A Framework of APT Detection Based on Dynamic Analysis

Yunfei Su [a], Mengjun Li[b] , Chaojing Tang and Rongjun Shen

School of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073,China

[a]suyunfei@nudt.edu.cn, [b]resolph@gmail.com

Keywords: Advanced persistent threat, dynamic analysis, APT detection.

Abstract. Advanced persistent threat (APT) is sophisticated cyber-attack and has attracted lots of attention in cyberspace. Traditional defense measures based on signature matching are insufficient to detect APT, such as Stuxnet, Operation Aurora, Duqu, Flame, Red October, Miniduke and so on. In this paper, we proposed a framework of APT detection which includes network traffic redirection module, user agent, reconstruction module, dynamic analysis module and decision module. The framework could effectively detect APT attacks compared with current defense systems. We provide a detailed example to illustrate how the framework detects APT attacks especially passive attacks.

Introduction

At present, cyber-attacks which lead to too much cost are more sophisticated and common in cyberspace. Original network attacks in cyberspace have evolved into APT attacks, which are more complicated, stealthy and have profound effects.

Advanced persistent threat has attracted lots of attention in recent years. The term of Advanced persistent threat may originate from two main sources [1].One is the security company Mandiant [2], which provided a detailed report about APT. The other is the U.S. Department of Defense (DoD). The term APT was reportedly as early as 2006. U.S. National Institute of Standards and Technology (NIST), defines APT in 2011 as" An adversary that possesses sophisticated levels of expertise and significant resources which allow it to create opportunities to achieve its objectives by using multiple attack vectors (e.g., cyber, physical, and deception). These objectives typically include establishing and extending footholds within the information technology infrastructure of the targeted organizations for purposes of exfiltrating information, undermining or impeding critical aspects of a mission, program, or organization; or positioning itself to carry out these objectives in the future. The advanced persistent threat: (i) pursues its objectives repeatedly over an extended period of time; (ii) adapts to defenders' efforts to resist it; and (iii) is determined to maintain the level of interaction needed to execute its objectives".[3]

Stuxnet, Operation Aurora, Duqu, Flame, Red October, Miniduke are typical examples of APT [4,5]. According to these examples, we could get the common characteristics: a) the objective is to sabotage key infrastructures or exfiltrate information including intellectual property, political activities schedule and so on, b) the stage is multistep and kept for a long time, and sometimes it may last for several years, c) APT attacks are sophisticated, more than one zero-day vulnerabilities are usually used, d) the targets of APT are not widespread but specific.

APT attack is difficult to detect with current commercial security products, such as Anti-virus products, intrusion detection system and so on. Present network security system generally includes firewall, network intrusion detection system and virus scanners, but APT attacks can easily evade such defense mechanisms without being award like the former APT.

Our contribution in this paper is proposing a general network security framework with the aim of detecting APT attacks. The main idea of the framework is to apply network dataflow to application dataflow reconstruction results in dynamically detecting the APTs without affecting the normal working.

APT Detection

The main countermeasures against Cyber-attacks are generally deploying anti-virus products/host intrusion detection system (HIDS) and network intrusion detection system (NIDS). All of them use a mechanism of blacklist, which keeps a set of specific data segments or rules for judging. For example, when the network traffic contains a specific string in the blacklist, the NIDS will issue an alert, similar to the others. Obviously this mechanism also called signature detection has a vital weakness: it needs the pre-defined signature extracting from the known cyber-attacks, then signature matching effects. So anomaly detection had been proposed in NIDS, which can be used to detect unknown attacks. But this method has a natural weakness, i.e. a relative high false positive which seriously degrades the availability.

APT attack generally exploits zero-day vulnerability to gain the access privilege of the targets, such as Stuxnet [4] exploited four zero-day vulnerabilities in windows operating system including Windows print spooler (MS10-061), LNK format (MS10-046), Win32k Keyboard Layout (MS10-073), and task scheduler (MS10-092). These exploits are nearly impossible to be detected by detector based on signature matching.

Original cyber-attacks usually use positive attacks to gain full access privilege by directly sending several crafted packets to the target without interacting with the user. In APT attacks, more passive attacks emerge, which need to interact with the users such as open a specific URL, download the email attachment and open it or insert the USB stick to the computer and so on. Spear phishing being used widely in APT attacks is one type of the passive attacks. Attackers exploit the crafted attachment (doc, xls, pdf, etc.) of normal Email sent by a personate companion or other persons who the victim trusts, once the victim opens the crafted attachment, arbitrary code could be run in the victim's computer through the embedded shellcode in the attachment. This kind of attacks is even more difficult to detect by the traditional detection measures using at present.

