Asian Journal of Business and Technology
Vol. 3, No. 1, (2020)
ISSN 2651-6713 (Print)
ISSN 2651-6721 (Online)

Honeypot Deployment for Educational Purposes at one University in Northern Luzon, Philippines

Jan Henry D. Mangapot
University of Luzon
Perez Boulevard, Dagupan City, Pangasinan, 2400 Philippines

Abstract *- All over the world, networks of all sizes are under attack daily. Modern firewalls, intrusion detection systems (IDS) and other detection and prevention tools are used to defend the networks against these malicious attacks. Traditionally, these tools were designed and built based on information from past attempts at breaking into a system, and from the stolen data. A different tool called honeypot was developed to capture the important information for improving the design of a firewall, IDS, anti-malware application or any other security solution. It is designed to reduce or eliminate the need for a system breach in order to learn the intentions, tools and procedures of the hackers in penetrating the system. A honeypot simply replicates a real network with fake data that a hacker will attempt to steal from. This paper analyzed the concept of honeypot with its functions and technology, which is becoming not just an important component in a layered system of protection against intrusions but also as valuable simulation resource for teaching security concepts in academic institutions like University of Luzon. Moreover, it implemented a Windows-based honeypot as additional learning tool for the teachers and students of its information security courses. Using combined criteria consisting of detection scope, emulation accuracy, quality of collected data, scalability and performance, reliability, extensibility, ease of use and setting up, embeddability, support and costs, the study deployed a low interaction, server-side type of honeypot appropriate for educational purposes at the University of Luzon. Furthermore, it presented honeypot's configuration and deployment requirements and highlighted the features and functionalities as additional security tool by simulating attacks using vulnerability assessment and penetration testing tools including Kali Linux and Nessus.*

Keywords: honeypot deployment

INTRODUCTION

All over the world, networks of all sizes are under attack daily. Modern firewalls, intrusion detection systems (IDS) and other detection and prevention tools attempt to defend networks against these malicious attacks. Traditionally, these tools have been built based on information from past attempts, successful or otherwise, at breaking into a system, and from stolen data [1]. A honeypot is designed to reduce or eliminate the need for a system breach in order to learn about what the hackers are using to penetrate the network.

A honeypot simply replicates a real network with fake data that a hacker will attempt to steal from, thus showing what their intensions are. These informations are important in the design of a better firewall, IDS, anti-malware application and other security solutions.

Honeypots have demonstrated their value as a research tool in the field of Information Technology and as a powerful educational tool in modern computer laboratories [2]. Honeypots are, in their most basic form, fake information servers strategically positioned in a test network, which are fed with false information disguised as files of classified nature. In turn, these servers are initially configured in a way that is difficult, but not impossible , to break into them by an attacker, exposing them deliberately and making them highly attractive for a hacker in search of a target. Finally, the server is loaded with monitoring and tracking tools so every step and trace of activity left by a hacker can be recorded in a log, indicating those traces of activity in a detailed way.

Honeypots can be used for many different purposes including, the monitoring of scanning activity of worms or bots, learning about compromised nodes, identifying new exploits and vulnerabilities, capturing new malware, studying hacker behaviour and looking for internal attacks from insiders, among others. Naturally, the purpose of deployment impacts both the honeypot technology selection and the way it will be deployed [3].

Many researchers, organizations and academic institutions are using trap-style networks to learn the tactics, techniques and procedures used by the hacker community to break into information databases without authorization, which could contain potentially sensitive information. Additionally, honeypots provide teachers and students with a tool that allows them to study security events thoroughly and in a modular way, which is a very desirable characteristic when it comes to teaching Information Security courses.

This research aimed to implement a honeypot in the laboratory room of University of Luzon's College of Computer Studies for educational purposes and is bound to the following objectives: (1) To find out the type of honeypot that should be implemented for University of Luzon in terms of level of Interaction, implementation environment, and type of communication architecture; (2) To determine the criteria to be used in selecting a honeypot for University of Luzon; and (3) To determine the functionalities of the honeypot to be installed in the laboratory room of University of Luzon.

