Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 14, no. 5, May 2018, pp. 1014-1022
DOI: 10.23940/ijpe.18.05.p20.10141022

A Method of Dynamically Associating Behavior Risks based on Time Thread in Smartphones

Zhenliu Zhou[a,*], Xiaoming Zhou[b], Weichun Ge[b], Yueming Pan[c], and Yu Gu[c]

[a]*Shenyang Key Laboratory of Information Security for Power System, Shenyang Institute of Engineering, Shenyang, 110136, China*
[b]*State Grid Liaoning Electric Power Co., Ltd, Shenyang, 110004, China*
[c]*State Grid Liaoning Electric Power Co., Ltd, Jinzhou, 121000, China*

Abstract

Behavior associated risks are analyzed and detected based on a series of software behaviors and user behaviors in a smartphone. Based on time threads, a method of dynamically associating and analyzing risks of behaviors is proposed. According to the time sequence of the behavior occurrence, the behaviors are organized into a behavior associated graph with time threads. Associated analysis and detection of risks among behaviors are achieved through matching association rules. The advantage of this method is that it cannot only realize dynamic analysis while behavior occurs, but can also realize static postmortem analysis using collected data sets. The dynamic association of behavior risk based on time thread improves behavior risk analysis and realizes the real-time risk detection in smartphones. Formalized definitions of behavior and behavior associated graphs are presented. Algorithms of behavior risk associating are described, and the results of the experimental analysis are given.

Keywords: behavior risk; behavior association; association analysis; smartphone security

(Submitted on February 21, 2018; Revised on March 26, 2018; Accepted on April 29, 2018)

1. Introduction

With the rapid development of mobile Internet technology, smart phones are used more widely in people's daily life and work. More and more people use smart phones to achieve mobile Internet, mobile payment and mobile office. The subsequent security problems of smart phones are also more prominent, such as personal privacy security, mobile payment security and mobile office security [6,9], which involve both personal information security and enterprise information security.

As the most widely used mobile phone operating system in smart phones, the Android system and its security have become a main focus for researchers. In addition to the security mechanism of Android's sandbox and authority control, detection of malware has become the main security research focus of Android mobile [1,3,11,16]. With the maturity of trusted computing technology, the software's trusted measurement technology is used in the security research of smart phones [4,8,14,15,17]. The trusted measurement of software properties and software behavior can guarantee the trustworthiness of APP and its behavior in smartphones. The way of restricting APP's behavior by permission control is usually limited by the user's neglect or vague understanding, which cannot achieve the purpose of restricting APP's malicious behavior. Therefore, depending on the system, as opposed to the users, to detect and restrict the behavior of the software is an effective way to enhance the security control of smartphones [2,10,13].

There are two main methods for the detection of software behavior: static analysis and dynamic detection [2]. However, due to its complexity and uncertainty, software behavior detection results usually interfere with user operations, and ultimately, users need to decide which should be accepted or rejected. Because of a lack of professional knowledge or worrying about the normal function, users usually adopt a passive acceptance strategy. This method is not effective in

* Correspond author.
E-mail address: zhouzl@sie.edu.cn

practical applications. For example, the mobile phone business software usually requires users to grant permission to read address book message. Users usually allow such permission requests, but they do not know why the APP needs to read the address book. If such permission requests are granted, what consequences will it bring? If permission is not granted, what consequences will it cause? Users are completely blind to the permission request because they worry that denial of authorization will cause the APP function exception. Similarly, if such a permission request is sent through a disguised and malicious APP, most users will also allow the operation instead of rejecting the request.

In addition, it is also inadequate to forecast security risk only through the detection of software behavior. In some cases, a single behavior does not necessarily constitute a risk, but it needs to be combined with a variety of other behaviors, including user behaviors, to make a risk judgment. For example, a cell phone receives SMS messages with URLs, and the mobile user clicks to access the website link, which can lead to a security risk (assuming the link is malicious). Another example is when the mobile phone connects via shared open WiFi and the user visits a financial payment website or accesses an enterprise's mailbox, which requires inputting his account and password. This series of software behaviors and user behaviors would lead to a serious security risk.

