Enterprise Cyber security Management Using Scapy and Kali Linux

Dewan Juel Rahman[1], Al Amin Mahfuz [2], Most. Mithyla Zaman[3]

[1]Associate Professor and Head, Department of Computer Science and Engineering
Rajshahi Science and Technology University, Natore-6400, Bangladesh
[2]Lecturer, Department of Computer Science and Engineering
Rajshahi Science and Technology University, Natore-6400, Bangladesh
[3]Research Student, Department of Computer Science and Engineering
United International University, Dhaka, Bangladesh

DOI: 10.29322/IJSRP.14.06.2024.p15015
10.29322/IJSRP.14.06.2023.p15015

Paper Received Date: 17th April 2024
Paper Acceptance Date: 28th May 2024
Paper Publication Date: 6th June 2024

Abstract- The source of this work is Enterprise Cyber Security Management using Kali Linux and Scapy. Scapy is intended to be used to interpret a large number of protocols, transfer them over the network, accept them, include requests and answers, and much more. A larger percentage of the most difficult tasks, such as identification, vestige routing, probing, unit testing, raids, or network innovation, can be handled by this with ease. It also performs admirably at a plethora of other inelastic tasks that most other tools are unable to handle. We also use Kali Linux because it enables us to evaluate the security of our network using tools and methods that an attacker would use, allowing us to identify and fix problems before an actual attacker does. The proposed idea can give a company a security mechanism that will make it easy to monitor the internet activity of its personnel by utilizing these resources.

Keywords: - **Internet *Security, Kali Linux, Scapy, Internet Security Monitoring.***

I. INTRODUCTION

Today, everyone around us uses the internet, and there are so many ways to communicate with each other via social media. Is this a very good thing? Yes, to communicate and share data and information with everyone, it can play a vital role. But in some cases, like in the middle of a meeting, while working, or even in the classroom, it will decrease productivity or sometimes spoil the whole purpose. So, the proposed system, Internal Security Monitoring of an Organization by Scapy and Kali Linux, can be a time-saving solution to these kinds of problems.

A Motivation

The desire to work on a difficult project in a fascinating field of network security was the main driving force behind this project. Lectures do not provide the chance to learn about a new topic in network security. Our greatest motivation came from a desire to use new tools.

B Project Goals

The main goal of this paper is to provide a security mechanism for an organization that will enable them to track their employees's online activities easily and examine if they are working properly or not. So that employees can be productive and efficient, and for that purpose, preventing access to specific websites for different groups of employees is important.

II. EXISTING SYSTEM & RESEARCH SCOPE

Employees of organizations sometimes browse internet for Facebook, We Tube or other sites at their workstation. And admin does not want this. But for some employees it is important to have access to those sites.

In figure: 1.1, we can see an organization where this project can be used. There is an admin who wants to monitor the employees. Employees will be divided into two or more groups as we can see in the figure ―Group A‖ and ―Group B‖. Employees of Group A can access to the restricted sites initially but if admin wants then we can restrict them from those websites. And employees of Group B are initially blocked from those sites and admin can also open those sites for them if needed.

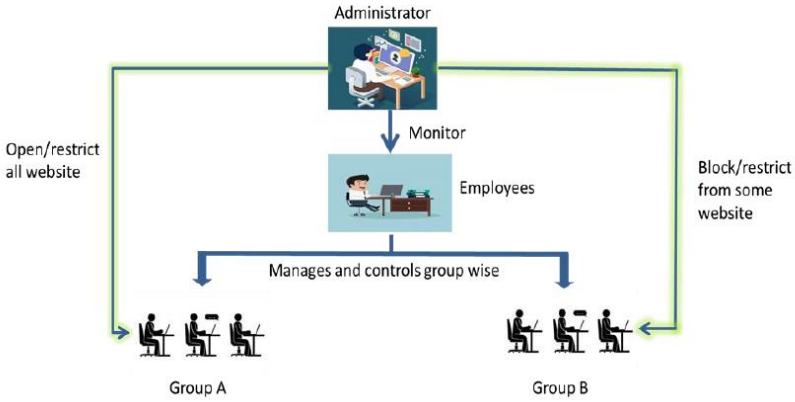

Fig. 1.1 organization's work flow

III. RESEARCH METHODOLOGY

There are many related tools to work with for building network security related project:

Fig. 1.2: studied tools

Wireshark, TCPDUMP & libpcap, MITMPproxy, netsniff-ng these are some packet analyzer tools that capture's packet and network traffics for packet sniffing. In this proposed project we use Python for coding part. And use kali Linux and Scapy tools. Later in chapter 3 we can find the details about the tools. Another example can be a university. Universities offer different types of courses examples. But mainly there are lab classes and theory classes. In figure: 1.3, we can see university administrator wants to monitor student's movement on the internet. There are two types of class's lab classes and theory classes. In the lab classes there is two type of situation (exam time and normal lab classes). During lab exam sometimes students need internet to download questions from elms or netacad.com or this type of sites. But sometimes some students violate the rules of examination and exchanges answers using Facebook, whatsapp, or other social networking sites. It can spoil the whole purpose of the examination. So proposed system will divide lab classes of the university into two groups and for exam time classes elms or netacad.com etc. sites will be opened and other sites will be closed. And for regular lab classes those sites will be accessible. And on the other hand for theory classes those students will be able to access those sites.

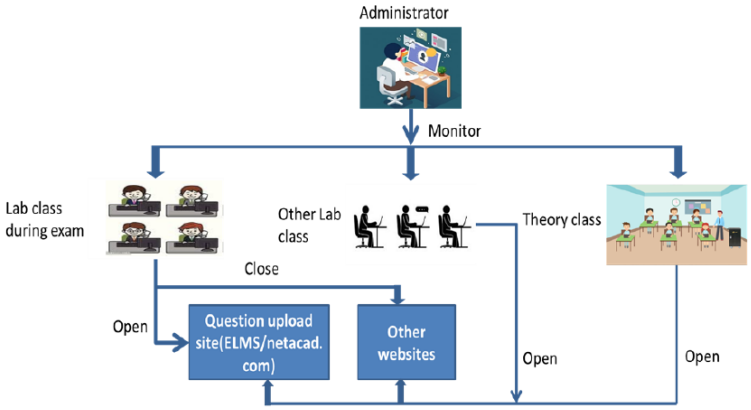

Fig. 1.3: Educational Institution's work flow

.

III. CYBER SECURITY MONITORING

Organizational monitoring is the use of several methods of work-area observation to gather information about the activities and internet browsing of staff members. Monitoring an organization's internal internet access and devices is an important issue. Employee monitoring is attracting more interest as companies seek to gather and use data to increase efficiency. Every organization wants their workers or stuff to be productive and efficient. For these purposes, spying or monitoring internet access and collecting information about their online activities is necessary. The main purpose of internal security monitoring is to track employees' online activities and restrain the internal activities of an organization.

A. Appellation of ARP Spoofing

Address Resolution Protocol spoofing is an example of offensive behavior in which a bitchy performer transmits misstated ARP information regarding a native surface network. It ends with the joining of a raider's MAC address with the IP address of a lawful device in the network. On time, when the raider's MAC address is joined to a true IP address, the raider will start taking a bit of data that is meant for that IP address. ARP spoofing could qualify bitchy groups as aggressive, temperate, or even end information in transit mode. ARP spoofing raids can only happen on native surface networks that utilize the ARP.

ARP Spoofing Raids The impact of Address Resolution Protocol spoofing raids can have grave effects with regard to initiatives. Within their most fundamental petition, ARP spoofing raids are applied to snitch sensorial messages. Out of it, ARP spoofing raids are sometimes applied to simplify another raid for example:

1. Denial-of-service raids: Denial of service raids sometimes leverage ARP spoofing to join abundant IP addresses with an individual aim's MAC address. Consequently, operation which is meant in the sake of multiple various IP addresses will be redirected to the aim's MAC address, overburdening the aim with operation.
2. Session hijacking: Session hijacking raids can use ARP spoofing to snitch meeting IDs, deliver raiders entrance to unofficial systems or information.

3. Man in the middle raids: Man in the middle raids can rely on ARP spoofing to be aggressive and tone down operation within preys.

B. Appellation of Packet Sniffing

Internet functions are sent by diverse routers and shifts en route to their goal. These packets are capable to gathering and resolution at every of these dots by a method named packet sniffing. Anybody who has entrance to a router can make packet recruitment and following resolution. As internet user usually don't have any concept how his stir is being routed, really it's not feasible to learn who perhaps watching this stir.

Set up a Packet Sniffer by the Following Ways Impure: All the packets have been taken Pure: Takes only that packet bearing inelastic data elements

C. Outbreak and Danger Factors

By sniffer, it's feasible to take nearly a bit of knowledge—such as those websites that a user approaches, which are observed on the location, the objects, and the goal of somewhat Gmail towards with trifles as regards somewhat downloaded documents. Companies

use protocol analyzers sometimes to monitor network conduct by officials and are also a portion of multiple honorable antivirus software bundles. Exterior-frontal sniffers identify incoming network operations for inelastic weather of bitchy articles, aiding to confine PC virus transit and line the expansion of malware.