METHODOLOGY

The research work includes the identification of the right type of honeypot to be deployed at University of Luzon. It was based on factors including the level of interaction, deployment environment, ease of use among others. The honeypot configured and deployed inside the computer laboratory of the University in a server mode. Attack simulations were conducted using penetration and vulnerability assessment tools so as to easily evaluate the performance and other functionalities of the deployed honeypot. A dedicated computer was used to attack the target computer where the honeypot is installed. Simply put, the honeypot installed and deployed in a controlled environment and for educational purposes only.

This study used focus group discussion and documents review in collecting the project study's data. Focus group discussion was utilized in the preparation of the computer laboratory to be used in the honeypot set-up and configuration. Discussion topics include identifying the computer to be used as honeypot, the attack computer specification, attack software to be used, attack simulation details and expected data to capture, with the Laboratory Administrator, one student assistant and the Dean of the College of Computer Studies as major participants.

White papers, blogs and research literatures related to honeypot, Windows-based low interaction honeypot software, vulnerability assessment and penetration testing software were studied and evaluated to identify the right mix of references for the chosen topic. Document review was also utilized in gathering information on criterions for evaluating honeypot, simulation attack methodologies and application of honeypots in educational institution.

RESULTS AND DISCUSSION

Type of honeypot that should be implemented for University of Luzon

In terms of level of interaction, low interaction honeypot is suitable for University of Luzon's operation compared to high interaction honeypot. A significant advantage of low interaction honeypot is that it requires very little maintenance and setup time. Also, since low interaction honeypot doesn't use real operating system and just emulates the services, there's no risk involved with allowing an attacker to be on the honeypot. Attackers can abuse the honeypot as much as they want without exposing critical systems and affecting other systems in the production environment.

In term of implementation environment, since the honeypot to be deployed is for educational purposes, research honeypot is appropriate. It will not be part of the institution's main production network thus has no direct security value to the institution's infrastructure. Finally, in terms of the type of communication architecture, server-side honeypot is more appropriate since the honeypot is to be deployed to wait for attacks rather than it initiating the communication. Furthermore, server-side honeypot require less implementation effort than client-side honeypot as it doesn't have to have a sophisticated crawler engine.

Criteria to be used in selecting a honeypot for University of Luzon

The evaluation criteria was adopted in selecting the low interaction server-side honeypot for University of Luzon. The criterions are the following: Detection Scope, Accuracy of Emulation, Quality of Collected Data, Scalability and Performance, Reliability, Extensibility, Ease of Use and Setting Up, Embeddability, Support and Costs [3].

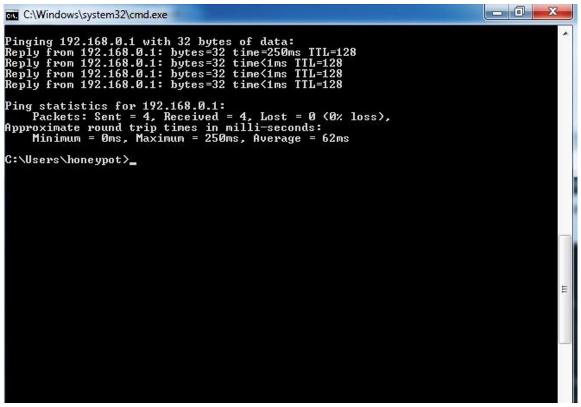
The search for low interaction server-side honeypot suitable for University of Luzon involved priority criterions before the application of the criterions stated above. The criterions include costs (it should be free), available documentation, online reviews and ease of setup. Based on the priority criterions, the selection was trimmed down between Honeyd and KFSensor – the two being free and popular. Finally, KFSensor was chosen over Honeyd because the latter is more difficult to set up. There was limited documentation especially on its network configuration and the scripts used to run services on the virtual honeypots. KFSensor on the other hand, is easy to install and configure. All it takes is few minutes for set up and to become operational. There's no need for special hardware and it runs even on low specification Windows machine. Its straightforward Windows interface controls all functionality. There's no need to edit complex configuration files and it comes pre-configured with all the major systems required.