Proposed Framework

The proposed framework is shown in Figure 1 which includes five main components: network traffic redirection module, user agent, reconstruction module, dynamic analysis module and decision module.

Network traffic redirection module.

This module copies the actual network traffic and redirects it to the Reconstruction module. This can be easily done by a supported switch or a server with two network cards.

User agent.

The main functionality of user agent is to providing auxiliary information in the host to the Reconstruction and Decision modules. The auxiliary information includes some basic operating information such as whether the mouse is moving, the process name of the most top window and so on which are useful to the Reconstruction and Decision modules. The user agent is similar to lightweight host monitor software.

Reconstruction module.

Using the reconstruction module we can reconstruct the application flow according to the network flow together with reducing the dependency to the host. The purpose is to restore the data which may contain malicious contents and then send the data to the dynamic analysis module. The form of the data may be a segment of HTML text, document (doc, xls, pdf, etc.) or executable file, it also can be several packets sent to the application installed in the host.

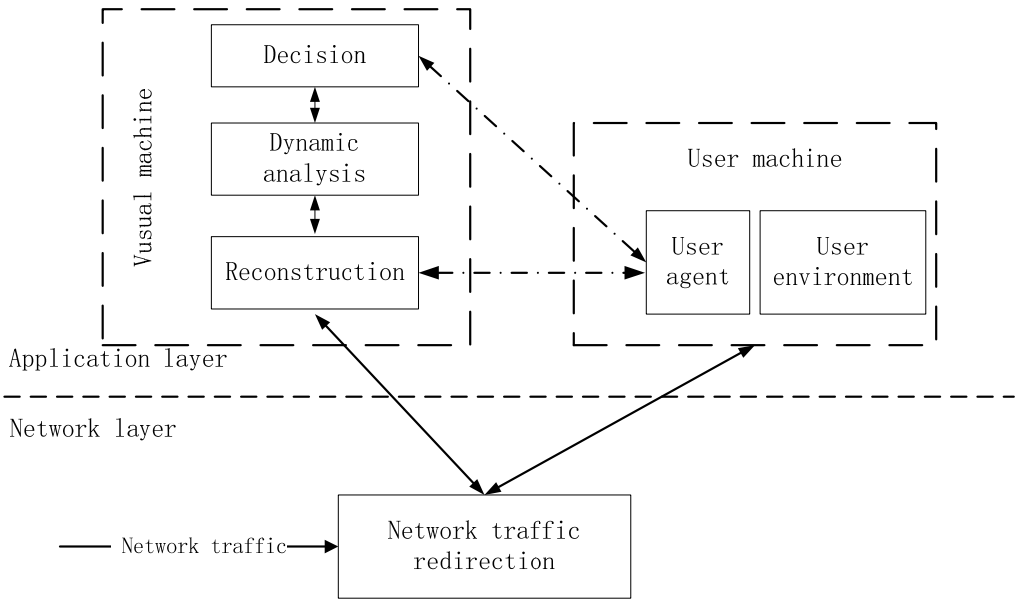

Fig. 1 Framework of APT detection

Dynamic analysis module.

With the auxiliary information provided by User agent module and the application data provided by the Reconstruction module, we can get a virtual environment whose context is similar to the any host in the inner-network. The main difference compared with the actual host is that visual environment is heavily armed by dynamic analysis systems which can effectively find out the malicious behaviors in the application data referred to the applications installed in the actual host.

Unknown attacks can be effectively revealed by Dynamic analysis due to malicious behaviors can't be hidden unless they don't do anything. So the exploitation with zero-day in APT can be always detected theoretically. For example, after the .doc document opened by WinWord.exe which is one of the components of Microsoft Office, if a request to a remote address is launched, we should log these suspicious behaviors and send them to the decision module. A second example, the exploitation of vulnerabilities has obvious behavior features, taking the exploitation of buffer overflow as an instance, to get arbitrary code running the input data should cover the memory address which could be used by EIP, these addresses called sensitive-address includes Return Address, Function pointer, VT pointer, etc. So we can monitor these addresses in runtime, once sensitive-address covering happens we should issue an alert. Such zero-day vulnerability detection systems are common and well-developed, Bitblaze[6] is one representative of them.

