IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555
Vol.6, No.2, Mar-April 2016

Honeypot system for local network attacks

M. Solomon Zemene
Department of Computer Science and Systems Engineering
Andhra University
Visakhapatnam, India

P.S. Avadhani
Department of Computer Science and Systems Engineering
Andhra University
Visakhapatnam, India

Abstract:---Honeypot systems are recently emerging network security technologies that can detect new attacks. Their main purpose is to study attackers and their attacking techniques and tactics. Local networks are exploited by variants of attacks and worms spreading within the network. In this paper we have implemented honeypot system in a local network. We have studied attacking malwares spreading across the Local Area Network. We have also observed the common services and port numbers which attackers make use of their vulnerabilities. We deployed the honeypot in a Linux virtual environment. In our honeypot system, we have used a low interaction honeypot called Dionaea. It has capabilities of logging different connection requests and emulates some services. Its main purpose is to collect malwares that spreads across networks using SMB services vulnerabilities in Windows systems.

Keywords: honeypot; malware; dionaea; Local network;

I. INTRODUCTION

In general, the continuous development of information technology (IT) enhanced each and every aspect of the society. In spite of the fact that every organization gets significant benefits from IT; it comes with much disaster caused by different attacks initiated by malicious groups. To reduce the threats of network security imposed by attackers, one needs to consider the vulnerable spots of its network at the early stage of infrastructure setup. More over after the initial infrastructure setup, it is helpful to constantly monitor and watch the security gaps in ones network, lest the newly emerging attacks and security gaps may lead to a disaster in the IT network framework as well as the data handling devices.

Different organizations use computers for various purposes around the world. Among the operation system they use, Windows operating system is the most used one. According to [1] about 1.25 billon Windows computers are running these days. Therefore, to address most of the network security issues, it needs to deal with Windows based attacks. As this operating system is used vastly, it has been challenged by different variants of cyber attacks. Even if it updates itself frequently diverse forms of worm, virus and adware are attacking this operating system. In our work we present a honeypot system implemented in a local network to track and collect Windows propagating worms.

Implementation of honeypots will be handy and powerful mechanism to track attackers and their new attack methods. In our work, we used server honeypot to study common vulnerable applications for worm propagation across local network. Honeypots can be computers or any other IT resources run as one of the network component and configured specially to lure attackers so that they will be ready to be attacked and even compromised, while they record and log all the traces and activities of the attacks to generate alerts and signals of malicious access.

II. BACKGROUND

A. Honeypots

Honeypots are defined as a resource whose sole purpose is to be compromised [2]. Based on their level of interaction, they can be classified as low, medium and high interaction honeypots. Low interactions honeypots are those with minimum interaction with attackers such as collecting connection logs; thus they are the least risky honeypots. The second kind of honeypot which is high interaction honeypot gives the attackers the real resource to interact with. It is the most risky honeypot due to almost the full freedom it gives to attackers. Between the low and high interaction honeypots, in the middle there is a medium interaction honeypot. Which are both medium in risk and interaction level.

According to their deployment honeypots can also be classified [3] as research and production honeypots. Production honeypots are used mostly for Intrusion prevention as an alert security entity in the working production network environment. Research honeypots are those honeypots used to study attackers and their tactics. These kinds of honeypots are used to develop a counter measure for specific threats and attacks.

In a campus network number of computers are interconnected to provide local as well as public services. In such compound networks, worm propagating across local networks are very annoying and may be the cause of loss of resources and time. Low interaction honeypot can be used to observe and control the propagation of worms across the network. In windows environment common file sharing and RPC services that are useful for local inter-networking, are suitable for worms to exploit common computer network vulnerabilities.

B. Worms

Worms [4] are program that are built to have a capability of propagating across a network by replicating themselves. They are different from virus by their automated spreading ability. Worms are known by their fast propagating behaviour. They can also invade vast numbers of systems and networks in a short period of time. They have impact on financial losses of organizations. There are different ways of worm propagating. The main spreading mechanism is by using buffer overflow. Worms can also spread using an exploit debug option in the routing program called sendmail. The other way of spreading is by cracking credentials to gain access to systems.