Functionalities of the honeypot

On top of the criterions in selecting the honeypot suitable for University of Luzon's objective, the following are the minimum functional requirements of the honeypot:

a. Ability to adapt to different ports as they are being scanned
b. Ability to log activities on specific ports

There are three (3) attack simulations conducted to determine how the honeypot handle and record attacks. The first attack simulation was the used of the ping command. The ping (Pocket Inter-Network Groper) command is a security tool, which is used to determine if a particular IP address is existing and can accept requests. It is used to diagnose if a computer is actually operating. It works by sending an Internet Control Message Protocol (ICMP) echo request to a specified interface on the network and wait for a reply. It is usually used for troubleshooting to test connectivity and determine response time.



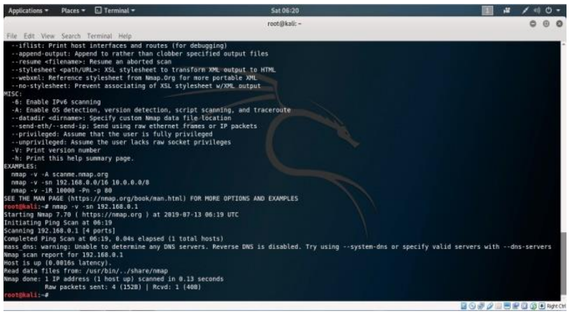
```
Pinging 192.168.0.1 with 32 bytes of data:
Reply from 192.168.0.1: bytes=32 time=250ms TTL=128
Reply from 192.168.0.1: bytes=32 time<1ms TTL=128
Reply from 192.168.0.1: bytes=32 time<1ms TTL=128
Reply from 192.168.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 250ms, Average = 62ms

C:\Users\honeypot>_
```

Figure 1. Pinging the honeypot

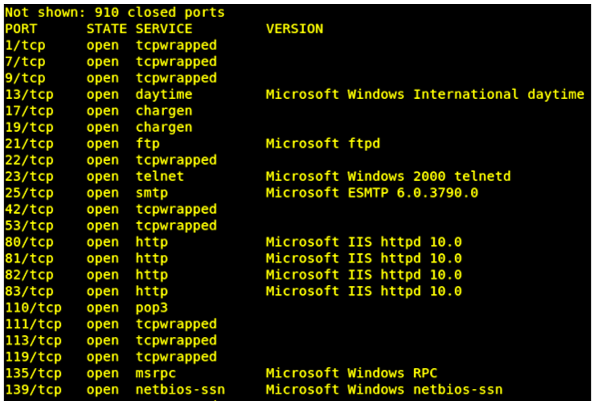
The actual screenshot in using the ping command is shown in Figure 1. The ip address used is the ip address of the honeypot.



Applications ▾ Places ▾ Terminal ▾
Sat 06:20
root@kali: ~
File Edit View Search Terminal Help

```
  --iflist: Print host interfaces and routes (for debugging)
  --append-output: Append to rather than clobber specified output files
  --resume <filename>: Resume an aborted scan
  --stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML
  --webxml: Reference stylesheet from Nmap.Org for more portable XML
  --no-stylesheet: Prevent associating of XSL stylesheet w/XML output
MISC:
  -6: Enable IPv6 scanning
  -A: Enable OS detection, version detection, script scanning, and traceroute
  --datadir <dirname>: Specify custom Nmap data file location
  --send-eth/--send-ip: Send using raw ethernet frames or IP packets
  --privileged: Assume that the user is fully privileged
  --unprivileged: Assume the user lacks raw socket privileges
  -V: Print version number
  -h: Print this help summary page.
EXAMPLES:
  nmap -v -A scanme.nmap.org
  nmap -v -sn 192.168.0.0/16 10.0.0.0/8
  nmap -v -iR 10000 -Pn -p 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES
root@kali:~# nmap -v -sn 192.168.0.1
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-13 06:19 UTC
Initiating Ping Scan at 06:19
Scanning 192.168.0.1 [4 ports]
Completed Ping Scan at 06:19, 0.04s elapsed (1 total hosts)
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.0.1
Host is up (0.0016s latency).
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.13 seconds
           Raw packets sent: 4 (152B) | Rcvd: 1 (40B)
root@kali:~#
```

