VOL. 11, NO. 7, APRIL 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

# AUTOMATED POLICY BASED REMOTE ATTESTATION IN TRUSTED COMPUTING

A. Saravanan[1], M. S. Irfan Ahmed[2] and S. Sathya Bama[1]
[1]Department of MCA, Sri Krishna College of Technology, Coimbatore, Tamil Nadu, India
[2]Department of MCA, Nehru Institute of Engineering and Technology, Tamil Nadu, India
E-Mail: a.saravanan21@gmail.com

**ABSTRACT**

With the rapid development of Internet and technologies, e-business flourished in almost all organisations. Progressively, organisations need to exchange and share data amidst their users as well as with other organisations. This data is often sensitive or confidential, and access to it desires to be secured. In this circumstance, trusted computing came in to existence which is a new security solution proposed by the Trusted Computing Group (TCG). It targets to provide an effective framework that allows distributed systems to ensure each other's integrity and trustworthiness. Several architectures exist to determine whether a remote system is trusted and to protect disseminated data. However, many approaches are static, inexpressive, or undermines the system security. This paper proposes an effective mechanism for remote attestation in trusted computing using automated policy negotiations that allows us to prove the integrity of a system and sensitive data.

**Keywords:** remote attestation, trusted computing, automated negotiations, trusted platform module, policy based attestation.

## INTRODUCTION

With the hasty technological development, more and more people depends on internet and web. As a result, the web applications as well as computer applications have pervaded in to all kinds of fields such as education, business, shopping and society. These applications require computer systems to guarantee confidentiality, availability, security, authenticity, integrity. However, the computer systems and network becomes much vulnerable to virus and malicious attacks. Thus there are some challenging issues exist to verify the integrity and trustworthiness of the computing system and platform. Even a perfectly secure operating system cannot verify its own integrity [1].

Also, businesses and social organisations require the sensitive data to be exchanged between employees and with other organisations. This becomes more important with employees in particular; when they use a mobile device in which the probability of theft is high that provides sensitive data in wrong hands. Privacy and business confidentiality requirements demand that only authorised people should be granted access to sensitive data, and the usage of sensitive data needs to be controlled even after the data has been disseminated to data consumers [2]. To provide a high assurance of system integrity and data protection, we must rely on a trusted hardware component (e.g., the Trusted Platform Module (TPM), whose specifications are defined by the Trusted Computing Group (TCG) [3]).

Attestation is one of the main functions in TPM. It allows a program to authenticate itself and is a means for one system to make reliable statements about the software it is running to another system. The remote party can then make authorization decisions based on those reliable statements. Thus the process of verifying integrity of remote systems using TPM is called remote attestation. However the verifier is required to maintain a large up-to-date database of acceptable software components that may exist on a data consumer's system which is a cumbersome task. By outsourcing the task to a specialist, we can reuse work done by the specialist verifier, thus reducing the cost of maintaining updates. Conversely, this complex implementation specifies only the decision regarding systems integrity [4]. A decision about the system's integrity is often not sufficient to ensure the expected behaviour of the software or a program [5]. Thus, an authorisation policy for accessing sensitive data is required to protect the data. This policy should stipulate whether remote attestation is required or not. Different organisations specify different constraints over system components in addition to verifying the systems integrity. The proposed architecture is designed to allow easy integration of secure applications with the remote attestation module.

The remainder of this paper is structured as follows: Next section provides an overview of trusted computing. Next, we suggest a model policy based remote attestation. Then we discuss about the related work. Finally, we give our conclusion and future work.

## TRUSTED COMPUTING

Trusted Computing is a new security solution proposed by the Trusted Computing Group (TCG) [3]. According to TCG, "An entity can be trusted if it always behaves in the expected manner for the intended purpose". "Trust" is a complicated notion that has been studied and debated in different areas (social science and humanities as well as computer science). A possible definition of the notion "trustworthy" is the degree to which the (security) behavior of the component is apparently obedient with its

stated functionality [6]. TCG works based on Trusted Platform Module [7].