In order to detect behavior security risks of smartphones more accurately, this paper associates software behavior and user behavior together in mobile phones to conduct risk detection and analysis. Association analysis is a time-consuming task, usually with high computational cost [5,12]. This paper proposes a method based on time thread to dynamically associate behavioral risk. According to the time sequence of the behavior occurrence, the behaviors are organized into a behavior associated graph with time thread, and associated analysis and detection of risks among behaviors are achieved through matching association rules. The advantage of this method is that it cannot only realize dynamic analysis while behavior occurs, but can also realize static postmortem analysis using collected data sets. Using time thread improves the efficiency of behavioral risk association analysis. The formal description of behavior is defined, the construction algorithm and pruning algorithm of the behavior association graph are described, and results of the experimental analysis are given.

2. Software behavior and user behavior

Software behavior refers to the evolution of the performance form and state of the software running process. The behaviors of software can be divided into functional behavior and non-functional behavior. Functional behaviors include services and functions provided by the software. Non-functional behaviors include performance characteristics of the software, such as logging, performance optimization, and so on. Normally, functional behaviors of software are predictable. If unanticipated functional behavior occurs while the software is running, it can be considered that such unanticipated behavior is a risky behavior or malicious behavior. User behavior refers to the user's physical operation response to the software on a smartphone, such as inputting text, clicking a button, and so on. The security of user behavior is an important factor that affects the security of computer systems, network systems and software systems.

Table 1. APP's malicious behaviors

Type of behavior	Behavioral expression
malicious chargeback	Automatically order mobile value-added service
	Automatically order all kinds of charge business
	Automatically filter bill SMS message
privacy theft	Get the contents of the address book, SMS, and call records
	Access to the installed software, accounts and passwords information
malware propagation	Send SMS, MMS, mail, etc. containing malicious code
	Download and copy malicious code
Tariff consumption	Traffic generated by an automatic access network
	Automatically mass spam messages, MMS, mail, etc.

The risk of a single software behavior mainly depends on the type of behavior to determine whether it is malicious or not. Four types of typical malicious behaviors of APP software in smartphones are summarized in Table 1. Assessing risks of user behavior is relatively complex. It is difficult to assess risks of an independent user behavior, but if it is combined with time, network environment and other contextual clues, it is much easier to judge its risks. User behaviors, which are closely related to risk, can usually be summarized as calling, reading messages, sending messages, accessing the network, downloading APPs, installing and running APPs, and so on. For example, the risk of a user's phone call depends on whether the number of the phone call belongs to a swindle phone or a phage phone. When a user accesses his enterprise mailbox, if the mobile phone network connection is on the 3G/4G network, the behavior is considered to be safely. If the mobile phone network connection is free and is on an open-sharing WiFi network, then there may be a risk of leakage of information.

There are independent risks that exist in software behavior and user behavior, but there are new and enhanced risks between them. This paper mainly focused on those associated risks among software behaviors and user behaviors in a smartphone.

3. Formal description of behavior

The concept of security ontology was proposed by Donner [7] in 2003. Security ontology refers to "a set of descriptions of the most important concepts and the relationships among them." Because of the advantages of ontology in a formal description and reasoning rules construction, this paper uses the concept of behavior ontology to describe software behavior and user behavior and their relationship.

Definition 1: BO::= {Sub，Obj，Time，Env，Op}

Among this definition, the Sub represents the subject of a behavior, the Obj represents the object of a behavior, Time indicates the time of occurrence of a behavior, Env represents the environmental state while a behavior occurs, and Op represents the specific operation of a behavior.

Subject of behavior is divided into two categories: software subject and user subject. In a specific smartphone, it can be considered that there is only one unique value of the user subject. Because there are various system software and application software installed on the smartphone, values of software subject are diverse.

Object types of behavior are more abundant. Objects operated by software and users can all be regarded as an object, such as the telephone number of dialing, the web address of network accessing, the SMS message, the APP package downloaded, etc.

A consistent formal representation of subject and object is defined below.

Definition 2: Sub::=Obj::={Index, Attrs}

Index is the unique index value of a subject or object, and the value 0 is specified as the sole value of the user subject. Other index values of subjects and objects are automatically generated by the system. Attrs represent a number of specific attribute values contained in the subject or object. Table 2 lists attributes of key subjects and objects.

