www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 3 Mar, 2014 Page No. 5123-5127

Enhanced Worms Detection By NetFlow

Manish Khule[1], Megha Singh[2], Deepak Kulhare[3]

[1]Student,C.S.E,Ciit,Indore,
mannykhule@gmail.com
[2]Assistant Professor, C.S.E, Ciit
[3]Assistant Professor, C.S.E, Ciit, Indore,

Abstract: ***Enterprise networks are facing ever-increasing security threats from worms, port scans, DDoS, and network misuse, and thus effective monitoring approaches to quickly detect these activities are greatly needed. Firewall and intrusion detection systems (IDS) are the most common ways to detect these activities, but additional technology such as NetFlow can be a valuable enhancement. A worm (malicious codes) can disturb network and normal network operation. Internet worms are causes significant worldwide disruption, a huge number of infected hosts generate traffic, which will impact the performance of the internet. Therefore this is one of the areas where researchers are concentrating to find effective detection system, which will presence the worms and reduce the worm's spread. This paper deals with a classified study of most important and commonly used methods for detecting internet worms using Netflow.***

Keywords: security, network intrusion detection, Netflow , Internet worms, anomaly detection.

1. Introduction

The Internet is persistently threatened by many types of attacks such as viruses, and worms. A worm is a self propagating program that infects other hosts based on a known vulnerability in network hosts. In contrast, a virus is a piece of code attached to another executable program, which requires human action to propagate. A major challenge in networking is how to detect new worms and viruses in the early stages of propagation in a computationally efficient manner.[1]During the past 20 years, thousands of different worms have been developed. Some of these worms have caused huge disruption to global networks. The most notable worms include Morris, Code Red and Code Red II, Nimda, Slapper, and Sapphire/Slammer worms, and recently, So-Big.F, MSBlast, and Mydoom. From the first worm that was released in 1988 (the Morris worm), the area of Internet worm detection has been a significant research problem. In order to understand the worm threat, it is necessary to understand the various types of worms, payloads, and attackers. Taxonomy of the various possible worms, payloads, and attackers as an initial guide to plausible defenses.This taxonomy is necessarily incomplete, simply because new tactics, payloads, and attackers may arise. This taxonomy is based on several factors: target discovery, carrier, activation, payloads, and attackers. Target discovery represents the mechanism by which a worm discovers new targets to infect. The carrier is the mechanism the worm uses to transmit itself onto the target [5-9]. Activation is the mechanism by which the worm's code begins operating on the target. Payloads are the various non-propagating routines a worm may use to accomplish the author's goal. Finally, the various possible attackers have different motives and would there- fore utilize different payloads.In addition, it is important to note that worms needn't be confined to a single type within each category. Some of the most successful worms are multi-modal, employing multiple means of target discovery, carrier, payload, etc, where the combination enables the worm to surpass defenses (no matter how effective) that address only a single type of worm. In this section, summary of previous approaches to worm detection has been done [6-15]. Usually, the detection methods are based on the feature of the Internetworm such as abnormal network traffic, content comparison, process scanning and detecting network connection.The current detection method for the Internet Worm two general categories: Signature-based Detection and Anomaly Detection. Signature-based detection is based on defining malicious patterns that the system has to detect. Signature-based detection suffers from the problemthat it requires a signature of each attack be known. In contrast, anomaly detection differs by constructing a profile of normal behaviors or activities on the network, and then looking for activities that do not fit the normal profile. Since not all the abnormal activities in the network are suspicious, anomaly detection has the problem of raising false alarms when it encounters normal traffic.The Internet worms diffuse quickly to infect servers, destroy information, embed backdoor , and consumer resource from network bandwidth In the trap oriented detection method, the surveillance ar ea can be separated into single host and the several network segments on the Internet. In this method, the accuracy is quite high and it is easy to differentiate between the normal and abnormal traffic. Therefore, the nodes have to collect the network flows (information which is produced from router), for finding abnormal traffic.[12].