Decision module.

This module integrates former information and gets a conclusion according to pre-defined criteria.

APT attack can be divided into three main stages: Reconnaissance, Exploitation and Command & Control (C&C). In stage 1 reconnaissance, information gathering for guiding the next attack is the primary job. Exploitation and C&C are the main stages in APT attack which lead to real destructions.

The proposed framework focuses on the latter two stages, i.e. stage 2 exploiting phase, stage 3 interacting with C&C server. The more detailed content will be presented in next section.

A Case of The Framework in Detecting APT

Figure 2 shows a typical deployment of the framework in actual network. In the network, firewall, NIPS, NIDS and honeypot together with our Reconstruction and Dynamic Analysis Server (RDAS) are deployed against network attacks especially APT attacks.

We assume a simplified APT attack scenario: attacker's objective is to get some files storing in the working computers locating in the organization's inner-network. The inner-network is protected by firewalls, NIPS, NIDS, honeypot and anti-virus software in every computer in the organization.

According to three stages we divided, the attacker should gather enough information for planning and directing his attack. After a few days, he may find out: the organization may have two

subnetworks, one is called demilitarized zone (DMZ) which provides www, email and other business services, and the other may be inner-network protected by another firewall which forbids any access initiated from the outside; there are several email addresses belong to the employees of the organization; the web server in DMZ has several critical vulnerabilities that can be exploited to control the web server. With information above, obviously positive attack cannot directly achieve the goal wanted, attack make two passive attack plans: Plan A, exploiting the web server in DMZ and tampering with the html files in the server, once the employees access the web server the browser will run arbitrary code such as downloading a Trojan into their computers and executing it (exploiting zero-day to browser); Plan B, sending an email personating one's companion which has a crafted attachment, for example, a .doc document, once the employee open the .doc document the Winword process will run arbitrary code such as releasing a Trojan and executing it (exploiting zero day to Winword).