Table 2. Parts of attributes of example subject/object

Subject	Object	Attributes
user		None
Apps		name
	App package downloaded	App Name, package name, time, MD5 value
	Phone number of dialing out	number, time, talk time, whether in contact list, regional operator, security label
	Phone number of dialing in	number, time, ring duration, whether answered, whether in contact list, regional operator, security label
	SMS Received	number, whether in contact list, time, content, security label
	SMS sent out	number, whether in contact list, time, content, security label, whether MassSMS
	WIFI Connected	Name, whether password set, connection time, whether open and shared
	Website Accessed	Website address, time, security label
	…	…

The environment Env refers to the network environment connected by a smartphone when the behavior occurs. Generally speaking, a 3G/4G network connection and private WIFI network connection can be considered a security network environment. Under this environment, users can confidently access private information such as personal accounts and passwords, as well as use network payments. But, if the smartphone is connected to a public shared WIFI (whether or not it needs a password to connect), it is considered to have low security, and this is not suitable for accessing personal privacy information or network payment activities.

4. Behavior associated graph with time threads

The concept of behavioral ontology is defined to facilitate the description of behavior and the relationship between them. Furthermore, this paper defines a behavior association graph to describe their relationship. A behavior associated graph is a

kind of directed graph. Nodes in the graph represent behavior ontologies, and directed edges in the graph represent relationships among the behaviors.

In fact, behaviors are in the order of time. This order greatly affects the relationships among behaviors. For example, mobile users usually receive fraudulent SMS before they can link a fraudulent web site in a short message. Usually after receiving a phone call and being hung up on, the user then may redial back. Therefore, it is necessary to consider the sequence of time among behaviors when studying the relationship among them. Behavior associated graph with time threads defined in this paper can greatly simplify the establishment of relationships among behaviors and improve the efficiency of relational monitoring and other kinds of operations.

Definition 3: Time Threaded Behavior Associated Graph is a directed graph defined as THBAG::={V, E}.

V represents nodes in the graph, which is defined as V::={BO, TT}. BO is a behavior ontology that represents a behavioral activity, and TT (Time Thread) is the time thread pointer that points to the next node in the graph. Assuming that the occurrence of the behavior follows a strict time sequence, the time thread pointer then links them to form a one-way linear directed graph according to the time sequence of these behaviors. If there are 2 or more behaviors whose time attributes are completely the same and cannot be distinguished from the order, then the behavior ontologies with the same temporal attributes will be linked sequentially according to the order of processing without having to form multiple branch links. This not only simplifies the logic structure, but also improves the efficiency of the algorithm. It also does not affect the establishment of the relationship among behaviors.

E represents a directed edge of the graph, which is defined as E::={P, Ass}. P is a pointer to another node in the graph that represents a risk association between these two nodes. Ass is a description of the relationship. The relationship is described by the logical expression of the attribute decision of the behavior.

Figure 1 is an example of the logic diagram of a behavior associated graph with time threads including 7 nodes.

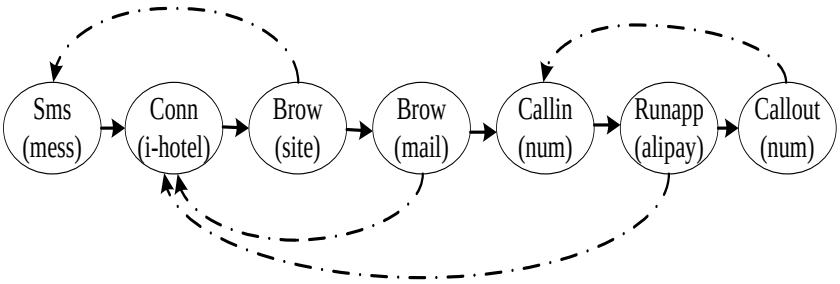

Figure 1. Example of logic diagram of a behavior associated graph with time threads

The pointer of the real line arrow in Figure 1 represents the time thread, and the directed edge of the dotted arrow indicates the risk association. The behavior scenes in Figure 1 include: (1) the user receives a SMS that contains an unsecured website URL link; (2) the user's smartphone is connected to an unsafe shared WIFI; (3) the user accesses the linked URL in the SMS and may encounters the pitfall risk of a fishing site; (4) the user accesses the enterprise mailbox through the unsafe WIFI network, and risks being monitored and having passwords stolen; (5) the phone rings twice and is not answered, which may be a call from malicious phone number; (6) the user is running an Alipay APP in the unsafe WIFI network environment, facing the risk of theft of funds; (7) the user finds that there is an unanswered call, dials back, then it and might plunge into a suction phone trap.