TPM [7] is a microprocessor chip attached to the main board with an ability of creating and storing keys, providing cryptographic algorithms like RSA, SHA-1, HMAC and functions as digital signing, identity and integrity measurements to the host systems for hardware as well as software authentication. This chip can be used with any major operating system. The TC systems would cryptographically seal off parts of the computer that deal with data and applications and give decryption keys only to programs and entities that were judged to be trusted [3]. Each TPM chip contains an RSA key pair called the Endorsement Key (EK). The pair is maintained inside the chip and cannot be accessed by software. The Storage Root Key (SRK) is created when a user or administrator takes ownership of the system. This key pair is generated by the TPM based on the Endorsement Key and an owner-specified password.

A second key, called an Attestation Identity Key (AIK) protects the device against unauthorized firmware and software modification by hashing critical sections of firmware and software before they are executed. When the system attempts to connect to the network, the hashes are sent to a server that verifies that they match expected values. If any of the hashed components has been modified since last started, the match will fail, and the system cannot gain entry to the network.

Remote attestation [8] is one of the important functionalities provided in the trusted computing platforms. A terminal platform [9], host of the TPM, can attest to its description of characteristics to a remote party. To guarantee the trustworthiness and freshness, the description of characteristics needs to be signed by the TPM. Usually this signature is generated by using the Endorsement Key (EK). For promising the accuracy of the information and protecting the privacy of the host of the TPM, TCG first develops a solution using a trusted third party (Privacy CA) [10].

TPM sends the public AIK, signed by EK, to the privacy CA who checks its validity and issues a certificate for the AIK. The host/TPM is now able to authenticate itself with respect to the certificate. By negotiating securely with the Privacy CA, TPM gets an Attestation Identity Key (AIK) certificate from Privacy CA and signs the message by using the AIK instead of EK. However current methods of remote attestation suffer from many critical drawbacks. For example, the Privacy CA needs to be involved in all the transactions of the attestation.

In version 1.2, the TCG have developed a new method of obtaining a certified AIK known as DAA (Direct Anonymous Attestation). The DAA protocol is based on three entities namely the TPM platform, the DAA Issuer and the DAA verifier. The issuer is charged to verify the TPM platform during the first step called Join step and to issue DAA credential to the platform. The platform uses the DAA credential with the verifier during the next step called Sign step. This protocol also supports a blacklisting capability so that verifiers can identify attestations from TPMs that have been compromised [11].

Platform boot processes are amplified to allow the TPM to measure each of the components in the system (both hardware and software) and securely store the results of the measurements in Platform Configuration Registers (PCR) within the TPM. Developers may a system processes that use the PCR values in a TPM to identify unsafe configurations at system boot thereby preventing inadvertent network connection while in an unsafe mode. The typical components of TPM is shown in the Figure-1.

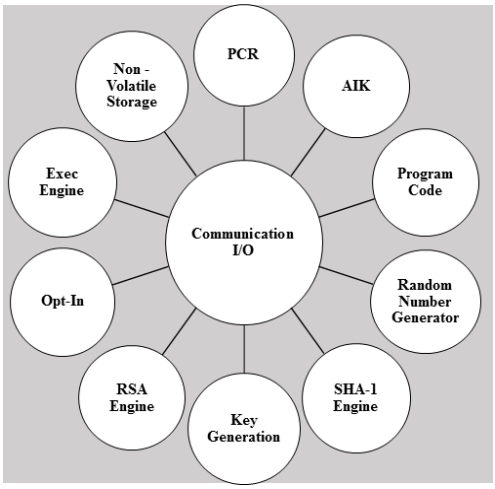


**Figure-1.** Components of trusted platform module.

**Integrity measurement architecture**

The runtime system of the device used by the requester must be attested to ensure its integrity. Attestation requires measurement of all components from the boot process up to the application layer. This is achieved by using the Integrity Measurement Architecture (IMA) [4]. All executable content, as well as application-related file content (configuration files, libraries, etc.) are measured (a SHA-1 checksum of the file is taken) before they are loaded. Each time a file is measured, a given PCR is extended with the new value, and this value is also added to the measurement list that is stored in the kernel. This measurement list is also accessible using the file system. To attest a platform, the value of PCR along with its measurement list is sent to the verifier (Attestation Authority) who can check the state of the software stack.