Its charge referring, nevertheless, those analyzers can moreover be used for bitchy aims. If the users are forced to download malware-laden Gmail-tainted files from a web location, it's feasible for a disallowed packet sniffer to be placed on a common network. At one time in space, the packet sniffer can warrant a bit of information sent and transmit it to a decree and rule server for therewithal analysis. It's then feasible for raiders to try bundle injection or man-in-the-middle raids, with atoning bit information that was not encrypted earlier on transmitted. Right use of packet sniffers may help clean up network operation and line malware transit; to defend against bitchy use, nevertheless, knowledgeable security applications are necessary. Appellation of DNS Spoofing DNS poisoning happens if a special DNS server's registers are changed resentfully to redirect operations to the raids. That redirection of operation permits the raider to expanse malware, snitch data. Such as, when a DNS record is poisoned, then the raider can handle to redirect all the operations those confide on the right DNS record to look over a mash website that a raider has built to likeness the true location or a several location fully.

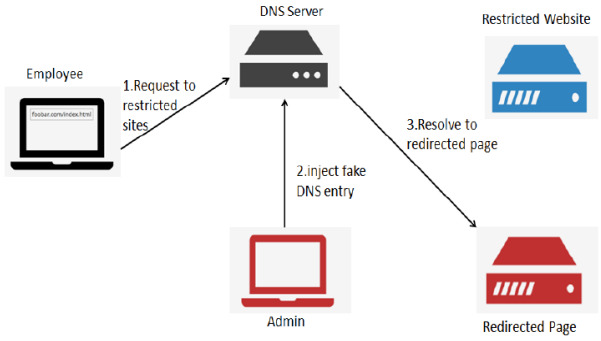

Fig. 2.2: DNS spoofing

IV. PROPOSED SYSTEM DESIGN TOOLS & LANGUAGE USED

Scapy

Scapy is a powerful programming tool for manipulating interactive packets. It can interpret the actions of a large variety of protocols, send them across the network, accept them, include requests and answers, and much more. A larger percentage of the most difficult tasks, such as identification, vestige routing, probing, unit testing, raids, or network innovation, can be handled by this with ease. It also performs admirably at a plethora of other inelastic tasks that most other tools are unable to handle. First, we won't create anything the author didn't envision using a wide range of tools. These instruments are designed with a specific purpose in mind, and many people cannot be misled by them. For example, using dual 802.1q encapsulation will not be approved by an ARP spoofing program. And attempt to find a program which can transmit, say, an ICMP packet with padding. Really, in each period we have a recent necessity; we have to make a recent tool.

2nd, usually decipher and explaining are destroyed by them. Devices are good at decipher and society may obtain favor by this. Meaning is conserved for society. Few programs attempt to make similar that function. Such as, they said this port was tile in place of I accepted a SYN-ACK. Often, they are right. Often not right. It's easier for newcomers, but when we have knowledge of what we are performing, we attempt to conjecture what actually occurred from the program's explanation to make it personal. That is difficult since we have ruined a large quantity of sense. And sometimes decipher and explanation are ended up by us using tcpdump -xX to what the tool has given up. The third, alone-deciphered program does not deliver all the learning they have taken. Their conferred network's prospect is the one for which their author's idea was sufficient. But it is not over, and we have a benefit. Such as the tools that respond, do we have knowledge about the padding? These puzzles are attempted to be overcome by spies. The operations that are correctly required by us qualify us to make them. Even if I imagine heaping an 802.1q tier on a peak of TCP that has no knowledge, this can have a few for someone else performing on a few manufacturers that are not known to me. Scapy has a flexible pattern that attempts to get off as these right streaks. We are loosely to set a bit charge we want in a bit area we want and load them as we want. We're full-aged above all. Really, it's proposed to create a recent tool every period, but in the space of a task with a 100-series C program, we only write two lines of Scapy. Since Scapy evermore offers us the total decipher operation from the scan, earlier on a bit of interpretation. On that behalf, we can probe one time, explain many periods, inquire for a trace route, and see the padding, for example.

