TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.1, January 2014, pp. 718 ~ 723
DOI: http://dx.doi.org/10.11591/telkomnika.v12i1.3192

A Formal Model of Trust Chain based on Multi-level Security Policy

Kong Xiangying*[1,2], Zhuang Yi[2], Wang Xiaoxing[2]
[1]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China
[2]Jiangsu Automation Research Institute, Lianyungang China
*Corresponding author, e-mail: Kongxy716@aliyun.com

Abstract

Trust chain is the core technology of trusted computing. A formal model of trust chain based on finite state automata theory is proposed. We use communicating sequential processes to describe the system state transition in trust chain and by combining with multi-level security strategy give the definition of trust system and trust decision theorem of trust chain transfer which is proved meantime. Finally, a prototype system is given to show the efficiency of the model.

Keywords: *Trust, Trust Chain, Finite State Automata, Multi-level Security Policy*

1. Introduction

Trust chain transfer is one of the key ideas of trusted computing. It builds a trust chain based on Trusted Platform Module (TPM) which is independent from CPU control and must be credible as trust root [1]; that is to say, before running any modules in the operating system, you must establish the trust of these modules' code. Presently studies on trust chain transfer can be mainly divided into two categories: technical realization and model theory. In technology realization research, trust chain transfer has got in-depth studied and achieves greater development. Trusted Computing Group provides integrity measurement scheme before loading and based on this [1], Maruyama et al. give trust transfer from Grub to the operating system [2].Sailer et al. achieve trust transfer from operating system to application code on Linux platform [3];Tian Li-ye et al. show a trust chain scheme with independent TPCM cryptographic algorithm [4]; Li Xiaoyong et al. establish dynamic multi-path trust transfer based on the characteristics of software [5], and it can adopt different ways and policy according to different types of software under Windows platform to control loading and running of software. However, integrity is only one of the trusted properties of software, and the integrity in loading is not identical to the trust in running. In model theory research, Li Li et al. apply temporal logic to model the trust chain and theoretically verify the transitivity of trust chain [6], but the trust theory is still entity-based integrity measure-ment; in papers non-interference is implemented to build trust chain transfer model [7-9], and yet, non-interference is hard to verify instrumentally. Meanwhile, Donglai Fu et al. modify the command TPM_CertifyKey and its authentication properties and verify on TPM emulator [10]. In addition, some scholars apply fuzzy math [11, 12], evidence theory [13, 14], behavior trac e[15], process algebra and so on to model and analyze the credibility of software [16, 17], but trust property of software in these results is a little different from that defined by TCG .

TCG gives definition of trust is: "an entity is trust if its acts always reach the desired goal in expected way". From the behavioral point, TCG gives the definition and emphasizes behavioral predictability and controllability, which implicates compliance with predefined rules. In this paper, based on finite state automata theory, we use multi-level security policy to establish a formal analysis method of trust chain. The paper is organized as follows: Part II is the definition of trusted entities; Part III proposes a trust system formal model (we call FSM-MSP) based on finite state automata and multi-level security policy; Part IV defines trust chain transfer decision theorem; finally, a prototype system designed shows the efficiency of the model.

Received May 13, 2013; Revised July 10, 2013; Accepted August 18, 2013

2. Trusted Entity in Trust Chain Transfer

The multi-level security strategy (MSP) is currently the most widely used policy in a variety of security systems. Its main idea is that subject could not read high security level object or write low security level object. Thus, it protects the unauthorized confidential information from leaking. Applying MSP to the trust chain transfer process can effectively control the transfer rules between different security level subject and object.

Trust chain is transferred with software entity as basic unit. TCG emphasizes entity behavior predictability in entity trust definition, but software behavior depends on its operation, which can be simply divided into Read (get the value of the object) and Write (change the value of the object). As the function of modern software varies complexly, a program running need not its code loaded but corresponding dynamic library and may also require invoking system call, accessing data file on external storage (disk) and interacting with user. We abstract these software entities in different types as a set of attributes and corresponding data value.

After system boot, entities will be constantly updated under their operation. For any entity x, y, $x\ Read\ y$ means entity x reads attribute and its value information from entity y. Hence, the trust of entity is predictability of the entity behavior. It can covert to judgment on whether the entity operation meets the agreed rules. The convention of entity state under its operation can be simulated by finite state automata theory, and the rules between operation and the entity can be described by multi-level security policy. In the next section, we present the trusted system formal model FSM-MSP based on the finite state automata combined with multi-level security strategy..