**Trusted boot**

A basic principle followed in trusted platform technologies is to verify the integrity or trust of every critical component before it is executed or loaded. A trust chain, starts from root of trust to hardware platform, operating system, and applications. The previous portion of code that is executed checks the integrity of the next component to be executed and passes trust, then trust can be extend into the whole computer system. The chain of

trust starts with the Core Root of Trust for Measurement (CRTM), which is a trusted code in the BIOS boot block. It reliably measures integrity values of other entities, and stays unchanged during the lifetime of the platform. The BIOS then measures hardware and the boot loader and passes control to the boot loader like Trusted Grub [12]. The boot loader measures the OS kernel image and passes control to the OS. Each step of the boot process extends the appropriate PCR value in the TPM with the measurements taken in that step. These measurements attest the integrity of the system. This process is shown in Figure-2.

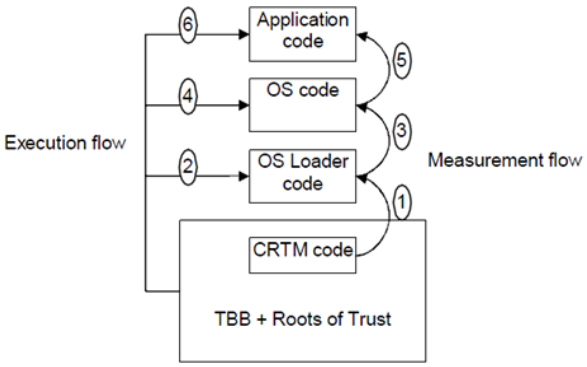


**Figure-2.** Trusted boot process.

**Remote attestation**

Remote attestation is one of the core functionalities provided by trusted computing platforms. It holds the promise of enabling a variety of novel applications. It is the process by which an application authenticates itself to a remote party. When asked to attest itself, Attester (the attesting party) reports to a remote party (the verifier) the configuration of a machine(s) to be attested through TPM. To guarantee integrity and freshness, PCR values and a fresh nonce provided by the remote party are digitally signed with an asymmetric key called Attestation Identity Key (AIK), which is under the sole control of the TPM. In this paper, an effective approach for remote attestation in trusted computing is presented. Our approach has the advantage of mitigating the possibility of granting access to systems with known vulnerabilities in a way that does not require the attestation authority to know the organisation policy. To ensure the correct functioning of the system, it is critical to ensure the integrity which must be verified during a secure boot procedure as mentioned in Schmidt *et al*. [13, 14, 15].

**PROPOSED ARCHITECTURE**

The proposed architecture introduces the remote attestation for integrity and access restrictions through policy enforcement engine for protected sensitive data. Figure-3 shows the proposed architecture. The goal of attestation is to prove to a remote party that your operating system and application software are intact and trustworthy.

