An Effective Cloud-Based Active Defense System against Malicious Codes

Zhenyu Zhang, Wujun Zhang, Jianfeng Wang, Xiaofeng Chen

An effective cloud-based active defense system against malicious codes

Zhenyu Zhang[1], Junwu Zhang[1⋆], Jianfeng Wang[1], and Xiaofeng Chen[1]

State Key Laboratory of Integrated Service Networks (ISN),
Xidian University, Xi'an, P.R. China
yijiedao@sina.com, wjzhang@xidian.edu.cn, wjf01@163.com,
xfchen@xidian.edu.cn

Abstract. With the rapid development of cloud computing technique, network security has attracted more and more attention. Of all the network threats, malicious code is the major one. Due to the surge of number and species diversity of the malicious code, it is intractable for the existing antivirus techniques to defense all of the attacks. In this paper, we construct an effective cloud-based active defense system against malicious code. The constructed system utilizes the honey-pot subsystem to collect threaten data, and multiple behavior analysis engines work in parallel to generate a comprehensive program behavior analysis report. Furthermore, there are intelligent algorithms running on several computing servers to achieve automatic intelligent analysis on the reports. Associated with the multiple scan engines form a comprehensive, reinforced and more intelligent active defense system.

Key words: Cloud computing,Honey-pot,Behavior analysis,Active defense.

1 Introduction

With the development of Internet technology, computer viruses and hacker techniques combined with more variants and other development trends. Traditional AV(Anti-Virus) modes are not available to defense malicious codes, the reasons are as follows: Hysteresis and limitations of the traditional stand-alone defense; high cost of the sample collection and low effectiveness; longer upgrade cycle, more consumption of resources, poor user experience.

The emergence of cloud computing provide the ideas to resolve these problems, such as "Cloud AV" [1, 2] , "Cloud AV" moves the complex computing components from client to cloud computing server. However, some of the existing implements only combine the AV engines together, profit-driven make the detection with one-sidedness, moreover, some others deploy their server in the cloud, but still using signature-matching which cannot detect unknown viruses. Oberheide proposed a multiple antivirus engines work in parallel residing on cloud system called "CloudAV" [2], CloudAV does enhance the detection rate

⋆ Corresponding author

but lack of discussing the sample analysis and collection parts. Some other systems like "MIDeA" [14] and "GrAVity" [15] utilize the same idea that make the security services or tools work in parallel to get better efficiency and data processing rate. In [12] Peter and Robin et al. show that how diversity AV(Anti-Virus) engines may help improve detection gains. Xu et al. proposed an on cloud collaborative security services composing system called "CloudSEC" [13] which is a dynamic peer-to-peer overlay hierarchy with three architectural components. Cristian and Gustavo et al. proposed an ontology based malware detection system deploy in the cloud named "nCLAVS" [1] which is specifically on Web service applications.

Our contributions We propose an active defense system with enhanced malware detection ability and can do actively defense against unknown virus.

- We deploy a hierarchy honey-farm system on cloud platform to collect malicious attacks in large scale. The honey-farm system combines "Potemkin" [9] with TCP conversation migrate technology.
- We deploy multiple behavior analysis engines work in parallel in cloud. Multiple engines ensure the analysis to be comprehensive.

2 Preliminaries

Cloud computing Cloud Computing distributed computing tasks into resource pool which is consisted of a large number of computers. This resource pool called "Cloud" [3]. The core of cloud computing is parallel computing which separates task into several parts, each part can be allocated to a separate processor. Parallel computing system can be specifically designed and contain supercomputers with more than one processor or clusters based on a number of stand-alone computers interconnected, the parallel computing cluster can process data and return the results to the user after treatment.

Honey-pot Honey-pot is a kind of computer on the Internet and with no defense policy, its internal runs variety data recorders and special self-expose programs to tempt network intrusion behaviors. Honey-pot aims to collect threat data. There are two types of honey-pot system in practice:Low-interaction honey-pots that offer limited services to the attacker like Nepenthe [4] honeyd [5]. High-interaction honey-pots that offer the attacker a real system to interact with. Such as Gen-III honey-net [6].

Behavior analysis Behavior analysis engine like CWSandbox [8], is based on the "sandbox" [7] which makes the suspicious program running in a virtual "sandbox" in the full show, "sandbox" will record all its actions. Once the program fully exposed its viral properties,"sandbox" will erase the traces of the virus and restore the system to its original normal state.