Figure 2. Scanning the Honeypot Using Nmap



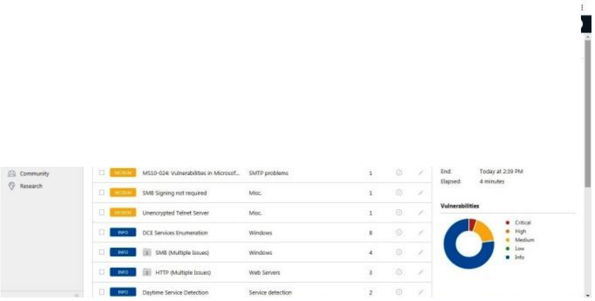
```
Not shown: 910 closed ports
PORT      STATE SERVICE        VERSION
1/tcp     open  tcpwrapped
7/tcp     open  tcpwrapped
9/tcp     open  tcpwrapped
13/tcp    open  daytime        Microsoft Windows International daytime
17/tcp    open  chargen
19/tcp    open  chargen
21/tcp    open  ftp            Microsoft ftpd
22/tcp    open  tcpwrapped
23/tcp    open  telnet         Microsoft Windows 2000 telnetd
25/tcp    open  smtp           Microsoft ESMTP 6.0.3790.0
42/tcp    open  tcpwrapped
53/tcp    open  tcpwrapped
80/tcp    open  http           Microsoft IIS httpd 10.0
81/tcp    open  http           Microsoft IIS httpd 10.0
82/tcp    open  http           Microsoft IIS httpd 10.0
83/tcp    open  http           Microsoft IIS httpd 10.0
110/tcp   open  pop3
111/tcp   open  tcpwrapped
113/tcp   open  tcpwrapped
119/tcp   open  tcpwrapped
135/tcp   open  msrpc          Microsoft Windows RPC
139/tcp   open  netbios-ssn    Microsoft Windows netbios-ssn
```

Figure 3. Nmap Report Showing Open Ports

The second attack simulation is scanning the honeypot using Kali Linux Nmap. The nmap reports show, among others, that the host is up (Figure 4) and the open ports (Figure 5). Nmap (or Network MAPper) is a free and powerful utility for network discovery and security auditing. It is useful for tasks such as network inventory, managing service upgrade schedules and monitoring host or service uptime. Nmap uses raw ip packets to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and versions) they are running, what type of packet filters/firewalls are in use, and more other characteristics [4].

The third attack simulation is scanning the honeypot using Nessus. Nessus is a vulnerability scanner used in vulnerability assessments and penetration testing engagements. It uses the Common Vulnerabilities and Exposures architecture for easy cross-linking between compliant security tools. It employs the Nessus Attack Scripting Language (NASL), a simple language that describes individual threats and potential attacks. Nessus performs its scans by utilizing plugins, which run against each host on the network in order to identify vulnerabilities. Plugins can be thought of as individual pieces of codes that Nessus uses to conduct individual scan types on targets. Plugins are numerous and wide in their capabilities. For instance, a plugin could be launched and targeted at a host to identify which operating systems and services are running on which ports [5].

Figure 4 shows the result in the Nessus monitor after a Nessus scan. As shown in the figure, vulnerabilities are presented and categorized according to criticality level.