5. Dynamic association method for behavioral risk with time threads

An efficient dynamic association method for behavioral risk is proposed in this paper. As opposed to previous methods for static data set or behavior set analysis, behaviors in this method are dynamically inserted into a behavioral associated graph, taking behavior time as a thread. This method can only scan the behavior set once to complete the risk association analysis among the new behavior and the behaviors that have happened.

This method mainly includes two algorithms. The first algorithm is the risk associating algorithm, which is used to associate a behavior with other previous behaviors after a behavior occurs. With the continuous occurrence of behaviors, a dynamic growth graph is formed. Each established association presents a risk of behavior security. Accompanying the growth of the behavior graph is a scale of the graph that will be bigger and bigger. Some behavior nodes that have occurred for a long time do not have associated values, and so these nodes need to be deleted from the graph and the associations among them should be cut off. This is called the pruning algorithm.

5.1. Risk associating algorithm

Algorithm 1 Risk associating algorithm

Input: Behaviors
Output: *a behavior associated graph with time threads*

```
1.  aRule: Association rule set
2.  THBAG={V, E};
3.  V={ϕ};
4.   E={ϕ};
5.  While GetBO(bo)
6.  {
7.        v.BO=bo;
          v.TT=NULL;
8.        Add&Thread(v, V);
9.        For each v' ∈ V
10.       for each r ∈ aRule
11.          if match(v, v', r)   Addedge(v, V);
12. }
```

To improve the efficiency of the algorithm, the front and rear pointers of the time thread should be set to implement the algorithm. Each time the behavioral ontology nodes are inserted, they should be put directly into the rear of the thread. When the directed edge of the risk association relationship is established, the front pointer of the time thread should be adopted to scan through the behavioral ontology nodes that already exist. The flow-process diagram is shown in Figure 2.

5.2. Pruning algorithm

Algorithm 2 Pruning algorithm

Input: t: a specified time threshold value, any behavior ontology node before this threshold value will be cut off.
Input: THBAG: *a behavior associated graph with time threads to be pruned.*
Output: *a behavior associated graph with time threads*

```
1.  THBAG={V, E};
2.  For each v ∈ V
3.  {
4.        If Gettimeattr(v)<t
5.        {
6.            for each (v' ∈ V)  and  (v<>v')
7.                if existedge(v, v')  deleteedge(v, v');
8.            deletevertex(v);
9.        }
10. }
```