3. FSM-MSP MODEL

We now give the definition of FSM-MSP model.

Definition 1. A FSM-MSP system is a seven-tuple $(E,\ OP,\ S,\ s_0,\ T,\ P,\ \succ)$, where

- E is the set of entities, namely software entities. They performance as processes or files in system. They can be divided into subject set and object set according to operation senders and actors. They are starts or endings in information flow allowed in system.
- OP is the set of operations, which are information exchanging ways between entities and contain Read and Write.
- S is the set of states, with an initial state $s_0 \in S$, which denotes the state of TCM starts when the system powers on. s_0 is trusted in accordance with TCG specification.
- T denotes state transition function, which is used to describe the change of system state caused by entities perform operation. It is defined as $E \times OP \times E \times S \to S$.
- P is a lattice with an expres-sion $(A, \leq, \wedge)$, which defines the constraint security policy with certain security level information exchange between the entities.
- $\succ$ is an information relation in entities. For any $x, y \in E$, $x \succ y$ denotes information can flow from x to y.

For the convenience of description, we make the following convention:

For any $x \in E$, $s \in S$, $s.l(x) \in A$ denotes the security level of x in the state s. $s.x$ denotes x in the state s.

For any $x, y \in E$, $op \in OP$ and $s, s' \in S$, $x \overset{s}{\succ} y$ denotes execute information flow form x to y under state s. $x \xrightarrow[s]{op} y$ denotes the state change from s to s'.

Definition 2. A state s is trusted iff

$\forall x, y \in E$, $s \in S$, we have $x \overset{s}{\succ} y \Rightarrow s.l(x) \leq s.l(y)$ (1)

It has been showed above that the initial state s_0 of TCM starts is trusted when the system powers on.

Definition 3. The state transition function T is trusted if $\forall s \in S$, $\forall op \in OP$,and $\forall x, y \in E$, and they meet:

1) $s' = T(x\ op\ y, s)$,
2) $\forall e \in E$, $s.l(e) \le t.l(e)$,
3) $\forall u, v \in E$, $u \overset{s'}{\succ} v \Rightarrow s'.l(u) \le s'.l(v)$,
4) If $op = Read \wedge s \ne s'$,then $s.l(y) \le s.l(x)$
5) If $op = Write \wedge s \ne s'$,then $s.l(x) \le s.l(y)$
6)

Definition 4. A trace of entity is the sequence of operations acted on the entity in accordance with the order of acted time.

The record criteria on the trace of entity is that if the entity in state s acts an operation $x\ Read\ y$, then record $s.x$ into the trace of y , however, if it acts $x\ Write\ y$, then record $s.y$ into the trace of x . The trace of entity reflects the changing process of entity.

Definition 5. $\forall e \in E$, $e = (A, \tau e, \delta)$, where A is the attributes of the entity, $\tau e \in (E)*$ is a map: $\delta : (\tau e)* \to A$. If it is an empty sequence $<>, \delta(<>)$ is the original form of entity, namely the form of entity before operations acted on after the system booted.

βe denotes the real trace of entity e after the system booted, where $\beta e \in \tau e$.

the decision theorem of trust chain

Before provide the decision theorem of trust chain, we give the specific rule of state transition function.

Definition 6. state transition function $T : OP \times S \to S$.

$\forall s, s' \in S$, $\forall op \in OP$, $\forall seq \in (E \times OP \times E)*$, we have:

- $T(\Lambda, s) = s$,where Λ is an empty operation;
- $T(x\ op\ y, s) =$

If $op = Read$ then

If $(s.l(y) \le s.l(x) \wedge \forall s", t \in \beta y(s".l(s".t) \le s.l(x)))$

Then $y \xrightarrow[s']{op} x$ else s

Else If $(s.l(x) \le s.l(y) \wedge \forall s".t \in \beta x(s".l(s".t) \le s.l(y)))$

Then $x \xrightarrow[s']{op} y$ else s

We have given the definition of state trust above the context. From micro point of view, the system at any time can only perform one operation, and as each operation corresponds to a transition function, therefore, if the system state s is trusted, the transition function corresponded to the currently executing operation is trusted, and furthermore, the state converted by the executing actions is trusted.

Trust chain transfer is the transmission of the system control, which performs as the system loads software from the hard disk or switches process caused by scheduling. These are all done by performing a series of operation so that the system changes meantime. We concern that in the process of the system changing from the state s (prepare to transfer) to the state s' (finish transferring), if the system is trusted in the state s , and also in the series states of the whole changing process, then we consider that trust chain has finished a trusted transmission. If all the trust chains transfers credibly, therefore, the whole system is trusted.