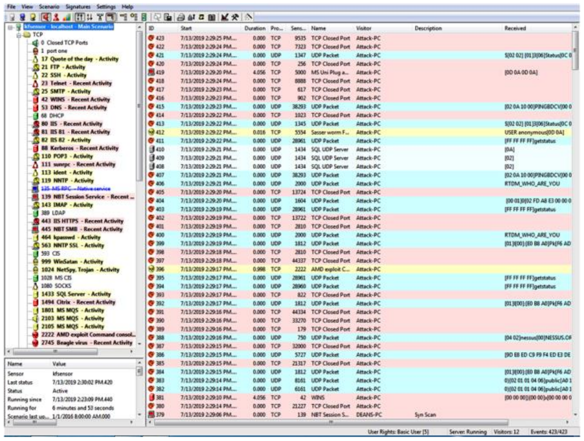


□	MEDIUM	MS10-024: Vulnerabilities in Microsof...	SMTP problems	1
□	MEDIUM	SMB Signing not required	Misc.	1
□	MEDIUM	Unencrypted Telnet Server	Misc.	1
□	INFO	DCE Services Enumeration	Windows	8
□	INFO	2 SMB (Multiple Issues)	Windows	4
□	INFO	3 HTTP (Multiple Issues)	Web Servers	3
□	INFO	Daytime Service Detection	Service detection	2

End: Today at 2:39 PM
Elapsed: 4 minutes

Figure 4. Result of a Nessus scan

The screenshots presented above are from the simulated attacks conducted from the attack computer. The corresponding report from these attacks as captured by the honeypot is presented in Figure 5. As shown, different colors are used to separate each attack.



kfsensor - localhost - Main Scenario
TCP
0 Closed TCP Ports
1 port one
17 Quote of the day - Activity
21 FTP - Activity
22 SSH - Activity
23 Telnet - Recent Activity
25 SMTP - Activity
42 WINS - Recent Activity
53 DNS - Recent Activity
68 DHCP
80 IIS - Recent Activity
81 IIS 81 - Recent Activity
82 IIS 82 - Activity
88 Kerberos - Recent Activity
110 POP3 - Activity
111 sunrpc - Recent Activity
113 ident - Activity
119 NNTP - Activity
~~135 MS RPC - Native service~~
139 NBT Session Service - Recent ...
143 IMAP - Activity
389 LDAP
443 IIS HTTPS - Recent Activity
445 NBT SMB - Recent Activity
464 kpasswd - Activity
563 NNTP SSL - Activity
593 CIS
999 WinSatan - Activity
1024 NetSpy. Trojan - Activity
1028 MS CIS
1080 SOCKS
1433 SQL Server - Activity
1494 Citrix - Recent Activity
1801 MS MQS - Activity
2103 MS MQS - Activity
2105 MS MQS - Activity
2222 AMD exploit Command consol...
2745 Beagle virus - Recent Activity

Name	Value
Sensor	kfsensor
Last status	7/13/2019 2:30:02 PM.420
Status	Active
Running since	7/13/2019 2:23:09 PM.440
Running for	6 minutes and 53 seconds
Scenario last up...	1/1/2016 8:00:00 AM.000