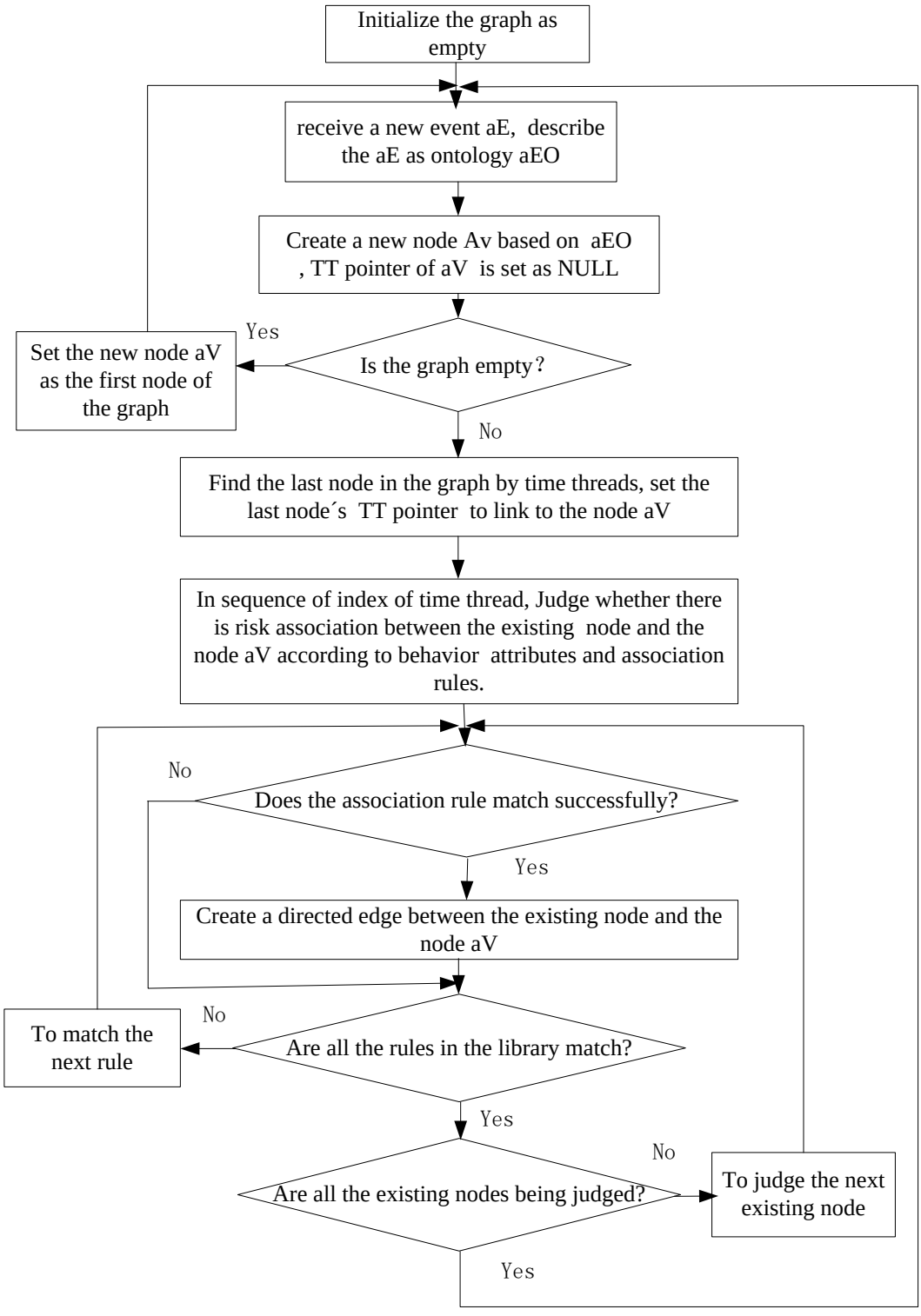

Figure 2. Flow-process diagram of risk associating algorithm

5.3. Association rules

Association rules are used to determine the risk association between two different behavior ontologies. The relationship between two behavioral ontologies mainly considers the object of behavior. Specifically, the object of the two behavior ontologies have security relevance, and the object attributes meet certain security risk conditions.

Take the risk association in Figure 1 as an example. There are 4 risk associations in Figure 1, which can be described as follows: (1) the user clicks the URL and surfs, which is included in the short message, and the URL is a risky website address; (2) the user accesses the enterprise mailbox through a public and open WIFI; (3) the user is running the Alipay APP for online payment through a public and open WIFI; (4) the user dials back an unanswered phone number, and the number is marked as a swindle.

The risk association rules are described formally using a logic relation decision formula. If all the logical decision condition values are "TRUE", the risk association can be judged to be true.

sRule1: Brow(site) $\wedge$ Sms (mess) $\wedge$ IsIncluded(site,mess) $\wedge$ Isrisk(site) $\rightarrow$ Risk association
sRule2: Brow(mail) $\wedge$ Isuserpass (mail) $\wedge$ Conn(wifi) $\wedge$ Isrisk(wifi) $\rightarrow$ Risk association
sRule3: Runapp(app) $\wedge$ Isdealmoney(app) $\wedge$ Conn(wifi) $\wedge$ Isrisk(wifi) $\rightarrow$ Risk association
sRule4: Callout(num1) $\wedge$ Callin(num2) $\wedge$ Isequal(num1, num2) $\wedge$ Isrisk(num2) $\rightarrow$ Risk association

The association rules in the library can be adjusted dynamically according to the actual application needs.