C. SMB protocol and its vulnerabilities

Nepenthes honeypot [5] is developed for malware collections and analysis. This honeypot has vulnerability modules which help to offer vulnerable service behaviour for malicious requests. It has also shellcode parsing module to parse and decode the malicious binaries. The other module that Nepenthes offer is fetch module which is built to enable downloading shellcodes after an URL is extracted from the malware. The last module is the logging module which logs all connection requests and packet information. If the vulnerability is new and not recognized by this honeypot, it will be logged for later manual analysis or for developing other honeypot tools. Nepenthes can even handle multi-stage spreading mechanisms of malwares and worms. This honeypot emulates IIS vulnerability and other backdoor used by worms.

Dionaea [6], which is the Nepenthes successor, is a malware collecting low interaction honeypot. It mainly focuses on vulnerabilities of Microsoft SMB service which is exploited by various malwares. The low interaction honeypot dionaea emulates vulnerabilities of windows file sharing and RPC services by keeping the real system from being accessed by attackers. This capability of dionaea, which is the main benefits of low-interaction honeypots, makes it less risky and easy to deploy. This honeypot tool captures worms and malwares coming from the internet. It has a module to parse a given Shellcode and extract URLs from it so that the worm can be easily be downloaded. Dionaea can detect malicious binaries by the help of Linux based virtual tool called LibEmu. In addition to normal IP packets, dionaea also supports IPv4 and Transport Layer Security (TLS). The honeypot minimize security risks by running in an isolated environment having non-administrator privilege. It emulates not only SMB protocol but also emulates HTTP, FTP, MSSQL, MySQL and VoIP services.

D. Related works

Honeyd [7], the most common low interaction honeypot, is built to simulate the network stack of multiple operating systems and services. The other low interaction honeypot is dionaea. Works [8] and [9] have done a malware collection using dionaea as a honeypot tool. The former work used Amazon cloud services to collect malwares from three different regions. They collected malwares and log connection requests coming from different parts of the world.

III. METHODOLOGY

We implement the honeypot system in VMware machine on top of Linux operating system host. Dionaea is installed in Ubuntu 12.04 VMware machine and set ready as a server (for 24 hours) waiting to catch attacks and worms. What dionaea does is when the attacker exploits the built simulated protocols such as SMB; it parses and extracts malware code and stores for analysis. This makes it ideal to implement dionaea in a campus network where SMB protocols are used in file sharing and printing purpose. Almost all malwares spread from host to host within the network using the vulnerabilities of RPC and SMB protocols. Since our goal is to study only the local attacks and worms, the honeypot is given local IP address. Unlike most of the former works [8], our honeypot is given local IP address. The honeypot was live for about 45 days. In the mean time, connection activity logs were taken daily for analysis.

IV. RESULTS AND DISCUSSIONS

Our honeypot was deployed from Oct 16, 2015 to Dec 01, 2015 which is totally 47 days. But the honeypot was out of duty for some technical issues for two days, thus it run for 45 days. Within those days, we have collected locally spreading malwares and connection requests. Since the numbers of hosts that are available in the local network are limited, we did not collect much numbers of malwares and connection requests. But the information we have gathered is helpful to analyze the security gap and figure out possible exploits available. To take security measures, it is mandatory to analyze the network and it behaviour. Generally this honeypot system results and analysis can represent most of the behaviours of local networks. Especially the connection requests made to the honeypot will characterize most of local network behaviour. But, the kind and variant of malwares available may vary from network to network.

The honeypot system recorded connection requests coming from different local hosts targeting different ports and services. Based on the kinds of connections we have divided them into three connection types. All the connection requests destined to multiple ports, Server Message Block (SMB) related connections and downloads that were offered to our honeypot are recorded by the system as shown in fig. 1. One can observe from the graph, that the three connection counts for each day are proportion to each other.