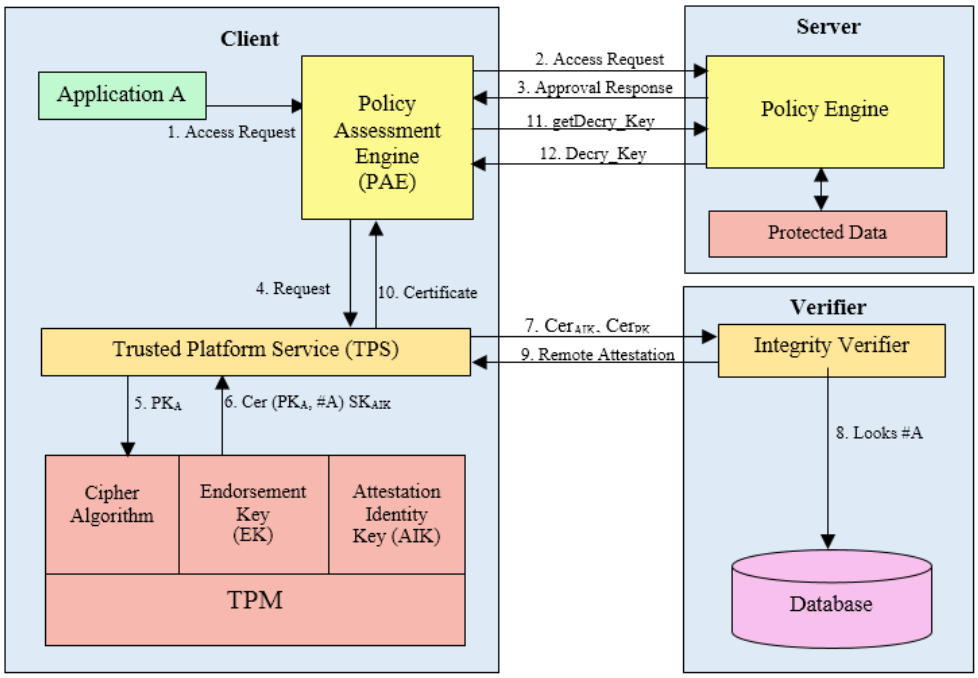


**Figure-3.** Proposed architecture of policy based remote attestation.

The verifier trusts that attestation data is accurate because it is signed by a TPM whose key is certified by the CA. A basic architecture works as follows:

1. The application "A" generates a public/private key pair $PK_A$ and $SK_A$ and sends the access request to the Policy Assessment Engine along with the keys.
2. The engine then encrypts the request and then forwards it to the server's Policy Engine.
3. Then the request will be assessed through authorisation policy on the server side for accessing sensitive data that specify whether remote attestation is required or not.
4. Based on the approval status, the PAE asks the TPS to certify it.
5. The TPM computes a hash value #A of the executable code of application "A".
6. The TPM then creates a certification including $PK_A$ and #A and signs it with the attestation identity key $SK_{AIK}$.
7. When application "A" wishes to authenticate itself to a remote party, TPS sends the cert. of its public key and hash value #A along with a cert. issued to the TPM by a trusted certification authority (CA).
8. The remote party verifies the certificate chain. And looks #A up in a database which maps hash values to trust levels.
9. If application "A" is deemed trustworthy, the attestation authority can issue a certificate for the successful integrity check of the data consumer's system.
10. PAE obtains the required credentials from TPS.
11. PAE then requests the decryption key from the policy engine, which verifies the credentials and check whether the access is authorised.
12. If it is authorized, the decryption key will be issued to the PAE for further access.

The process in the above framework is depicted in sequence diagram in Figure-4.