2. GENERAL WORM MECHANISMS

A worm should be able to identify already infected tar gets and refrain from reinfecting them. Interestingly, the first three requirements are already enough. The fourth merely

imprvoes efficiency. Also, if a service on the target system is capable and willing to propagate the worm without having its security compromised, then a worm can do without any kind of system compromise at all. A compromise of the target system to some degree is customary nonetheless, especially when some other purpose, like espionage, sending of spam or attacks on other systems is intended.[12-15] A second r ea- son for target system compromise is that many worms use security vulnerabilities to obtain resources on the tar get system. The advantage is that in this way the basic execution services of the target system become available to the worm and any functionality its designer wants can be easily implemented. The typical worm uses a propagation mechanism that works like this:

However , inorder for a worm to propagate as fast as possible, it is a sound design choice to not impair the functionality of an infected host until the worm has completed most or all of its intended propagation activity from that host. In addition, the damage may be done later to delay the discovery of the worm or in order to allow coordinated attacks from several infection generations

3. METHODS FOR DETECTING INTERNET WORMS USING NETFLOW

3.1 NetFlow Overview

NetFlow is a traffic profile monitoring technology developed by Darren Kerr and Barry Bruins at Cisco Systems, back in 1996. As a de facto industry standard, NetFlow describes the method for a router to export statistics about the routed socket pairs, and it's now a built-in feature for most Cisco routers as well as Juniper, Extreme and some other vendor's routers and switches.

When a network administrator enables the NetFlow export on a router interface, traffic statistics of packets received on that interface will be counted as "flow" and stored into a dynamic flow cache.

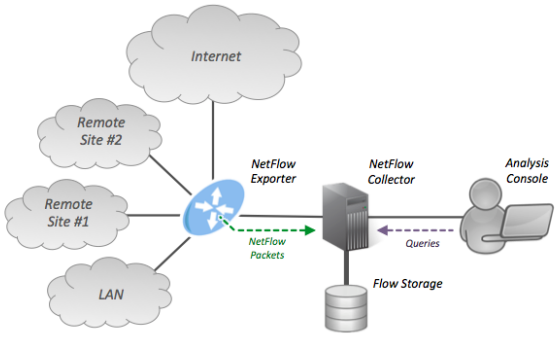

Figure1: NetFlow Architecture

3.2 Flow-based analysis methods

1-Top N and Baseline

2- Top N session

3- Top N data

4- Pattern Matching

5- Port matching

6- IP address matching

7- Match a special IP or IP list

we looked at what NetFlow is and how it can be used in the early detection of worms, spammers, and other abnormal network activity for large enterprise networks and Internet service providers. The paper discussed some of the most common methods of flow-based analysis: Top N, Baseline and Pattern Matching techniques.

3.3 NetFlow Implementation

The difficult task when performing flow-based analysis is that the administrator must evaluate a very large number of flow records. If he is just relying on the Top N, baseline and pattern matching methods, the administrator will merely get a coarse view of network abnormities. We've seen many times there are moderately intensive worms and other abnormal activities which appear intangible amongst the immense amount of legitimate traffic that is typically found in a large enterprise network. Those malicious hosts will not show up in the Top N lists, nor will we know in advance what key fields and values to 'grep' -- yet these are still malicious hosts that must be addressed.

In order to identify the abnormities more effectively and accurately, a better way to narrow the analyzable flow records is required. Fortunately, for most types of TCP-based worms and other abnormities, there is another useful field in flow records: analysis based on the TCP flags.

Worms, by their replicating nature, are programmed to seek as many victims as possible. Typically they send out hundreds or even thousands of probes to large blocks of IP addresses in a very short period of time. If a worm was designed to spread via TCP (as most of them are), during its propagation there will be a lot of corresponding TCP SYN packets sent out as it seeks vulnerable services in other hosts.

The patch a worm takes as it makes its way outside the corporate network, there are three possibile results to its SYN scan.

1-The first possibility is that the destination host is alive, and the corresponding vulnerable service that was targeted is running. Figure 1

Figure2: Destination host is living and the TCP port is open

2- The second possible result from the worm's SYN scan is that the destination host it attempts to connect to is not living, as shown below in Figure 2.

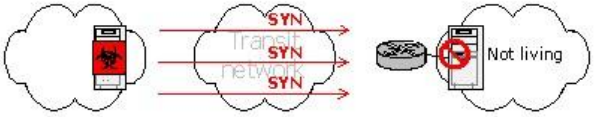

Figure3: Connect to a 'dead' destination host

3- The third possible result is that the destination is alive but connection attempts from the worm are not functional, as shown in Figure 3.

Figure4: Destination host with closed TCP port

3.4 Three steps to process a flow file for TCP flags

When doing flow-based analysis, a captured flow file can be processed by the following steps.

1)The first step is to search through the flow file, filter out all flow records that have only the SYN bit set, extract the source IP addresses of every flow record, count the occurrence of every unique IP, and then finally sort the records by the number of counts for each one. Following this process, we will end up with a suitable list of potentials. The administrator can set a threshold depending on the network size and traffic volume, whereby hosts whose counters are above the threshold should be considered as potentially malicious, and those under the threshold should be considered as benign.