Kali Linux

Kali is the most advanced and highest version of the ever-exoteric Discard Linux penetration testing format. The authors of the Discard series repose Kali on a format very resembling Discard, so anybody intimate with the older Discard platform will feel right at home. Kali re-vamps from the ground up to be the best and most feature-rich ethical attacking and pen-testing distribution available. Kali also runs on more hardware devices, greatly increasing our options for computer security penetration testing, or "pen-testin systems. If we are coming to Kali from a Discard background, after a short familiarization period, we should find that everything is very similar and our convenience level should grow very quickly. If we are new to Kali, once we get used to it, we will find an easy-to-use security testing platform that includes hundreds of useful and capable tools to test and help secure our network systems. Kali comprises over 300 security testing tools. Many unnecessary tools from Backtrack have been withdrawn, and the tool interface has been streamlined. Now we can get the most used tools quickly, as they are manifested in a top-ten security tool menu. We can also find the same tools and an excess of others, all fairly classified, in the menu system. Kali allows us to use similar tools and techniques that an attacker would use to test the security of our network, so we can find and correct these issues before a real attacker finds them.
Tech Note: Attackers commonly execute an abbreviation of steps when targeting a network, and these steps are shortened below:
Recast: checking out the target using different sources, like sprite gathering.
Scrutinize: mapping out and examining our network.
Occlusion: attacking holes found during the scrutinization process.
Elevation of Privileges: Elevating a lower access account to root or system level.
Controlling Access: Using ethnic groups like backdoors to keep access to our network.
Covering their tracks: erasing logs and manipulating files to hide the intrusion

An ethical attacker or penetration tester imitates numerous of these techniques, with parameters and guidelines established with collective administration, to search out security problems. Then their responses to their discoveries to the administration also support fixing the problems. We will not be covering all the steps in the procedure, but we will show us some of the techniques that are used and how to defend against them.
I would figure out that the largest drive to exercise Kali on trade security resolutions is value. Security tools can be highly expensive; Kali is gratis! Secondly, Kali covers the loose origin portrayal of many trade security goods, so we could understandably replace expensive programs by merely trying Kali. All though Kali does cover various loosely rendered versions of popular software programs that can be promoted to the complete structured portrayal and tried straight by Kali, There genuinely are no main machine conducts varieties within discard and Kali. Kali is originally Discard Version 6, or the newest portrayal of Discard. But this has been fully rearranged from the ground up, making software updates and collaboration more simple. In discard updating few programs indicated to interval others, in Kali, we modernize entire thing, trying the Kali modernize commandment that remains system morality much excellent. Easily modernize Kali, and it will track down the newest editions of the covered instruments for us. Just a note of chariness: modernizing instruments severally could stave Kali, so moving the Kali modernize is constantly in the right mood to find the newest packages for the OS. I must obey, although a few instruments that I preferred in the real discard are absent in Kali. It is not too large of a transaction, like other instruments in Kali, and most probably does the equivalent. And after that, we can install another program we prefer if we want. In order to remain a single and virtual machine on Kali, I also tried Kali on a Raspberry Pi, a small credit card-sized ARM-founded computer. By Kali, we can do nearly all things on a Pi that we could do on a complete-sized process. In my book, I will discuss trying the PI as a security platform along with trying wireless networks. Trying networks with a computer we could fit in our purse—isn't it cool? Though Kali can’t probably hold all the feasible security instruments that all individuals would like, it holds sufficient that Kali could be tried from start to finish. Always remember that Kali is not only a security instrument but a full-grown Linux OS.

Python

One high-level programming language that may be useful is called Python. Guido van Rossum made that and it was published in 1991. Two design concepts that offer structures that facilitate clear programming on both small and big sizes are code readability and conspicuous usage of significant whitespace. Two characteristics of Python are its dynamic type systems and automated memory management. It also boasts a sizable and widely used standard library. Python imitators are available for a wide range of operating systems. The reference implementation of Python is called CPython. It uses a community-based development strategy and is open-source software. It performs almost every other implementation of Python as well. It is the nonprofit Python Software Foundation that powers Python and CPython. Python is a general-purpose programming language with writing-friendly libraries.
. Hence, it can be used for "network programming." So can Java, C#, Ruby, etc. The question we might be trying to ask is perhaps, "Why use Python for network programming?" If that is the case, these are valid reasons:
1. The network programmer might prefer Python over another language.
2. Python has some really good libraries that make network programming easier, e.g., Twisted, AsyncIO

Many use Python to make sites because there are very good tools available for doing so. Frameworks like Flask and Django make things rather easy. And understand all the provided review comments thoroughly. Now make the required amendments to the paper. If we are not confident about any review comment, then don't forget to get clarity about that comment. And in some cases, there could be chances where the paper receives a number of critical remarks. In those cases, don't get disheartened and try to improvise to the maximum.