5.4. Algorithm analysis and experimental results

In the process of the risk associating algorithm with time threads, a behavior ontology node is put into the graph while the behavior occurs, and the risk association relationships between this node and others are constructed promptly. Therefore, the efficiency of the algorithm mainly considers the efficiency of inserting a certain behavior to the graph and constructing the risk associations. When a new behavior ontology node is put into the behavior associated graph, it is inserted directly into the tail of linked list according to time threads, so the efficiency of the algorithm depends mainly on the time of establishing risk associations. To construct risk associations between the new node and others, it needs to scan through the linked list in the graph according to time threads. While scanning, for each existing node in the graph, the association rules determine whether there is a risk association between them. Assuming that the number of behavior ontology nodes that exist in the graph is n, and the number of association rules in the association rule base is m, then the time complexity of the risk associating algorithm is O (nm), and the time complexity of the pruning algorithm is O (n). Thus, the time complexity of the risk associating algorithm is mainly related to the number of behavior ontology nodes and the number of association rules. The high efficiency of establishing risk associations can be improved by cutting off the unvalued behavior ontology nodes that exceed the specified time threshold.

The smartphone used in this article is the Coolpad 7620L-W00, and the system version is 4.4.2. Real-time risk association analysis is conducting on a PC machine after acquiring mobile data. The PC machine is configured as: Intel(R) Core(TM) i5-3570 CPU 3.40GHz processor, 8GB memory, Windows 7 Service Pack 1 operating system.

The following Table 3 and Table 4 are the experimental data for the 15-day monitoring and analysis of the test machine.

Table 3. 15 days experimental statistical analysis data

Number of monitoring behavior	586
Associated quantity	61
Association type	9
Association rate	10.41%
Association accuracy	100%
associated average response time on PC	0.78 秒

Table 4. Statistical analysis table of association risk of experimental data

Association type	Associated quantity	Risk description
Dial in->Dial out	5	Dial back a malicious phone number
Receive SMS -> Send SMS	11	Reply to malicious SMS
Receive SMS ->Visit website	8	link to a malicious URL in SMS
Receive SMS ->Download APP	4	Download the unknown APP provided by SMS
Open shared WIFI->Access mailbox	9	Disclosure of mailbox account information
Open shared WIFI->pay by Alipay	12	Alipay account information may be leaked
Collection of address book	5	Disclosure of personal contact privacy information
Collection of SMS	5	Disclosure of personal message privacy information
Collection of photos	2	Disclosure of personal photo privacy information

The experimental data shows that the accuracy of the method is very high, and the risk associated response speed of the PC terminal can meet the practical requirements. Though, the associating accuracy is closely related to the risk recognition rate of the association rules and the object attributes. With the complexity of the object property and an increase in the number of behaviors, the probability of the accuracy could be reduced. On the other hand, if the associating response is implemented directly on the mobile smart phone, the average response time of the risk association can be further improved.

6. Conclusions

In this paper, a method of dynamically associating behavior risks based on the time thread is proposed, which is mainly applied to the association analysis while behavior occurs. But, if there is a strict time sequence between behaviors, this method can also be used for post hoc association analysis. Presently, this method is mainly used for the association analysis of behavioral risk in mobile phones. In fact, this method can also be used in the association analysis of other types of events. Further work is needed to combine the information acquisition and analysis technology of mobile phones to develop an APP to monitor the behavior risk. More work is needed to extend the method and use the association analysis and risk detection in other fields. One of the ongoing works is to cooperate with the electric power supervision department to analyze the related event data provided by the 95598 power supply service system.

Acknowledgements

This work was partly financially supported through grants from the Liaoning Natural Science Foundation of China (Grant: 2015020020) and the State Grid Corporation Science and Technology Project (Contract number: 2017YF-34). The authors thank the 3 anonymous reviewers for their helpful suggestions.