2)The second step is to search through the flow file again to extract all flow records where the source IP addresses are the ones found in the "potential malicious" list as generated in step 1 above. By taking a second look at the flow file in this way, we will get a detailed connection table for every potential host. The results of this search will be used for our third and final step in this process, and will help us to further identify the behavior of our suspicious hosts.

3)The third step will give us some very meaningful data about the worm-infected hosts on our network. First read the output taken from step 2, and then for each host count the number of appearances of every unique destination port. Sort by the number of occurrences, and we will then get an IP address and its corresponding active ports table. The following is an example output generated by a little shell script that performs this task, as written by the author.

3.5 ICMP issues

One of the purposes of ICMP is to provide feedback about problems in the communication environment of a network. Sometimes a ICMP type/code in the flow records could also be used to help us locate the potential malicious hosts.

1- ICMP destination unreachable

According to ICMP implement guidelines, if the destination network or the destination host is unreachable, the gateway MAY send destination unreachable messages to the source host, as shown below in Figure 4.

Figure 5: Destination unreachable

2- ICMP port unreachable

For UDP requests, hosts with closed ports may send back ICMP port unreachable messages to the source host. If a worm spreads with UDP, it may then trigger many ICMP port unreachable flow records in the packets returned. This is shown below in Figure 5

Figure6: Destination host with closed UDP port

3-Pattern matching methods

Another ICMP-based flow analysis method is pattern matching. Some worms and network attacks are carried out using ICMP, as we saw with the W32.Nachi.worm. When a host is infected with the worm, it will send out ICMP echo requests to the outside with a fixed length of 92 bytes. So we simply need to filter out the flow records with ICMP type 8 that have a 92-byte packet length, and hosts infected with this worm will be caught.

3.6 Special zones in the Enterprise

We could use this characteristic to monitor the security of the servers using NetFlow.

1) ingress traffic

we find any flow record whereby the destination IP contains a server IP, but the destination port is not in the server's functional port list and additionally the TCP flags in the flow record contains ACK (but not RST/ACK), an alert should be triggered.The above suggestions perhaps indicates two points. First, it tells us that the firewall in front of the host has something wrong with it, as it has let a connection (which should be prohibited) get established. An exception to this would be that the connection launched by outside incorrectly contains only a ACK packet; regardless, this kind of connection should not have appeared. Secondly, the appearance of this flow record also indicates the server may have an abnormal port open to outside.

2) egress traffic

When we see any flow record whereby the source IP contains a server IP but the source port is not on the server's list of functioning ports, and additionally the TCP flags in the flow record are not RST/ACK, an alert should be triggered.As well, if we spot any data being transferred at the same time as the above, a red alert should be immediately raised! It is quite possible that the server has been broken into. Perhaps a backdoor has been actived, and maybe a new service has been enabled.

4. IMPLEMENTATION GUIDELINES

Let's see how "Troubleshooting Reports" are helpful in identifying a SYN scan and infected hosts. Generally TCP-SYN worm scan analysis is effective at switch level because of the visibility of LAN IP addresses. So it is better to choose a LAN interface/port for SYN scan analysis.

1-First step is to identify the conversations with only the SYN bit set. Using ManageEngine NetFlow Analyzer, it is possible to filter out potential sources trying to contact large number of destinations with SYN bit set.[14]

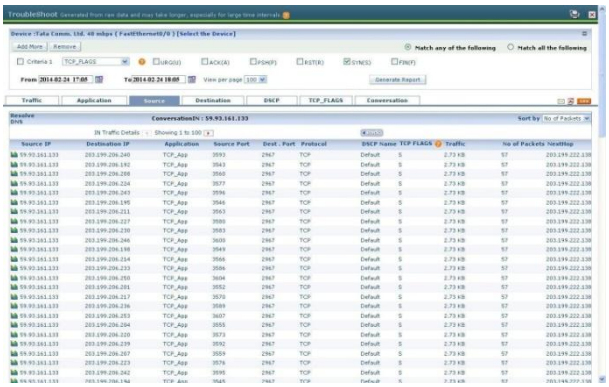

Figure7: Identify the conversations with only the SYN bit set

2. In the second step, we can drill down from each and every potential source to analyze the type of traffic. As you see in the below picture it seems to be a W32.Spybot.ACYR worm spreading through an un-patched windows machine using port 2967.[15]