ID	Start	Duration	Pro...	Sens...	Name	Visitor	Description	Received
423	7/13/2019 2:29:25 PM...	0.000	TCP	9535	TCP Closed Port	Attack-PC		
422	7/13/2019 2:29:24 PM...	0.000	TCP	7323	TCP Closed Port	Attack-PC		
421	7/13/2019 2:29:24 PM...	0.000	UDP	1347	UDP Packet	Attack-PC		$[02 02] [01]3[06]Status[0C 0
420	7/13/2019 2:29:24 PM...	0.000	TCP	256	TCP Closed Port	Attack-PC		
419	7/13/2019 2:29:20 PM...	4.056	TCP	5000	MS Uni Plug a...	Attack-PC		[0D 0A 0D 0A]
418	7/13/2019 2:29:24 PM...	0.000	TCP	8888	TCP Closed Port	Attack-PC		
417	7/13/2019 2:29:23 PM...	0.000	TCP	617	TCP Closed Port	Attack-PC		
416	7/13/2019 2:29:23 PM...	0.000	TCP	902	TCP Closed Port	Attack-PC		
415	7/13/2019 2:29:23 PM...	0.000	UDP	38293	UDP Packet	Attack-PC		[02 0A 10 00]PINGBDCV[00 0
414	7/13/2019 2:29:22 PM...	0.000	TCP	1023	TCP Closed Port	Attack-PC		
413	7/13/2019 2:29:22 PM...	0.000	UDP	1345	UDP Packet	Attack-PC		$[02 02] [01]3[06]Status[0C 0
412	7/13/2019 2:29:22 PM...	0.016	TCP	5554	Sasser worm F...	Attack-PC		USER anonymous[0D 0A]
411	7/13/2019 2:29:22 PM...	0.000	UDP	28961	UDP Packet	Attack-PC		[FF FF FF FF]getstatus
410	7/13/2019 2:29:21 PM...	0.000	UDP	1434	SQL UDP Server	Attack-PC		[0A]
409	7/13/2019 2:29:21 PM...	0.000	UDP	1434	SQL UDP Server	Attack-PC		[02]
408	7/13/2019 2:29:21 PM...	0.000	UDP	1434	SQL UDP Server	Attack-PC		[02]
407	7/13/2019 2:29:21 PM...	0.000	UDP	38293	UDP Packet	Attack-PC		[02 0A 10 00]PINGBDCV[00 0
406	7/13/2019 2:29:21 PM...	0.000	UDP	2000	UDP Packet	Attack-PC		RTDM_WHO_ARE_YOU
405	7/13/2019 2:29:20 PM...	0.000	TCP	13724	TCP Closed Port	Attack-PC		
404	7/13/2019 2:29:20 PM...	0.000	UDP	1604	UDP Packet	Attack-PC		[00 01]0[02 FD A8 E3 00 00 0
403	7/13/2019 2:29:19 PM...	0.000	UDP	28961	UDP Packet	Attack-PC		[FF FF FF FF]getstatus
402	7/13/2019 2:29:19 PM...	0.000	TCP	13722	TCP Closed Port	Attack-PC		
401	7/13/2019 2:29:19 PM...	0.000	TCP	2810	TCP Closed Port	Attack-PC		
400	7/13/2019 2:29:19 PM...	0.000	UDP	2000	UDP Packet	Attack-PC		RTDM_WHO_ARE_YOU
399	7/13/2019 2:29:19 PM...	0.000	UDP	1812	UDP Packet	Attack-PC		[01]I[00]-[E0 B8 A0]Pk[F6 AD
398	7/13/2019 2:29:18 PM...	0.000	TCP	2810	TCP Closed Port	Attack-PC		
397	7/13/2019 2:29:18 PM...	0.000	TCP	44337	TCP Closed Port	Attack-PC		
396	7/13/2019 2:29:17 PM...	0.998	TCP	2222	AMD exploit C...	Attack-PC		
395	7/13/2019 2:29:17 PM...	0.000	UDP	28961	UDP Packet	Attack-PC		[FF FF FF FF]getstatus
394	7/13/2019 2:29:17 PM...	0.000	UDP	28960	UDP Packet	Attack-PC		[FF FF FF FF]getstatus
393	7/13/2019 2:29:17 PM...	0.000	TCP	822	TCP Closed Port	Attack-PC		
392	7/13/2019 2:29:17 PM...	0.000	UDP	1812	UDP Packet	Attack-PC		[01]I[00]-[E0 B8 A0]Pk[F6 AD
391	7/13/2019 2:29:16 PM...	0.000	TCP	44334	TCP Closed Port	Attack-PC		
390	7/13/2019 2:29:16 PM...	0.000	TCP	33270	TCP Closed Port	Attack-PC		
389	7/13/2019 2:29:16 PM...	0.000	TCP	179	TCP Closed Port	Attack-PC		
388	7/13/2019 2:29:16 PM...	0.000	UDP	750	UDP Packet	Attack-PC		[04 02]nessus[00]NESSUS.OF
387	7/13/2019 2:29:15 PM...	0.000	TCP	32000	TCP Closed Port	Attack-PC		
386	7/13/2019 2:29:15 PM...	0.000	UDP	5727	UDP Packet	Attack-PC		[9D E8 ED C9 F9 F4 ED E3 DE
385	7/13/2019 2:29:15 PM...	0.000	TCP	21317	TCP Closed Port	Attack-PC		
384	7/13/2019 2:29:15 PM...	0.000	UDP	1812	UDP Packet	Attack-PC		[01]I[00]-[E0 B8 A0]Pk[F6 AD
383	7/13/2019 2:29:14 PM...	0.000	UDP	8161	UDP Packet	Attack-PC		0)[02 01 01 04 06]public[A0 1
382	7/13/2019 2:29:14 PM...	0.000	UDP	6161	UDP Packet	Attack-PC		0)[02 01 01 04 06]public[A0 1
381	7/13/2019 2:29:10 PM...	4.056	TCP	42	WINS	Attack-PC		[00 00 00])[00 00]-[00 00 00 0
380	7/13/2019 2:29:14 PM...	0.000	TCP	21227	TCP Closed Port	Attack-PC		
379	7/13/2019 2:29:06 PM...	0.000	TCP	139	NBT Session S...	DEANS-PC	Syn Scan	