We use x, y denote respectively two entities corresponding to two successive states $s, s' \in S$ in the trust chain transfer. $op_1 op_2 ... op_n$ are the executing operations between the two states, where $op_i \in OP$, . $i = 1..n$. $s_1 s_2 ... s_{n-1}$ are the middle states between s and s'.

$$\begin{aligned} s_1 &= T(x\ op_1\ y, s), \\ s_i &= T(x\ op_i\ y, s_{i-1}), \\ s' &= T(x\ op_n\ y, s_{n-1}), \\ where\ \ i &= 2..n-1 \end{aligned} \tag{2}$$

Now we propose the definition of trust transfer and trusted system.

Definition 7. For the trust transfer above, we consider the trust chain of the system meets credibly transitive when the system fulfills the conditions as follow:

1) the state s trusted;
2) the state transition function T is trusted.

The definition of state transition function reflects the credibility transmission between entities. In other words, operations in violation of the security policy defined by the system never occur in the trust transferring.

Definition 8. A system is trusted iff

1) the initial state s_0 is trusted;
2) for any reachable state s, s is trusted.

The definition is intuitive and understand-able, but could not reflect the intrinsic mechanism of trust transfer or the internal relation of state transition (namely trust chain transfer). Therefore, we proposed Theorem 1 based on FSM-MSP model to judge if a system meets the requirements of a trusted system.

Theorem 1. A system simulated by FSM-MSP model is trusted.

Proof: That is to proof the conditions of Definition 8.

For condition 1), we have pointed out s_0 is TPM. TPM is trusted according to the definition of TCG, so the condition is clearly established;

For condition 2), state transition function T is trusted in accordance with Definition 3. Now we use inductive method to proof. Under the effect of trusted state transition function, the state from s_0 to all reachable s are trusted, and the states sequence of them can be expressed as $s_0 \circ \sigma$, where σ denotes states sequence, $\circ$ denotes concatenation and $s_0 \circ \sigma$ denotes the states sequence after s_0.

(1) when the length of σ is 0, then $s_0 \circ \sigma$ is clearly securable;

(2) when the length of σ is n, $n > 0$, s' is the last state of $s_0 \circ \sigma$, and s' is trusted, and if operation $x\ op\ y \in E \times OP \times E$ occur before s', then $T(x\ op\ y, s')$ is trusted can be deduced from Definition 2 and 3.

Above all, the system meets the two conditions of Definition 8, so the Theorem 1 is true on the basis of inductive method.

4. The Design of Prototype System and Verify Model

Based on FSM-MSP model, we implement a trust transferring prototype system on the Linux platform. System previously defines the system dynamic link library and the data files each program allowed to access, and these multilevel security policies are stored on disk and signed by TPM for protection. We, through modifying the system function do_fork (), add a callback function. The function not only checks the integrity of the code, but according to the predefined security policy, verifies the trust of the operation to determine whether the process

has access in violation of security policy, so as to ensure the trust of all reachable states of the system. The code skeleton changes are shown in Figure 1.

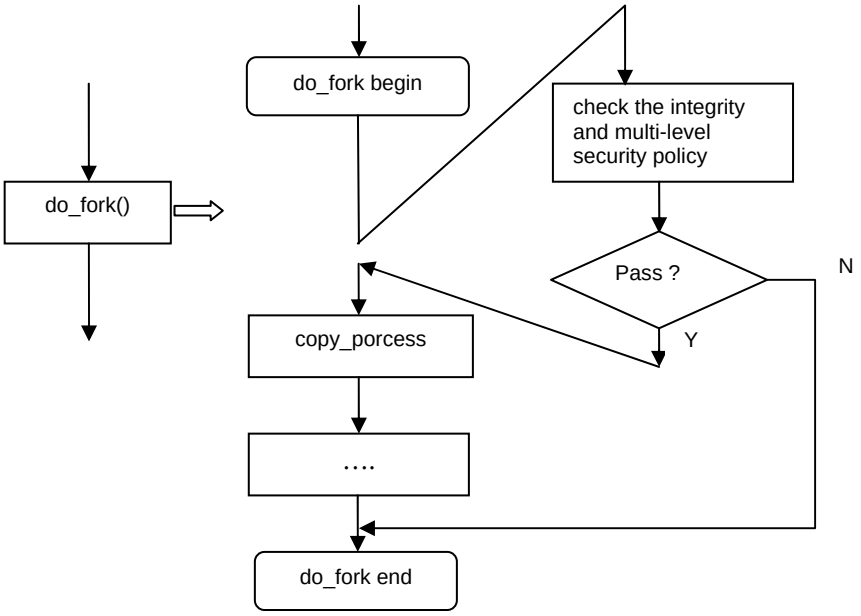

Figure 1. The schematic diagram of modifying do_fork

The system configuration is Pentium IV 2.4 G Hz of CPU and 1 GB of memory. The system startup time test results show that integrity measures increase cost by about 13%. The additional time cost mainly lies in the multi-level security policy check.