Specially, the number of SMB connection and the download offers are much related to each other. We observe the ratio of the two connections for each day is almost similar. The ratio of the download offer to the SMB connection count for each day is found to be around ½ with only very small deviation. In other words, out of all SMB packets half of them are download offers given to the honeypot. Other than SMB protocol connections, the system experienced different ports corresponding to various services emulated by the dionaea honeypot.

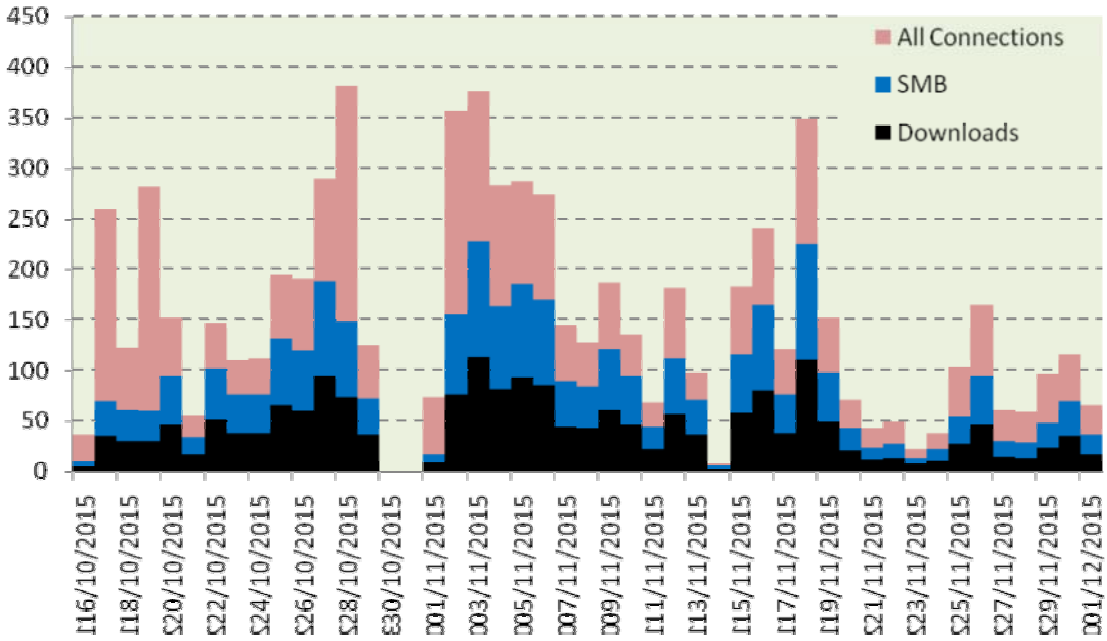

Figure 1. The overall connection attempts to the honeypot system.

A. *Targeted services*

Now let us see the TCP ports that setup connections to our honeypot. Fig. 2 shows the most frequently occurring port numbers. Our honeypot listens to the following port numbers;

21, 42, 69, 80, 135, 139, 445, 443, 1433, 3306, 5060, 5061

Based on our observation port 445 and port 139 are the leading port numbers that make connection requests to our honeypot. These two ports 445 (SMB directly over TCP/IP) and port 139 (SMB over NetBIOS) are of the same service [12] whose purpose is network files sharing for Windows. This dominancy of SMB connections shows that most of the attacks in Windows environment are originating from file sharing vulnerabilities. The other connections are HTTP (port 80), TFTP (port 69) and FTP (port 21 and 23) service connection requests.