User Rights: Basic User [5] | Server: Running | Visitors: 12 | Events: 423/423

Figure 5. Report of the KFSensor Honeypot

The honeypot used is a Windows-based KFSensor. KFSensor as a security solution that acts as a honeypot designed for Windows systems whose job is to attract and detect attackers and worms by simulating vulnerable system services and Trojans. It acts as a decoy server to divert attacks from critical systems and provide a higher level of information that can be achieved by using firewalls and network intrusion systems alone [6]. It contains many innovative and unique features such as remote management, a Snort compatible signature engine and emulations of Windows networking protocols. It is convenient to use due to its GUI-based management console, extensive documentation and low maintenance requirement.

CONCLUSIONS

The type of honeypot appropriate for University of Luzon's purpose of implementation is a server-side low interaction honeypot. Furthermore, it is a research honeypot since it will be implemented for educational purposes and will not be deployed in the production network. Criteria for evaluating the KFSensor honeypot for University of Luzon's honeypot implementation like Detection Scope, Accuracy of Emulation, Quality of Collected Data, Scalability and performance, Reliability, Extensibility, Ease of Use and Setting Up, Embeddability, Support and Costs were useful.

RECOMMENDATIONS

It is recommended that the University of Luzon should implement a Windows-based low interaction server-side honeypot for its computer laboratory. The concept of honeypot including its advantages and functionalities as additional layer of security solution can be learned and appreciated by its users easily using this type of honeypot.

REFERENCES

[1] Ponten, A. (2017). *Evaluation of Low Interaction Honeypots on the University Network*. Retrieved on April 10, 2019 from http://lnu.diva-portal.org/smash/record.jsf?pid=diva2%3A1121560&dswid=-6419.

[2] Lopez, M., and C. Resendez (2008). *Honeypots: Basic Concepts, Classification and Education Use as Resources in Information Security Education and Courses*. Proceedings of the Informing Sciences & IT Education Conference 2008.

[3] Grudziecki, T., Jacewicz, P., Juszczyk, L., Kijewski, P., Pawlimski, P. (2012). *Proactive Detection of Security Incidents*. Retrieved on Jannuary 11, 2019 from enisa website, www.enisa.europa.eu .

[4]Lyon, G. (2019). *Nmap Package Description*. Retrieved on April 9, 2019 from https://tools.kali.org/information-gathering/nmap.

[5] Obbayi, L. (2018). *A Brief Introduction to the Nessus Vulnerability Scanner*. Retrieved on April 8, 2019 from https://resources.infosecinstitute.com/a-brief-introduction-to-the-nessus-vulnerability-scanner/#gref.

[6] Chandel, R. (2019). *Threat Detection for your Network Using Kfsensor Honeypot*. Retrieved on April 8, 2019 from https://www.hackingarticles.in/hack-the-android4-walkthrough-ctf-challenge/.