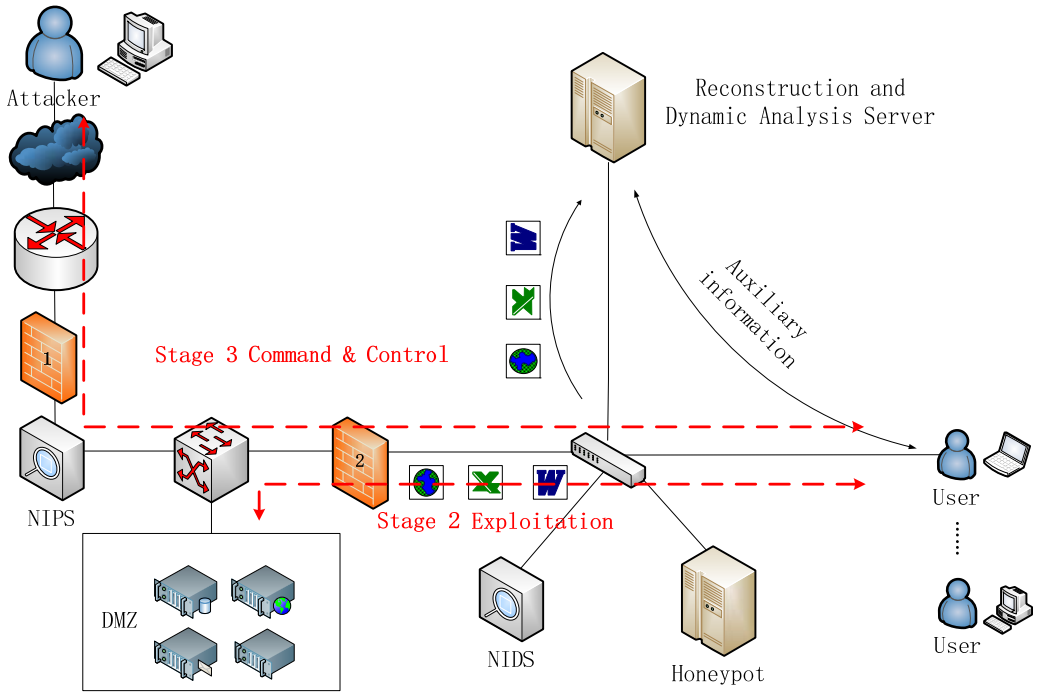

Fig. 2 detailed deployment of the framework in actual network

At stage 2, the attacker will start his plan. Choosing Plan A, the attacker may exploit SQL injection vulnerability to gain the administrator privilege, and then tampers with the index page which may be inserted with elaborate html code segment. Due to the web server is frequently accessed by the employees of the organization, the attack will easily succeed. Choosing Plan B, the attacker need send an email to the specific employee, including malicious content. Through society engineering, the employee may open the document attached in the email, and then malware embedded in the document executes. The attack procedure in stage 2 can bypass all defense measures deployed: 1.the network packet to the web server is permitted by the first firewall, 2.the SQL injection can bypass the NIPS through code obfuscation, 3.the request to the url is initiated by employees, so it has nothing to do with honeypot and the second firewall, otherwise, exploitation in Plan A is unknown attack with normal behaviors which NIDS is not able to detect, 4.the exploitation in Plan B is also unknown attack which NIDS and anti-virus products are not good at,5 the malware downloaded from the Internet or released from documents can be obfuscated skillfully to bypass any anti-virus installed in employees' hosts.

At stage 3, the malware such as RAT has already run in the host, the RAT may connect to the C&C to get commands or upload sensitive information through HTTP tunnel which can pass through two firewalls and evade the detection of NIDS/NIPS, honeypot again has nothing to do in this stage.

From the above, we can conclude that traditional defense measures do a little effort against APT attacks, that's why Stuxnet, etc. still can succeed in heavily armed network environment ignoring all of defense measures. With the proposed framework in this pater, we can detect these attacks. As last

section say, we focus on the latter two stages. The main functionality is integrated into the Reconstruction and Dynamic Analysis Server (RDAS).RDAS is a super server which keeps the mirror of all active hosts in the inner-network.

Stage 2 detection

In stage 2, when web server been accessed, the malicious html page is delivered to RDAS except for been delivered to the employee host. With the help of user agent module in the employee host, RDAS can simulate the context of employee host. If the malicious html page is explained by Internet explorer 11, the corresponding mirror in RDAS will use the same browser to explain the malicious html page. Similarly, after malicious .doc document in the email was delivered, if the .doc document is opened by Microsoft word 2013, the same as the mirror do. The translation from network traffic to application dataflow is done by the reconstruction module in RDAS, then dynamic analysis module will work. As said in section 3 dynamic analysis can identify the malicious behaviors hidden in the network traffic. After decision, RDAS may issue an alert or ignore it.

Stage 3 detection

In stage 3, RDAS can inspect the network traffic to identify suspicious network connections. Normal network connections generally initiate by user of host, if there are lots of network traffic between a host and an outside address, RDAS will check if there are user activities in the host together with other connection information, the user agent will return the relational information, if the net connection is hidden in user mode or the process which the connection affiliates is hidden, the connection is much suspicious. For example, the attacker installs RAT in the host after exploitation, the communication between attacker and the host masquerades http request and response whose contents are encrypted, so that the network connection can pass through the firewalls and bypass the NIDS/NIPS, but when RDAS receives these network packets, it finds that (through user agent module) the http network packets don't come from the browser processes, or they come from browser process which can't be seen on the desktop, then we can get a conclusion that the host is very likely under control by the attacker.

Related Works

Most research focus on malware analysis, including static analysis [7-9], dynamic analysis [10-14] and hybrid analysis [15,16]. In the paper[16], the authors extend the malware target recognition architecture initially proposed in [15] to an operational model for organization self-discovery of malware with low effective scan times and low false positive rates through successive data reduction and analysis.

In [17,18], the authors proposed an analytical security model considering the security analytics using Big Data. Their architecture is directed towards dealing with operational concerns in security organizations that aim to use existing security tools with Big Data analytics. Since their work is aimed towards operational side of security analytics therefore, it does not demonstrate any methodology of practical analysis of security threats as compared to our framework.

Ussath, Martin, Feng Cheng, and Christoph Meinel [19] proposed a Security Investigation Framework (SIF) that needs to process all investigation relevant information from different sources like prefiltered log files or forensic reports with novel correlation algorithms and rules. But this approach is prone to lead false alarm without semi-manual creation and the utilization of multiple information sources which could be a hard work.

Conclusion And Future Work

Stuxnet, Operation Aurora, Duqu, Flame, Red October, Miniduke have a much greater impact on network defense systems due to their sophisticated exploitation and their ability to evade detection. Traditional defense measures could not effectively detect the unknown attacks and passive attacks which are common in APT attacks.

