Program Anomaly Detection: Methodology and Practices

Xiaokui Shu
IBM Research
Yorktown Heights, NY, USA
xiaokui.shu@ibm.com

Danfeng Yao
Department of Computer Science
Virginia Tech
Blacksburg, VA, USA
danfeng@cs.vt.edu

ABSTRACT

This tutorial will present an overview of program anomaly detection, which analyzes normal program behaviors and discovers aberrant executions caused by attacks, misconfigurations, program bugs, and unusual usage patterns. It was first introduced as an analogy between intrusion detection for programs and the immune mechanism in biology. Advanced models have been developed in the last decade and comprehensive techniques have been adopted such as hidden Markov model and machine learning.

We will introduce the audience to the problem of program attacks and the anomaly detection approach against threats. We will give a general definition for program anomaly detection and derive model abstractions from the definition. The audience will be walked through the development of program anomaly detection methods from early-age n-gram approaches to complicated pushdown automata and probabilistic models. Some lab tools will be provided to help understand primitive detection models. This procedure will help the audience understand the objectives and challenges in designing program anomaly detection models. We will discuss the attacks that subvert anomaly detection mechanisms. The field map of program anomaly detection will be presented. We will also briefly discuss the applications of program anomaly detection in Internet of Things security. We expect the audience to get an idea of unsolved challenges in the field and develop a sense of future program anomaly detection directions after attending the tutorial.

Keywords

Anomaly detection; intrusion detection; program trace; program analysis; formal language; detection accuracy

1. INTRODUCTION

Program attacks are one of the oldest and fundamental threats to computing systems, which evolve and constitute latest attack vectors and advanced persistent threats.

CCS'16 October 24-28, 2016, Vienna, Austria

ACM ISBN 978-1-4503-4139-4/16/10.

DOI: http://dx.doi.org/10.1145/2976749.2976750

Anomaly-based intrusion detection discovers aberrant executions caused by attacks, misconfigurations, program bugs, and unusual usage patterns. The approach models normal program behaviors instead of the threats. It does not bear time lags between emerging attacks and deployed countermeasures as standard defenses do, which are built upon retrospects of inspected attacks. The merit of program anomaly detection is its independence from attack signatures. This property enables proactive defenses against new and unknown threats.

Program anomaly detection systems (a.k.a. host-based intrusion detection systems) follow Denning's intrusion detection vision [1]. Conventional systems were designed to detect illegal control flows or anomalous system calls based on two primitive paradigms: *i)* n-gram short call sequence validation that was introduced by Forrest et al. [2]; and *ii)* automaton transition verification, which was first described by Kosoresow and Hofmeyr [5] (DFA) and formalized by Sekar et al. [7] (FSA) and Wagner and Dean [10] (NDPDA). The two paradigms were advanced with machine learning models [6], hidden Markov models [11, 3, 13, 12], and neural network models.

The detection accuracy of program anomaly detection methods relies on the description precision of normal program behaviors and the completeness of the training [9]. Early-age program attacks, e.g., return addresses manipulation and library/system call injection, incur great variation from normal behaviors. Thus, they can be distinguished from relatively imprecise descriptions of normal program behaviors, e.g., n-gram system call anomaly detection [2] – regular grammar description of system call traces. However, modern program attacks utilize indirect means of control flow manipulation, e.g., data-oriented programming [4], or abuse programs within legal control flows, e.g., denial of service attacks (DoS). The emerging stealthy attacks diminish the effectiveness of conventional anomaly-based intrusion detection models and lead to the development of new models, e.g., long trace event correlation analysis [8], describing program behaviors through context-sensitive grammar.

This tutorial aims to give the audience an overview of program anomaly detection and inspire people to explore future directions and solve open issues. The tutorial will explain program anomaly detection from both practical and theoretical perspectives, presenting a field map for the audience to understand the evolution of the field as well as potential future directions.

We outline the sketch of the tutorial below and describe the subtopics in the following sections.

- Introduction to program attacks and primitive anomaly detection paradigms.

- Formal definition of program anomaly detection and the evolution of detection systems.

- A tale of two paths: program anomaly detection and control-flow enforcement.

- Unsolved issues and possible future directions.