3 Active Defense System Architecture

In the proposed system, we deploy the LSSC (large scale sample collection) subsystem and VIA (virtual isolate analysis) subsystem on the cloud platform to master diversity threat behaviors. The skeleton of the proposed defense system architecture as shown in Fig. 1.

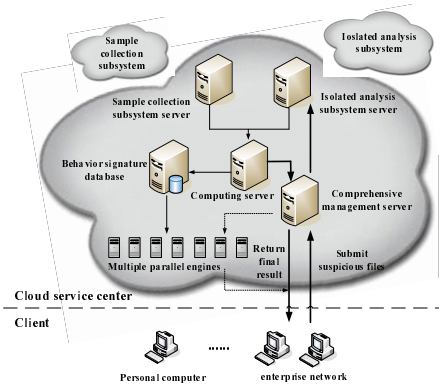

Fig. 1. Architecture of active defense system

3.1 Key System Components

Sample Collection Subsystem We deploy LSSC in a hierarchical way to make two types of honey-pot cooperate. Low interactive honey-pot would possibly unable to response some of the malicious requests, we use the TCP conversation migration to redirect data stream to the high interaction honey-pot to offer for higher level of capture.The LSSC structure as shown in Fig. 2.

Isolated Analysis Subsystem Since it's difficult for a single engine to capture comprehensive behaviors of a suspicious program, VIA integrates virtual monitors from different vendors work in parallel to perform a comprehensive monitoring.

Computing Server Computing server mainly process the data reported from the two subsystems using intelligent detect algorithms. Suspicious malicious codes will be given a weighted value, so behavior characteristics meet certain threshold will be stored in special format to form a behavior characteristic database.

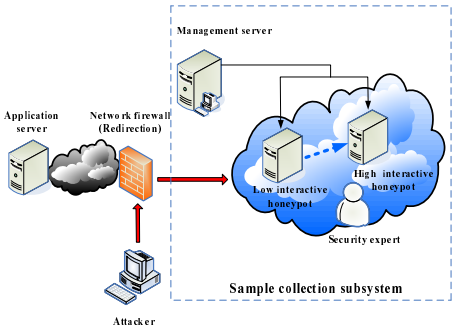

Fig. 2. Sample collection subsystem

Integrated Scheduling Management Server Integrated scheduling management server is responsible for components scheduling. Once received a request,the management server initiate multi-engine in a combination of optimized configuration to do a full scan of the file. If the file can not be judged, this server is responsible for submit it to VIA to do further analysis.

4 Performance Analysis

Multiple antivirus engines We deploy multiple AV engines work in parallel the same as proposed in [2]. We select several popular AV engines and collect 2718 old virus samples offered by "www.Bioon.com", and a new virus sample set of 283 offered by "bbs.kafan.cn" . Table 1 shows the detection rate of each engine when them work respectively on these samples.

Table 1. Detection rate of each engine

	Engine 1	Engine 2	Engine 3	Engine 4	Engine 5
2718 old samples	26.3%	49.42%	51.69%	51.06%	97.42%
283 new samples	10.03%	27.36%	43.45%	43.47%	56.53%

We also do the test of detecting the samples with different AV engines combinations. Make U denote the virus sample data set. If engine A can detect U_A samples out of U while engine B can detect $U_B(|U_A| \geq |U_B|)$, so the sum detect samples are

$$U_{SUM} = U_A \cup U_B$$

Even if $U_A \cap U_B = U_B$, the U_{SUM} is no less than either U_A or U_B. So we can conclude that multiple engines work concurrently will surely improve the detection rate. There is one thing that we should be concern: certain engine may did a false alarm. Its easy to realize that if we introduce more engines and the false alarm will be reduce accordingly.

Table 2. The main analysis items for each engine

	Threat Expert	CW Sandbox	Anubis	Joe sandbox	Cuckoo
File system modification	√	√	√	√	√
Registry modification	√	√	√	√	√
Memory modification	√	√	√	√	√
Process creation	√	√	√	√	√
DLL injection		√	√	√	√
API call	√	√		√	√
IP involved	√			√	
Network traffic	√	√		√	√
Source trace	√			√	

Multiple behavior analysis engines The proposed VIA subsystem is consist of several analysis engines. Table 2 demonstrate their report details of each engine. The empty cells do not indicate that tools do not have the coordinate functions but show the inadequacy of tools. multiple engines work in parallel will surely enhance the analysis efficiency and the analysis report will be more comprehensive.

