtual machine is divided into "open-box" virtual machines and "closed-box" virtual machines. Trusted operating system is running in the "closed-box" virtual machines, and particular applications run on the system; simple applications run on the "open-box". Terra is proved to be a successful architecture with TVMM.

IV. FILTER-BASED ATTESTATION FOR WEB SERVICES

A. Architecture Based on Filter Mechanism

Virtual trusted attestation can effectively improves the credibility of certification and achievability, but it is still deficient. For the literature[2] attestation strategy proposed by Vivek Haldar both server and client need language virtual machine. The attestation strategy heightens server
communication burden; and do not prevent the middle attack. This attestation is not flexible, and attested program however needs attest next time.

On the foundation of Vivek Haldar and the ideas of PBA filter-based trusted remote attestation thought is proposed in the paper in order to improve previous attestation proposal. The architecture of filter-based trusted remote attestation is shown in Figure 1. The architecture of filter-based trusted remote attestation is composed of Attestation Filter, Attestation Requester, Actual Server, Virtual Server and Client Application.

In the figure 1, Virtual Server indicates the virtual mirror of Actual Server. When client program arrives to Server, Attestation Filter judges whether it is first access. It is forced to run in the Virtual Server If it is first so. And then program behavior is attested by traditional method. If the running result is trustworthy, and then it is allowed to access Virtual Server; otherwise the request is denied. Attestation Filter is the core part of the attestation proposal. The server can determine which program behavior needs to attest, which program behavior didn’t need to attest, so the filtering strategy of Attestation Filter is particularly important. The attestation process of filter-based trusted remote attestation is as follows:

1. Client Application sends request to the Server.
2. Attestation Filter judges whether it is first access. If it is first access and then turn into the step (2), or else turn into the step (6).
3. Attestation Requester sends client request to the Virtual Server.
4. The client program is forced to run in the Virtual Server and attestation results return to Attestation Requester.
5. The Attestation Requester sends attestation results to the Attestation Filter. If the client request is attested to be trustworthy the Hash value of client program is appended into the Attestation Filter, or else the client request is rejected.
6. The Attestation Filter sends client request to the Actual Server.
7. The Actual Server responses to the Attestation Filter.
8. The Attestation Filter sends response from the Actual Server to the Client Application.

B. Filter-based Attestation for Web Server

For specific server, the filtering strategy should be specific. Here is filtering strategy for Web server. At first, Web server accessing is classified into four different classifications:

1. Read web data;
2. Write web data;
3. Modify web configuration;
4. Other webs.

And then web data are also classified into public data and private data. At last, different user permissions to users are setup. Filtering process for Web server is shown in Figure 2.

When user requests reach to the server and Attestation Filter will classify the access requests. The access sort after classification, user identity and HASH value of request parameter form a sequence. Then the sequence in the attestation trusted table is searched. If the trusted table has the sequence this request accessing server is allowed, and otherwise this request is sent to the virtual server to attest its
behavior. If its behavior is trustworthy, the request sequence is inserted into the attestation trusted table and request is allowed, and if not the request is rejected.

C. Filtering Mechanism on Large Server

For some large-scale servers they may host many different services. In reference the architecture design of Terra system, a number of different trusted virtual machines are installed in the server. Different services attestation is located in respective virtual machines, and each virtual machine has own attestation filter. This filtering mechanism for attestation on large server is more flexible and may significantly increase the efficiency of attestation. The attestation architecture of filtering mechanism on large server is shown in Figure 3.

V. Prototype Validation

A. Architecture of Prototype Validation

We build trusted attestation environment on Linux by means of open source TPM Emulator and IAIK jTSS. Tomcat 6.0 is selected as Web server, free MySQL Server 5.1 as database. And MyEclipse 7.0 is chosen as IDE as it can provide Struts2.0 and Hibernate frameworks. The environment of prototype validation for filter-based trusted remote attestation is shown in Figure 4.

B. Software-based TPM Emulator

TPM Emulator enables not only the implementation of flexible and low-cost test-beds and simulators but, in addition, provides programmers of trusted systems with a powerful testing and debugging tool. Thanks to its portability and interoperability, the TPM emulator runs on a variety of platforms, including Linux, Mac OS X, and Windows and is compatible with the most relevant software packages and interfaces.