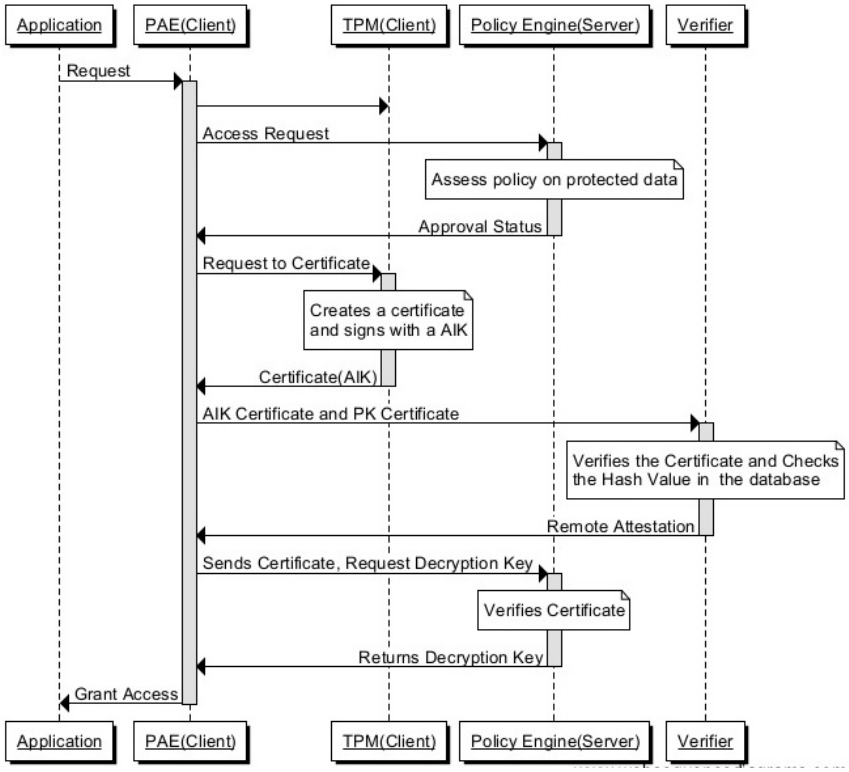


**Figure-4.** Sequence diagram for the proposed method.

## RELATED WORK

Remote attestation is an integral part of Trusted Computing and the ability to verify the software and/or hardware running on a machine is of paramount importance. There has been however, some criticism about whether this is a viable or impractical solution, since it requires the attesting authority (or verifier) to know apriori the needed software configurations as well as their checksums. Lyle *et al*. show that this is indeed a viable solution for web services [16]. The Integrity measurement architecture (IMA) proposed by Sailer *et al*. [4] uses binary attestation to measure all the programs and code before they are loaded into the system to run. This architecture was extended by Jaeger *et al*. in [17], where

the proposed architecture only measures the programs and code that are needed by the verifier. Sadeghi *et al*. [18] provide reasons as to why binary attestation may not be the most useful form of attestation. Sandhu *et al*. [19, 20] and Gowadia *et al*. [21] have described various security architectures to protect disseminated or shared data.

Software platform cannot guarantee that the software it is running is reliable, it is advisable to move the required program code away from the untrusted platform. The technique used for this is called code-splicing, which involves splitting the program code into "critical" and \non-critical" sections, so that only non-critical code is run on the untrusted platform. This in turn guarantees that code that is critical has not been tampered with. These ideas have been presented by Ceccato *et al*. [22], Dvir *et al*. [23] and Zhang *et al*. [24]. Kennell *et al*. proposed a system called "Genuinity", which verifies if the hardware and the software running on a system are "genuine" [25]. This is achieved without the use of any special hardware and by using a timed execution of a checksum function that provides a fingerprint of the running applications. The time taken to execute the checksum is verified by the verifier. As shown by Shanker *et al*. [26], this is not a viable solution due to the assumptions that are to be imposed and it is also prone to substitution attacks. A timing based remote attestation mechanism to prove the integrity of the system was proposed by Schellekens *et al*. [27]

Alawneh *et al*. propose architecture to protect data within an organisation [28]. The proposed system binds the sensitive content within the organisation to specific devices, thereby restricting the content from being leaked to other devices. There have been several techniques that have been proposed for property-based attestation [29, 30]. In [31] the authors have proposed the idea of a Property Manifest, which can be used to define security policies for policy-based attestation. Haldar *et al*. proposed the concept of semantic remote attestation in [5], wherein the security of the system is guaranteed through program analysis. In [32], Yu *et al*. propose a system for guaranteeing the freshness of the integrity measurement that is used in the attestation. Researcher [33] on the meaning and method of remote attestation, divide the abstract models into four types and analyze the shortcoming of the existing models and the improved method. A more general extension to the binary attestation is property-based attestation [34], which should determine whether the target machine to be attested fulfills certain requirements. The authors [35] propose model-driven remote attestation that is attesting remote system from behavioural aspect. The literature [36] studied the methods of establishing the anonymity of trusted terminal platform based on TPM chip.