The proposed framework in this paper is much effective in detecting APT compared with traditional defense measures, especially in detecting passive attacks. There are five main components

in the framework: network traffic redirection module, user agent, reconstruction module, dynamic analysis module and decision module. In the framework, dynamic analysis is the key module to detect known/unknown attacks. Dynamic analysis has a natural advantage in detecting malicious behaviors which can reveal the real intension in the files.

In the example illustrated in Figure 2, we provide a typical APT attack process together with corresponding detecting process, i.e. how RDAS works. Through the example, we demonstrate the efficiency of the framework proposed for detecting APT attacks.

References

[1] Andress J., "Advanced Persistent Threat, Attacker Sophistication Continues to Grow?" ISSA Journal,2011

[2] Mandiant, M Trends, the Advanced Persistent Threat. 2010

[3] National Institute of Standards and Technology, Managing Information Security Risk: Organization, Mission, and Information System View. Gaithersburg, National Institute of Standards and Technology, http://fismapedia.org/index.php?title=NIST_SP_800-39_Appendix_B

[4] Nikos Virvilis, Dimitris Gritzalis, "The Big Four-What We Did Wrong in Advanced Persistent Threat Detection?" 2013 Eighth International Conference on Availability, Reliability and Security, Sept. 2013

[5] Frankie Li, Anthony Lai, Ddl Ddl , "Evidence of Advanced Persistent Threat: A Case Study of Malware for Political Espionage," 2011 6th International Conference on Malicious and Unwanted Software, Oct. 2011

[6] Dawn Song, David Brumley, Heng Yin, Juan Caballero, Ivan Jager, Min Gyung Kang, Zhenkai Liang, James Newsome, Pongsin Poosankam, Prateek Saxena, "BitBlaze: A New Approach to Computer Security via Binary Analysis," ICISS 2008, December 2008

[7] T. Abou-Assaleh, N. Cercone, V. Keselj, and R. Sweidan, "N-gram based detection of new malicious code," in Proc. 28th Ann. Int. Comput.Softw. Appl. Conf., Sep. 2004

[8] O. Henchiri and N. Japkowicz, "A feature selection and evaluation scheme for computer virus detection," in Proc. IEEE 6th Int. Conf. Data Mining, Dec. 2006

[9] J. Kolter and M. Maloof, "Learning to detect and classify malicious executables in the wild," J. Mach. Learning Res. ,Dec. 2006.

[10] M. Bailey, J. Oberheide, J. Andersen, and Z. Mao, "Automated classification and analysis of Internet malware," in Proc. 10th Int. Symp. Recent Adv. Intrusion Detection, 2007

[11] M. Christodorescu, S. Jha, and C. Kruegel, "Mining specifications of malicious behavior," in Proc. 6th Joint Meeting Eur. Softw. Eng. Conf. ACM SIGSOFT Symp. Found. Softw. Eng., Sep. 2007

[12] A. Dinaburg, P. Royal, M. Sharif, and W. Lee, "Ether: Malware analysis via hardware virtualization extensions," in Proc. 15th ACM Conf. Comput. Commun. Security, Apr. 2008

[13] T. Lee and J. J. Mody, "Behavioral classification," in Proc. EICAR, Apr.2006

[14] A. Moser, C. Kruegel, and E. Kirda, "Limits of static analysis for malware detection," in Proc. ACSAC, 2007

[15] T. Dube, R. Raines, G. Peterson, K. Bauer, M. Grimaila, and S. Rogers,"Malware target recognition via static heuristics," J. Comput. Security, vol. 31, 2011.

[16] Thomas E. Dube, Richard A. Raines, Michael R. Grimaila, "Malware Target Recognition of Unknown Threats," IEEE SYSTEMS JOURNAL, SEPTEMBER 2013

[17] J. Howes, J. Solderitsch, I. Chen & J. Craighead, "Enabling trustworthy spaces via orchestrated analytical security", ACM, CSIIRW 2013

[18] Ahn, Sung-Hwan, Nam-Uk Kim, and Tai-Myoung Chung. "Big data analysis system concept for detecting unknown attacks." Advanced Communication Technology (ICACT), 2014 16th International Conference on. IEEE, 2014.

[19] Ussath Martin, Feng Cheng, and Christoph Meinel. "Concept for a security investigation framework." New Technologies, Mobility and Security (NTMS), 2015 7th International Conference on. IEEE, 2015.