Running entirely in software, the TPM emulator can further be used to enhance virtual machines, thus enabling the execution of TPM-based software in a trustworthy virtualisation environment[8]. Software-based TPM Emulator can be downloaded from http://tpm-emulator.berlios.de/.

C. Trusted Computing for the Java Platform

Trusted Computing comprises multiple layers of hard and software. While the hardware consists of the Trusted Platform Module and related trusted building blocks, the main software components include the TPM hardware driver and a Trusted Software Stack (TSS). This TSS is typically developed in pure C and can therefore not directly be used from other languages such as Java.

For that reason, the Institute for Applied Information Processing and Communications (IAIK) at Graz University of Technology develops jTSS wrapper to provide language bindings for Java. The goal is to make the Trusted Service Provider Interface (TSPi) layer of the TSS stack available to Java developers in an object oriented fashion[9]. jTSS wrapper can be downloaded from http://trustedjava.sourceforge.net/.

D. Prototype Implementation

Privacy CA that was released by OpenTC is used for user identity attestation at first. The right of super administrators has to be used instead of virtual server in order to insure the filtering privilege. There are four tables in the server database, user information table, messages table, attestation trusted table and attestation management table respectively. The attestation management table is used for attesting user request for super administrators.

The filter is composed of MD5 function, trusted attestation table and database operation. The MD5 function of request parameter is based on Struts Action. The trusted attestation table is the foundation of filtering mechanism, it
is a refreshed list, and its records also can be deleted by super administrators.

Normally basic operation right is qualified to registered users, and therefore basic operations don’t need to be attested. Consequently basic operations are added into the trusted attestation table. The normal trusted attestation table is shown in Table 1.

<table>
<thead>
<tr>
<th>ID</th>
<th>User ID</th>
<th>Web ID</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2</td>
<td>3458D74D455BE38213C173C480C24E5E</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>3458D74D455BE38213C173C480C24E5E</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>3458D74D455BE38213C173C480C24E5E</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>2</td>
<td>0D5877E09361D389534B1D0C910D1DC9</td>
</tr>
</tbody>
</table>

When users want to do some actions that are not in the attestation trusted table, the Action will send the request to the attestation management table and user requests are attested by super administrator. If super administrator allows the request, the attestation trusted table is refreshed. The attestation trusted table after refresh is shown in Table 2.

<table>
<thead>
<tr>
<th>ID</th>
<th>User ID</th>
<th>Web ID</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2</td>
<td>3458D74D455BE38213C173C480C24E5E</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>3458D74D455BE38213C173C480C24E5E</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>3458D74D455BE38213C173C480C24E5E</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>2</td>
<td>0D5877E09361D389534B1D0C910D1DC9</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>2</td>
<td>0D5877E09361D389534B1D0C910D1DC9</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>6</td>
<td>0</td>
</tr>
</tbody>
</table>

VI. CONCLUSION
The remote attestation specifications based on the binary and property released by TCG have limitations of one-off and static attestation. That is to say, the attestation methods are not based on entity semantic attestation or entity behavior attestation. The known attacks can be prevented by TCG attestation specifications, but unknown attacks may not be detected.

The language virtual machine provides foundation for entity behavior attestation, and it is possible that the program behavior is attested in semantic level. As a result, attestation creditability is increased. But it is quite difficult to define semantic attributes of program. Furthermore, it is impossible that the defined semantic attributes cover with all of security characteristics.

The language virtual machine is replaced as the virtual mirror server of actual server, the program is first attested in the virtual server and then it runs in the actual server. In order to achieve this attestation policy the attestation filter has to be appended to server to carry out program filtration. The remote attestation based on filtering mechanism increases attestation efficiency effectively by building on the trusted attestation table.

The paper analyses the traditional remote attestation, point out their disadvantage, and in the basic of semantic remote attestation, proposed filter-based trusted remote attestation for Web services. We build a trusted environment through open-source TPM Emulator and IAIX jTSS on Linux, select Tomcat as Web server. The prototype of remote attestation based on filtering mechanism is validated by the attestation trusted table.

ACKNOWLEDGMENT
This work is Supported by the Natural Science Foundation of Shanxi Province under Grant No. 2009011022-2; Supported by Shanxi Scholarship Council of China Grant No. 2009-28.

REFERENCES