## CONCLUSIONS

In this paper, we have presented a policy based approach that allows us to prove the integrity of a system through remote attestation over protected data. It allows an organisation to provide their authorisation policies on data with different security levels. This system relies on a trusted hardware component, the Trusted Platform Module and uses the Integrity Measurement Architecture. In the future, we would like to evaluate our system with different policy-based data protection frameworks, different platform as well as with different attestation mechanisms.

## REFERENCES

[1] Ahmad-Reza Sadeghi. 2008. Trusted Computing - Special Aspects and Challenges, SOFSEM 2008, High Tetras, Slovakia. pp. 98-117.

[2] Gopalan A., Gowadia V., Scalavino E. and Lupu E. 2012. Policy driven remote attestation. In Security and Privacy in Mobile Information and Communication Systems, Springer Berlin Heidelberg. pp. 148-159.

[3] The Trusted Computing Group. http://www.trustedcomputinggroup.org.

[4] Sailer R., Zhang X., Jaeger T. and Van Doorn L. 2004. Design and Implementation of a TCG-based Integrity Measurement Architecture. In USENIX Security Symposium. 13: 16-16.

[5] V. Haldar, D. Chandra and M. Franz. 2004. Semantic remote attestation - a virtual machine directed approach to trusted computing. In USENIX Virtual Machine Research and Technology Symposium. pp. 29-41.

[6] Benzel T.V., Irvine C.E., Levin T.E., Bhaskara G., Nguyen T.D., Clark P.C. 2005. Design principles for security. Technical Report NPS-CS-05-010, Naval Postgraduate School.

[7] TCG, TCG TPM Specification Version 1.2 Revision 103, https://www.trustedcomputinggroup.org/specs/TPM/.

[8] Joshua Guttman, Amy Herzog, Jon Millen, Leonard Monk, John Ramsdell, Justin Sheehy, Brian Snien, George Coker, NSA, Peter Loscocco, NSA. Attestation: Evidence and Trust, Mitre Technical Report, MTR080072.

[9] Jiqiang Liu Jia Zhao Zhen Han. 2008. A Remote Anonymous Attestation Protocol in Trusted Computing, The 4th international Workshop on Security in Systems and Networks (SSN2008). 22nd

IEEE International Parallel and Distributed Processing Symposium.

[10] Privacy CA. http://www.privacyca.com/.

[11] E. Brickell, J. Camenisch and L. Chen. Direct anonymous attestation. In: Proceedings of the 11th ACM conference on Computer and communications security, CCS '04, pages 132-145, New York, NY, USA, 2004. ACM.

[12] Trusted Grub. http://sourceforge.net/projects/trustedgrub/.

[13] U. Schmidt, A. Leicher, I. Cha and Y. Shah. 2010. Trusted platform validation and management. International Journal of Dependable and Trustworthy Information Systems (IJDTIS). 1(2): 1-31.

[14] Trusted Network Connect. http://www.trustedcomputinggroup.org/files/resource_files/51F9691E-1D09-3519-AD1C1E27D285F03B/TNC_Architecture_v1_4_r4.pdf.

[15] Huang X. and Peng Y. 2009. An Effective Approach for Remote Attestation in Trusted Computing. In WISA 2009: Proceedings of the 2nd International Symposium on Web Information Systems and Applications. pp. 80-83.

[16] J. Lyle and A. Martin. 2009. On the feasibility of remote attestation for web services. In: Computational Science and Engineering, 2009. CSE '09. International Conference on. 3: 283-288.

[17] T. Jaeger, R. Sailer, and U. Shankar. 2006. Prima: Policy-reduced integrity measurement architecture. In: Proceedings of the eleventh ACM symposium on Access control models and technologies, SACMAT '06, pp. 19-28, New York, NY, USA.

[18] A.-R. Sadeghi and C. Stuble. 2004. Property-based attestation for computing platforms: caring about properties, not mechanisms. In: Proceedings of the 2004 workshop on new security paradigms, NSPW '04, pp. 67-77, New York, NY, USA.