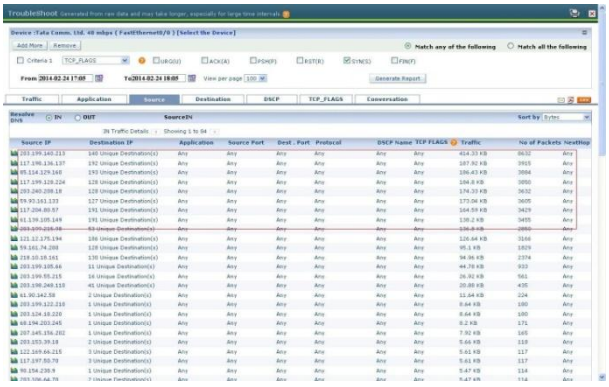

Figure8: Analyze the type of traffic.

5. CONCLUSION

This article series has discussed the flow-based detection of worms and abnormal activities. We talked about the basic concept of NetFlow, and then the first two of the five flow-based analysis methods were put forward. The last part of the paper discussed the final three analysis methods. In summary, these five methods of analysis are Top N and Baseline, Pattern Matching, TCP flags, ICMP issues and special zone for large enterprises. With these methods, network administrators can detect network-wide abnormities much more effectively.

There is no silver bullet for security detection on large network infrastructure, but with NetFlow we may attain further insight into the traffic crossing our entire network -- and make it run better.

.

References

[1].Al-Hammadi, Y.; Leckie, C, "Anomaly detection for Internet worms", Integrated Network Management, 2005, 9th IFIP/IEEE International Symposium on 15-19 May 2005, vol. 2, pp.133-146.

[2].Ellis R. D., Aiken G. J., Attwood S . K., and Tenaglia S ., "A Behavioral Approach to Worm Detec tion", WORM'04, Washington, DC, USA, 29th October 2004,vol.13, pp.71-79.

[3]Schechter S., Jung J., Berger W. A., "Fast Detection of Scanning Worm Infections", 7th International Symposium on Recent Advance in Intrusion Detection (RAID), September 2004, vol.19, pp-17-57.

[4].Shou-Chuan L., Wen-Chu K., and Mu-Cheng H.,"Defending against Internet Worm-like Infestations", the 18th "2008, vol.15,pp.581-586 International Conference on Advanced Information Networking and Applications, 2004 vol.1 pp.152-157.

[5] Weaver N., Paxson V., Staniford S., Cunningham R., "A taxonomy of computer worms", Procee dings of the 2003 ACM workshop on Rapid Malcode, 2003, vol.23,pp.785-791.

[6] "V . P. D. Moore et al., "Inside the Slammer Worm," IEEE Sec. & Privacy, vol. 1, 2003, pp. 33–39.

[7]D. G. Glazer, "Computer Worms," May 2005, http://www. R search.umbc .edu/~dgo rin1/is432/worms.htm.

[8] "Morris (Computer Worm)," retrieved July 2007, http://en. wikipe-dia.org/wiki/Morris_worm.

[9] "F-Secure Virus Descriptions: Nimda," retrieved July 2007,http://www.f-secure.com/v-descs/nimda.shtml, 2001.

[10] C. S . D. Moore, "The Spread of the Witty Worm," IEEE Sec. & Privacy, vol. 2, 2004, pp. 46–50.

[11] R. A. et al., Snort 2.1 Intrusion Detection, 2nd ed., Syngress.

[12] S . E. D. Bolzoni and P. Hartel, "POSEIDON: A 2-Tier Anomaly - Based Network Intrusion Detection System," Proc. 4th IEEE Int'l Wksp. Info. Assurance, 2006.

[13]Yini Wang, Sheng Wen, Yang Xiang, Senior Member, IEEE, and Wanlei Zhou, Senior Member, IEEE," Modeling the Propagation of Worms in Networks:ASurveyAug2012.

[14]Sheng Wen, Student Member, IEEE, Wei Zhou, Jun Zhang, Member, IEEE,Yang Xiang, Senior Member, IEEE, Wanlei Zhou, Senior Member, IEEE, and Weijia Jia, Senior Member, IEEE "Modeling Propagation Dynamics of Social Network Worms" Vol.24 , No .08,Aug 2013.

[15] Mohammad Reza Faghani, Student Member, IEEE, and Uyen Trang Nguyen, Member, IEEE" A Study of XSS Worm Propagation and Detection Mechanisms in Online Social Networks" VOL.8,No.11,November 2013.