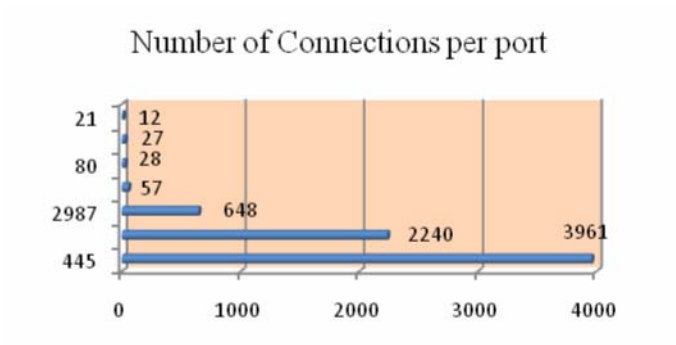

Figure 2. Top port numbers targeting the honeypot system.

B. *Attacking hosts*

Within these 45 days a total of 6,997 connections were recorded from about 122 local computers. Out of these 4,010 were accepted connections where as 2,978 were rejected local connections within the LAN. The rest 9 connections were ftp data listening connections. Most of the connections attempted were coming from the same subnet where the honeypot was running and the rest were from about 40 subnets. From this main subnet where the honeypot is running, we have detected about 5,769(3,680 accepted and 2,089 rejected) connections. Other than this subnet, one subnet has made 173 connections. Table 1 show the subnet and connection counts made to the honeypot. From this we can observe that attacks are not limited within the sub network, but jumps from one subnet to another.

TABLE I. CONNECTION DISTRIBUTIONS ACROSS SUBNETS.

	aaa.bbb.xxx.0	aaa.bbb.yyy.0	Other Subnets
Total Accepted Connections	3,680	173	157
Rejected Connections	2,089	110	779
SMB connections	3,669	173	119
Download offers	1,829	87	58

When we come to number of hosts which actively participate by sending connection requests, we noticed that most of the running systems connected to the LAN have made connection attempts to our honeypot. In fig. 3, we have shown the number of hosts requesting multiple ports and the hosts that

send SMB related connections to the honeypot. From the graph one can see that most of the hosts are participating with SMB connections. This indicates that most of daily connections that are shown up in the LAN are SMB service related. And hence, malwares can easily spread across the network using this service vulnerability.

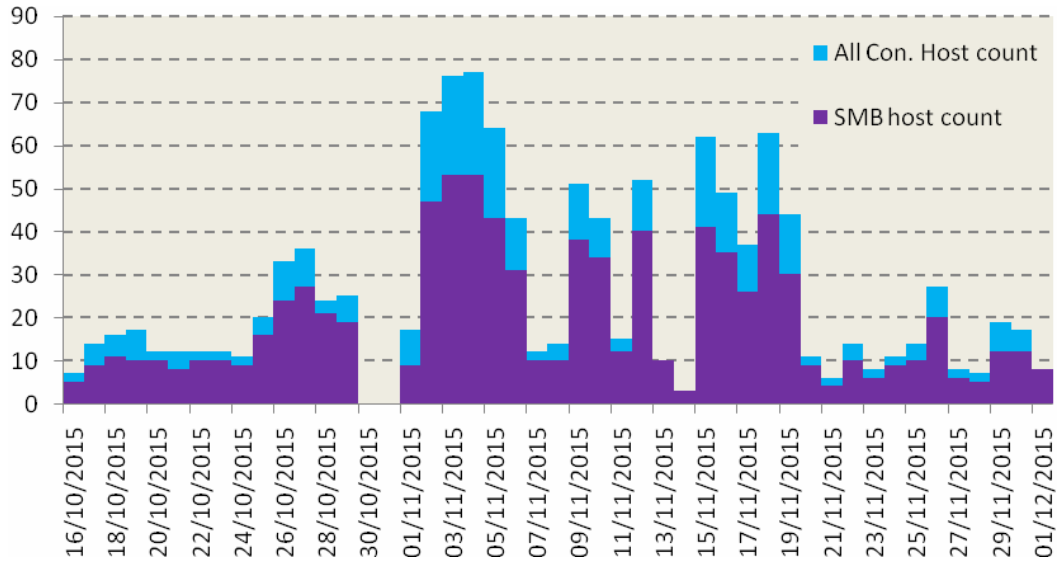

Figure 3. Number of hosts for SMB and other ports connection requests.