[19] J. Park, R. S. Sandhu and J. Schifalacqua. 2000. Security architectures for controlled digital information dissemination. In: Proc. of ACSAC, p. 224.

[20] R. S. Sandhu, K. Ranganathan and X. Zhang. 2006. Secure information sharing enabled by Trusted Computing and PEI models. In ASIA CCS, pp. 2-12.

[21] V. Gowadia, E. Scalavino, E. C. Lupu, D. Starostin and A. Orlov. 2010. Secure crossdomain data sharing architecture for crisis management. In: Proceedings of the tenth annual ACM workshop on Digital rights management, DRM '10, pp. 43-46, New York, NY, USA.

[22] M. Ceccato, M. Preda, J. Nagra, C. Collberg and P. Tonella. 2007. Barrier slicing for remote software trusting. In Source Code Analysis and Manipulation, 2007. SCAM 2007. Seventh IEEE International Working Conference on. pp. 27-36.

[23] O. Dvir, M. Herlihy and N. Shavit. 2005. Virtual leashing: Internet-based software piracy protection. In Distributed Computing Systems. ICDCS 2005. Proceedings. 25th IEEE International Conference on. pp. 283-292.

[24] X. Zhang and R. Gupta. 2003. Hiding program slices for software security. In: Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization, CGO '03, Washington, DC, USA, IEEE Computer Society. pp. 325-336.

[25] R. Kennell and L. H. Jamieson. 2003. Establishing the genuinity of remote computer systems. In: Proceedings of the 12th conference on USENIX Security Symposium, Berkeley, CA, USA. 12: 21-21.

[26] U. Shankar, M. Chew and J. D. Tygar. 2004. Side effects are not suffient to authenticate software. In: Proceedings of the 13th USENIX Security Symposium. pp. 89-101.

[27] D. Schellekens, B. Wyseur, and B. Preneel. 2008. Remote attestation on legacy operating systems with trusted platform modules. Sci. Comput. Program. pp. 3-22.

[28] M. Alawneh and I. M. Abbadi. 2008. Sharing but protecting content against internal leakage for organisations. In DBSec. pp. 238-253.

[29] Chen, L., Landfermann, R., Löhr, H., Rohe, M., Sadeghi, A. R. and Stüble C. 2006. A protocol for property-based attestation. In: Proceedings of the first

ACM workshop on Scalable trusted computing. pp. 7-16.

[30] A. Nagarajan, V. Varadharajan, M. Hitchens and E. Gallery. 2009. Property based attestation and trusted computing: Analysis and challenges. In NSS. pp. 278-285.

[31] A. Nagarajan, V. Varadharajan, M. Hitchens, and S. Arora. 2008. On the applicability of trusted computing in distributed authorization using web services. In DBSec. pp. 222-237.

[32] A.Yu and D. Feng. 2010. Real-time remote attestation with privacy protection. In: Proceedings of the 7th international conference on Trust, privacy and security in digital business, TrustBus'10, Springer-Verlag. pp. 81-92.

[33] ZHANG Qiang, ZHU Li-na, ZHAO Jia. 2008. Research on Method of Remote Attestation in Trusted Computing, Control and Management, Microcomputer Information. 24(4).

[34] Kühn U., Selhorst M., Stüble C. 2007. Property-Based Attestation and Sealing with Commonly Available Hardand Software. In: ACM-STC.

[35] Liang Gu, Xuhua Ding, Robert H. Deng, Yanzhen Zou, Bing Xie, Weizhong Shao, Hong Mei. 2008. Model-Driven Remote Attestation: Attesting Remote System from Behavioral Aspect. The 9th International Conference for Young Computer Scientists, Zhang jiajie, China.

[36] Y. Ai-Ming, C. Xiao-Bo and F. Deng-Guo. 2010. Research of Platform Anonymous Identity Management Based on Trusted Chip. Chinese Journal of Computers. 33(9).