References

1. M. K. Alzaylaee，S. Y. Yerima，S. Sezer. “EMULATOR vs REAL PHONE: Android Malware Detection Using Machine Learning”. ACM on International Workshop on Security and Privacy Analytics ACM, 2017:65-72.
2. I. Burguera，U. Zurutuza，S. Nadjm-Tehrani. “Crowdroid: Behavior-based Malware Detection System for Android”. ACM Workshop on Security and Privacy in Smartphones and Mobile Devices ACM, 2011:15-26.
3. D. Chen, H. M. Zhang, X. L. Zhang. “Detection of Android Malware Security on System Calls”. Advanced Information Management, Communicates, Electronic and Automation Control Conference IEEE, 2017:974-978.
4. P. Q. Chen. “Software Behavior Based Trustworthiness Attestation for Computing Platform”. Journal of Software 7.1(2012):55-60.
5. X. Q. Cheng, X. L. Jin, Y. Z. Wang. “Survey on Big Data System and Analytic Technology”. Journal of Software,2014,25(9):1889−1908.
6. M. Dhingra. “Legal Issues in Secure Implementation of Bring Your Own Device (BYOD)”. Procedia Computer Science, 2016, 78:179-184.
7. M. Donner. “Toward a Security Ontology”. IEEE Security & Privacy1.3(2003):6-7.
8. W. Dan. “Trusted Analysis Model for Interactive Behavior of a Software System Based on Slicing Technology”. Journal of Beijing University of Technology, 39.5(2013):713-721.
9. W. Enck，D. Octeau，P. Mcdaniel，S. Chaudhuri. “A Study of Android Application Security”. Usenix Conference on Security USENIX Association, 2011:21-21.
10. L. G. Lei, J. W. Jing, Y. W. Wang, Z. W. Zhang. “A Behavior-Based System Resources Access Control Scheme for Android”. Journal of Computer Research and Development, 2014,51(5):1028-1038.
11. Z. M. Lin. “Classifying Android Malware with Dynamic Behavior Dependency Graphs”. Trustcom/bigdatase/ispa IEEE, 2017:378-385.
12. J. Y. Liang, C. J. Feng, P. Song. “A Survey on Correlation Analysis of Big Data”. Chinese Journal of Computers, 2016(1):1-28.
13. M. Schur，A. Roth，A. Zeller. “Mining Behavior Models from Enterprise Web Applications”. Joint Meeting on Foundations of Software Engineering 2013:422-432.
14. Z. H. Tan. “A Novel Trust Model Based on SLA and Behavior Evaluation for Clouds”. Privacy, Security and Trust IEEE, 2017:581-587.
15. J. Tian，H. Jiao. “A Kind of Dynamic Software Behavior Trust Model Based on Improved Subjective Logic”. Intelligent Automation & Soft Computing (2016):1-9.
16. Q. Q. Ye. “Analyzing Security Property of Android Application Implementation Using Formal Method”. International Conference on Engineering of Complex Computer Systems IEEE, 2016:214-217.
17. L. Zhuang，M. Cai，L. Chen. “Software Behavior-Based Trusted Dynamic Measurement”. Journal of Wuhan University 56.2(2010):133-137.

Zhenliu Zhou received his Bachelor's Degree from Shenyang Aerospace University, Shenyang, China, in 1994, his Master's Degree from the China Academy of Space, Shenyang, China, in 2000, and his Ph.D. Degree from the University of Chinese Academy of Sciences, Beijing, China, in 2008. Now, he is a Professor at the Shenyang Institute of Engineering, and the Director of the Shenyang Key Laboratory of Information Security for Power System, Shenyang, China. His current research interests include network security, trusted computing and information processing.

Xiaoming Zhou received his Bachelor's Degree from Liaoning Technical University, Fuxin, China, in 2000, his Master's Degree from North-eastern University, Shenyang, China, in 2006, and his Ph.D. Degree from Shenyang Institute of Automation Chinese Academy of Sciences, Shenyang, China, in 2009. Now, he is a Senior Engineer at the State Grid Liaoning Electric Power Co., Ltd, and Section Chief of the Operating & Monitoring (control) Centre. His current research interests include power grid operation monitoring and information processing.

Weichun Ge received his Bachelor's Degree from the Northeast Electric Power University, Jilin, China, in 1984, his Master's Degree from Northeast Electric Power University, Jilin, China, in 1987, and his Ph.D. Degree from North China Electric Power University, Baoding, China, in 1992. Now, he is a Senior Engineer at the State Grid Liaoning Electric Power Co., Ltd, and the Minister of the Ministry of Science and Technology Communication.

Yueming Pan received his Bachelor's Degree from the Northeast Electric Power University, Jilin, China, in 1997, and his Master's Degree from North China Electric Power University, Baoding, China, in 2017. Now, he is a Senior Engineer at the State Grid Jinzhou Electric Power Co., Ltd, and the Director of the Operating & Monitoring (control) Centre. His current research interests include power grid operation monitoring and information processing.

Yu Gu received his Bachelor's Degree from Shenyang Institute of Engineering, Shenyang, China, in 2009, and he is a Master's student from the North China Electric Power University, Baoding, China. Now, he is an Engineer at the State Grid Jinzhou Electric Power Co., Ltd. His current research interests include power grid operation monitoring and data mining.