C. *Malwares collected*

Since compared to the internet local network has limited number of hosts, we have collected only two malwares that spread across the networks. Our network is infected by these malwares that spreads from one subnet to another. From the total number of hosts that made connection attempts to the honeypot, 85 of them offered malware URL downloads. We have seen about 943 unique URLs offered to our honeypot. From the two malwares one of them offers 1,961 URLs to our honeypot. Top 5 download URLs that are offered to our honeypot system are shown in the fig. 4. We have seen that the two malwares contaminated different subnets, so that a given subnet is attacked only by one of them. Table 2 show the malwares distribution in the local network.

TABLE II. REPEATEDLY DOWNLOADED MALWARES.

	downlo ads	offers	unique download URL	unique IP/port combinat ion	No. of offering hosts
Malware1	1951	1961	930	88	83
Malware2	13	13	11	2	2

The downloaded malwares have used different port numbers in their URL. They use 13 arbitrary non-standard port numbers within the range 1411 to 6653. Table 3 shows the top port numbers used by malwares to upload themselves to victim hosts.

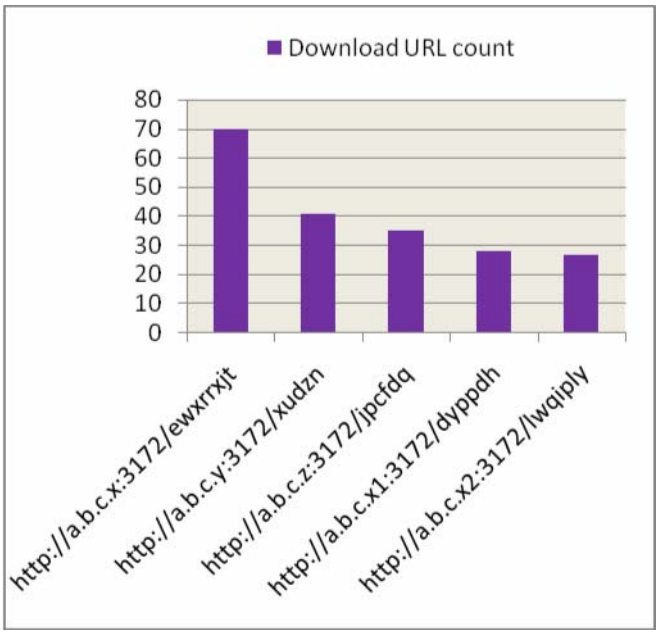

Figure 4. Top 5 download URLs offered.

TABLE III. TOP 5 PORT NUMBERS OFFERED FOR MALWARE DOWNLOADS.

	Used Port	Number of downloads	Victim Hosts Using the port
1	3172	576	54

2	1840	116	16
3	2883	54	2
4	1550	48	1
5	3643	37	2

We have seen the content of the malwares and the DLL they used. The malwares that are found in our local network are MS Windows GUI executable programs developed for 32 bit processors. These malicious programs are built with function calls and APIs that most malwares and worms [10] are built from. We can mention some of the function calls used by collected malwares. Windows function calls GetUserNameA, VirtualQuery, IsClipboardFormatAvailable and others are used by the malwares. Based on the [11] we have observed some function calls that are used by common spywares are also shown on the malwares we have collected. Typically, they used function call GetDC that enable them to capture user screen. To identify the version of the Operating system, they use GetVersionEx function. They also used VirtualAllocEx to allocate memory in the remote process. The malwares even try to change the protected read only memory region to an executable by using VirtualProtectEx function.