Hierarchical honey-pot system We deploy the honey-pot system in cloud center in a hierarchy construction by traffic redirection and migrate technology, make the low and high interactive honey-pot cooperate. So the LSSC we proposed in this paper will sure be more efficient and comprehensive. Artail [10] and Bailey [11] implemented such a hybrid honey-pot system and proved its' efficiency.

5 Conclusion

The proposed system overcame the high cost and low efficiency of deploying honey-pot system under traditional antivirus mechanism and can collect network threats in large scale. On the other side, multiple behavior analysis engines work concurrently, avoid one-sidedness caused by single engine, ensure the reliability and practicability of analysis results from various aspects. The proposed system can deal with severe situation on network security.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (Nos. 61272455 and 61100224), Doctoral Fund of Ministry of Education of China, Program for New Century Excellent Talents in University, and China 111 Project(No. B08038).

References

1. C. A. Martnez and G.I.Echeverri and A.G.C.Sanz. Malware detection based on cloud computing integrating intrusion ontology representation. In *IEEE Latin-American Conference on Communications (LATINCOM)*, Bogota, pages 1−6, 2010.
2. J. Oberheide and E. Cooke and F. Jahanian. CloudAV:N-version antivirus in the network cloud. In *Proc. of the 17th USENIX Security Symposium,San Jose,Calofornia,USA*, pages 91−106, 2008.
3. M. Armbrust and A. Fox and R. Griffith. A view of cloud computing. In *Communications of the ACM*, Vol.53, No.4, pages 50−58, 2010.
4. P. Baecher and M. Koetter and T. Holz. The nepenthes platform: An efficient approach to collect malware. In *Recent Advances in Intrusion Detection, 9th International Symposium, Hamburg, Germany*,RAID, pages 165−184, 2006.
5. P. Niels. A Virtual Honeypot Framework. In *Proceedings of 13th USENIX Security Symposium, San Diego, CA, USA*, pages 1−14, 2004.
6. E. Balas and C. Viecco. Towards a third generation data capture architecture for honeynets. In Proceedings from the Sixth Annual IEEE SMC,Information Assurance Workshop, 2005,NY,USA, IEEE, pages 21−28, 2005.
7. W. Wright and D. Schroh and P. Proulx. The sandbox for analysis: concepts and Eevaluation. In *Proceedings of the 2006 Conference on Human Factors in Computing Systems,CHI 2006,Quebec,Canada*,pages 801−810, 2006.
8. C. Willems and T. Holz and F. Freiling. Toward Automated Dynamic Malware Analysis Using CWSandbox. In *Security & Privacy*, Vol.5, No.2, pages 32−39, 2007.
9. M. Vrable and J. Ma and J. Chen. Scalability, fidelity, and containment in the potemkin virtual honeyfarm. In *Proceedings of the 20th ACM Symposium on Operating Systems Principles 2005, Brighton, UK*, SOSP, pages 148−162, 2005.
10. H. Artail and H. Safa and M. Sraj. A hybrid honeypot framework for improving intrusion detection systems in protecting organizational networks. In *Computers & Security*, Vol.25, No.4 pages 274−288, 2006.
11. M. Bailey and E. Cooke and d. Watson. A hybrid honeypot architecture for scalable network monitoring. In *Technical Report CSE-TR-499-04, U. Michigan*, 2004.
12. G. B. Peter and E. B. Robin and G. Ilir and S. Vladimir. Diversity for Security: A Study with Off-the-Shelf AntiVirus Engines. In *IEEE 22nd International Symposium on Software Reliability Engineering, Hiroshima, Japan*, ISSRE 2011, pages 11−19, 2011.
13. J. Xu and J. Yan and L. He and P. Su and D. Feng. CloudSEC: A Cloud Architecture for Composing Collaborative Security Services. In *Cloud Computing, Second International Conference, CloudCom 2010, November 30 - December 3, 2010, Indianapolis, Indiana,USA, Proceedings*, pages 703−711, 2010.
14. V. Giorgos and P. Michalis and I. Sotiris . MIDeA: a multi-parallel intrusion detection architecture. In *Proceedings of the 18th ACM Conference on Computer and Communication Security, Chicago, Illinois, USA*, CCS, pages 297−308, 2011.
15. V. Giorgos and I. Sotiris . GrAVity: A Massively Parallel Antivirus Engine. In *Recent Advances in Intrusion Detection, 13th International Symposium, Ottawa, Ontario, Canada. Proceedings*, RAID, pages 79−96, 2010.