2. PREREQUISITE KNOWLEDGE

System security researchers at all levels are welcome to the tutorial. We aim to *i)* introduce the problem of program anomaly detection to junior researchers/students, and *ii)* discuss the formalization of the problem, unsolved issues and possible future directions with senior researchers/students. A basic understanding of system security is required, e.g., call stack operations, buffer overflow and countermeasures, protection rings, control flows in programs. Related and advanced knowledge like automata theory, hidden Markov model, machine learning mechanisms, or correlation analysis, are not required, but could help develop a deeper understanding of some subtopics in the tutorial.

3. REFERENCES

[1] D. E. Denning. An intrusion-detection model. *IEEE Transactions on Software Engineering*, 13(2):222–232, 1987.

[2] S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff. A sense of self for Unix processes. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 120–128. IEEE Computer Society, 1996.

[3] D. Gao, M. K. Reiter, and D. Song. Behavioral distance measurement using hidden Markov models. In *Proceedings of the International Symposium on Research in Attacks, Intrusions and Defenses*, pages 19–40. Springer, 2006.

[4] H. Hu, S. Shinde, S. Adrian, Z. L. Chua, P. Saxena, and Z. Liang. Data-oriented programming: On the expressiveness of non-control data attacks. In *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2016.

[5] A. P. Kosoresow and S. A. Hofmeyr. Intrusion detection via system call traces. *IEEE software*, 14(5):35–42, 1997.

[6] W. Lee and S. J. Stolfo. Data mining approaches for intrusion detection. In *Proceedings of the USENIX Security Symposium*, pages 6–6. USENIX Association, 1998.

[7] R. Sekar, M. Bendre, D. Dhurjati, and P. Bollineni. A fast automaton-based method for detecting anomalous program behaviors. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 144–155. IEEE Computer Society, 2001.

[8] X. Shu, D. Yao, and N. Ramakrishnan. Unearthing stealthy program attacks buried in extremely long execution paths. In *Proceedings of the 2015 ACM Conference on Computer and Communications Security (CCS)*, pages 401–413. ACM, October 2015.

[9] X. Shu, D. Yao, and B. G. Ryder. A formal framework for program anomaly detection. In *Proceedings of the 18th International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, pages 270–292. Springer, November 2015.

[10] D. Wagner and R. Dean. Intrusion detection via static analysis. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 156–168. IEEE Computer Society, 2001.

[11] C. Warrender, S. Forrest, and B. Pearlmutter. Detecting intrusions using system calls: Alternative data models. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 133–145. IEEE Computer Society, 1999.

[12] K. Xu, K. Tian, D. Yao, and B. G. Ryder. A sharper sense of self: Probabilistic reasoning of program behaviors for anomaly detection with context sensitivity. In *Proceedings of the 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, May 2016.

[13] K. Xu, D. D. Yao, B. G. Ryder, and K. Tian. Probabilistic program modeling for high-precision anomaly classification. In *Proceedings of the IEEE 28th Computer Security Foundations Symposium*, pages 497–511. IEEE, July 2015.

Xiaokui Shu is a Research Staff Member in the Security Services Team (GSAL) at the IBM Thomas J. Watson Research Center. He received his Ph.D. degree in computer science at Virginia Tech. His research interests are in system and network security, such as intrusion detection, data leak detection, and mobile security. He graduated from Virginia Tech with Outstanding Ph.D. Student Award at the Department of Computer Science and graduated from University of Science and Technology of China (USTC) with Guo Moruo Award as an undergraduate. He succeeded at his first real-world penetration test at USTC and won the first prize in Virginia Tech Inaugural Cyber Security Summit Competition. Dr. Shu is an active member of the security research community serving as a shadow PC member and reviewer for top-tier security conferences and journals.

Danfeng (Daphne) Yao is an associate professor in the Department of Computer Science at Virginia Tech, Blacksburg. She is an Elizabeth and James E. Turner Jr. '56 Faculty Fellow and L-3 Faculty Fellow. She received her Computer Science Ph.D. degree from Brown University in 2007. She received the NSF CAREER Award in 2010 for her work on human-behavior driven malware detection, and most recently ARO Young Investigator Award for her semantic reasoning for mission-oriented security work in 2014. She received the Outstanding New Assistant Professor Award from Virginia Tech College of Engineering in 2012. Dr. Yao has several Best Paper Awards (ICICS '06, CollaborateCom '09, and ICNP '12). She was given the Award for Technological Innovation from Brown University in 2006. She held a U.S. patent for her anomaly detection technologies. Dr. Yao is an associate editor of *IEEE Transactions on Dependable and Secure Computing (TDSC)*. She serves as PC members in numerous computer security conferences, including ACM CCS.