5. Conclusion

In this paper, we proposed FSM-MSP model based on the finite state automata combined with mutil-level security policy. Finite state automata theory can describe state transition in a trust chain, while multi-level security policy can depict rule restrictions in the process of trust chain transfer, thus the existing achievements of multi-level security policy engineering can be applied in the trust chain transfer to indentify the entity credibility, so that this can effectively solves the problem that current trust chain theory model is difficult to be engineered.

References

[1] Trusted Computing Group, TCG Specification Architecture Overview Specification Revision 1.2 [EB/OL]. 2007, http://www.trustedcomputing.org.
[2] Hiroshi Maruyama. Taiga Nakamura, Seiji, et al. *Linux with TCPA integrity measurement*. IBM, 2003, Tech Rep: RT0575.
[3] Reiner Sailer, Xiaolan Zhang, et al. *Design and implementation of a TCG-based integrity measurement architecture*. The 13th Usenix Security Symposium, San Diego, 2004.
[4] Tian Li-Ye, Shen Chang-Xiang. Productive information system oriented trust chain scheme. *Telkomnika*. 2012; 10(5): 1093-1100
[5] Li Xiaoyong, Han Zhen, and Shen Changxiang. Transitive Trust and Perform-ance Analysis in Windows Environment. *Journal of Computer Research and Development*. 2007; 44(11): 1889-1895.
[6] Li Li, Zeng Guo, Sun Chenbo. Tempora-logic-based Model for Chain of Trust of Trusted Platform. *Computer Science*. 2008; 35(4): 265-267.
[7] Zhao Jia, Shen Changxiang, Liu Jiqiang, and Han Zhen. A Noninterference-Based Trusted Chain Model. *Journal of Computer Research and Development*. 2008; 45(6): 974-980.

[8] Zhang Xing, Huang Qiang, Shen Chang-Xiang. A Formal Method Based on Noninterference for Analyzing Trust Chain of Trusted Computing Platform. *Chinese Journal of Computers*. 2010; 33(1): 74-81.
[9] Kong Xiangying, Zhuang Yi. Research on Trust Chain Transfer Model Based on Dynamic Intransitive Non-interference. *JCIT: Journal of Converg-ence Information Technology*. 2012; 7(21): 157-163.
[10] Donglai Fu, Xinguang Peng, Yuli Yang. Authentication of the Command TPM_CertifyKey in the Trusted Platform Modlue. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2012; 10(2): 855-863.
[11] Tang Wen, Chen Zhong. Research of Subjective Trust Management Model Based on the Fuzzy Set Theory. *Journal of Software*. 2003; 14(8): 1401-1408.
[12] Tang Wen, Hu Jian Bin, Chen Zhong. Research on a Fuzzy Logic-Based Subjective Trust Management Model. *Journal of Computer Research and Development*. 2005; 42(10): 1654-1659.
[13] Yuan Lulai, Zeng Guosun, Wang Wei. *Trust Evaluation Model Based on Dempster-Shafer Evidence Theory*. 2006; 52(5): 627-630.
[14] Huo Ying, Zhuang Yi, Xue Yu, Method of Fuzzy Evaluation Based on Group Consistency Intensity. *Control and Decision*. 2011; http://www.kzyjc.net:8080/CN/abstract/abstract12446.shtml
[15] Que Yanwen. Software Behavior. Publishing House of Electronics Industry, Beijing, China. 2004: 52-68.
[16] Josang A. *An Algebra for Assessing Trust in Certification Chains*. In: Proceedings of the Network and Distributed Systems Security (NDSS'99) Symposium, 1999. The Internet Society. San Diego.
[17] Fu Ning, Zhou Xingshe, Zhan Tao. QPi: A Calculus to Enforce Trustworthiness Requirements. *Journal of Computer Research and Development*. 2011; 48(11): 2120-2130.