V. CONCLUSION

We have deployed honeypot system to collect malwares and log connection attempts in the local network. Within 45 days, our honeypot system encounters 6,997 connection requests. From these connection requests about 6,201 are connections related to Windows file sharing and printing services. We have studied the common exploits of local networks from the honeypot. We can conclude that most of the attacks in local networks are using SMB service vulnerabilities. Especially malwares spread through the network using this service. Therefore, to control and manage the spreading of malwares across the network, one needs to consider security aspects of services used for file sharing and printing purpose. These services must be configured and used in appropriate ways to limit spreading of malwares. The honeypot system not only collects malwares but also helps to isolate the hosts that are infected by malwares. The logged information gathered from different connection requests targeting variety of ports have also assist to figure out other possible attack areas of services and port numbers. From the local network perspective, one ought to consider the possible attacks coming from the network, and we recommend taking other security measures such as anti-virus and firewalls that will secure individual hosts. Besides, organizations need to equip their networks with network security tools and methods that will add extra layer of security.

REFERENCES

[1] Insider www.businessinsider.com by Matt Rosoff

[2] L. Spitzner, "Honeypots: Tracking Hackers," Boston, USA: Addison-Wesley, Parson Education, ISBN 0-321-10895-7, 2003.

[3] The Honeynet Project. Know Your Enemy : Honeynets (May 2005) http://old.honeynet.org/papers/honeynet/

[4] https://crypto.stanford.edu/cs155old/cs155-spring06/16-worms.pdf

[5] P. Baecher, M. Koetter, T. Holz, M. Dornseif, and F.C. Freiling. "The Nepenthes Platform: An Efficient Approach to Collect Malware." In Diego Zamboni and Christopher Krügel, editors, RAID, volume 4219 of Lecture Notes in Computer Science, pages 165–184. Springer, 2006.

[6] Dionaea: A low interaction honeypot. https://github.com/rep/dionaea.

[7] Developments of the Honeyd Virtual Honeypot http://www.honeyd.org.

[8] Al Awadhi, E.Salah, K. ; Martin, T. "Assessing the Security of the Cloud Environment" Conference and Exhibition (GCC), 2013 7th IEEE Nov. 2013 pp: 251 - 256

[9] Saxena, U. Bachhan, O.P. ; Majumdar, R. "Static and dynamic malware behavioral analysis based on arm based board" Conf. on Computing for Sustainable Global Development (INDIACom), Pp 272 - 277 Mar 2015

[10] Youngjoon Ki, Eunjin Kim, and Huy Kang Kim "A Novel Approach to Detect Malware Based on API Call Sequence Analysis" International Journal of Distributed Sensor Networks Volume 2015.

[11] Windows Functions in Malware Analysis – Cheat Sheet http://resources.infosecinstitute.com/

[12] Microsoft SMB Protocol and CIFS Protocol Overview https://msdn.microsoft.com

AUTHORS' PROFILES

Solomon Zemene received his M.Tech in Electronics and Computer Engineering from Addis Ababa University, Ethiopia. He is currently a PhD candidate in Andhra University, Visakhapatnam, India. His research interest includes Network Security and Cryptography.

Dr. P. S. Avadhani is a professor in the department of Computer Science and Engineering of Andhra University. He did his Masters Degree and PhD from IIT, Kanpur. He has guided 15 Ph. D Scholars from various institutes. He received many honors and he has been the member for many expert committees, member of Board of Studies for various universities, Resource person etc for various organizations. He is a Life Member in CSI, AMTI, ISIAM, ISTE, YHAI and in the International Society on Education Technology. He is also a Member of IEEE, and a Member in AICT. He has published about 200 refereed scientific papers.. His research areas include Cryptography, Data Security, Algorithms, and Computer Graphics, Digital Forensics and Cyber Security. He has supervised the dissertations of 15 doctoral students. Invited by Microsoft Corporation to Malaysia to attend a conference on .Net Technologies, June 2004. Served as member of National Board of Accreditation of AICTE and inspected various Engineering Colleges all over India as a member of NBA. Delivered Invited talks at many National and International Conferences, Chaired many sessions at National and International Conferences at many places in India and abroad.