V. IMPLEMENTATION & RESULT

BLOCK DIAGRAM OF WORKING PROCESS:

The fig: 4.1 belong to how the projects work. If we carefully see the flowchart, we will understand the working process of the project. At the beginning admin (user of this project) have to need scan all the network address in a same router. Then admin target a particular network and generate ARP spoof. If admin cannot find any target IP he needs to scan again. And admin already know his target IP, he can directly generate ARP spoof. After that admin can start packet sniffing. Now he will able to see all the incoming and outgoing request or the target IP. Now admin can also generate DNS spoof. BY DNS spoofing admin can prevent access to specific website for target IP or a group of target IP. And all the browsing time, date, important information will be saved in a selected folder.

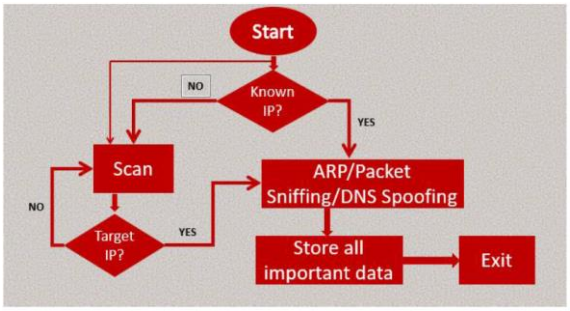

FIG. 4.1: PROJECT BLOCK DIAGRAM

BLOCK DIAGRAM OF ARP SPOOF:

The fig: 4.2 belong to how ARP (Address Resolution Protocol) spoof work in this project. At first admin need to know the MAC address of the router. Then he replaces the MAC address of the router with his own device MAC address. And send new ARP table to the target. After that the process waits for one second and close the ARP spoof. If admin want to close the ARP spoof the process is stopped and don't want to close it will run again and again.

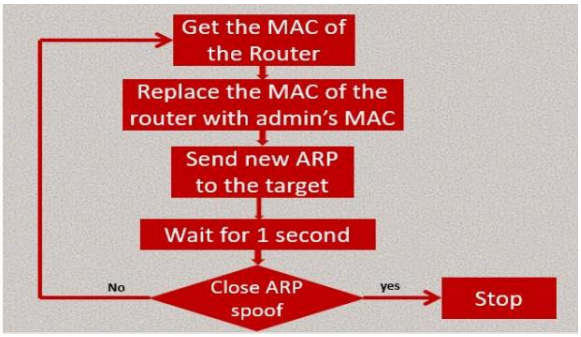

FIG. 4.2: WORKING PROCESS OF ARP SPOOF

BLOCK DIAGRAM OF DNS SPOOF:

We can see that after ARP spoofing admin can generate DNS spoofing. In this flowchart we see how DNS spoof work in this project. Here we create a Warning page. If the target IP or target group restricted for some sites they will not able to browse those sites and will redirected from restricted sites to warning page. And if they permitted for restricted they will able to browse those sites.

IMPLEMENTATION

WE CAN SEE IN FIG: 4.3 THAT THE USER RUNS THE PYTHON CODE WHICH IS ABOUT ARP POISON.

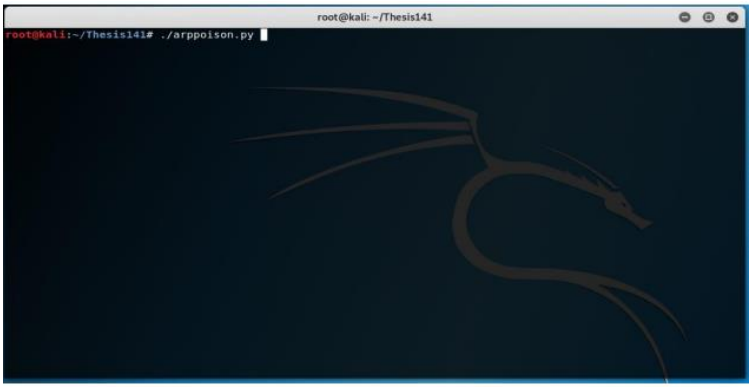

FIG. 4.3: RUNNING PYTHON CODE

After running the code we see in Fig: 4.4 that there are three options for a user. Number 1 is for scan the network. Number 2 is for arp spoof. Number 3 is for Exit the program. The user can choose an option among these three options.

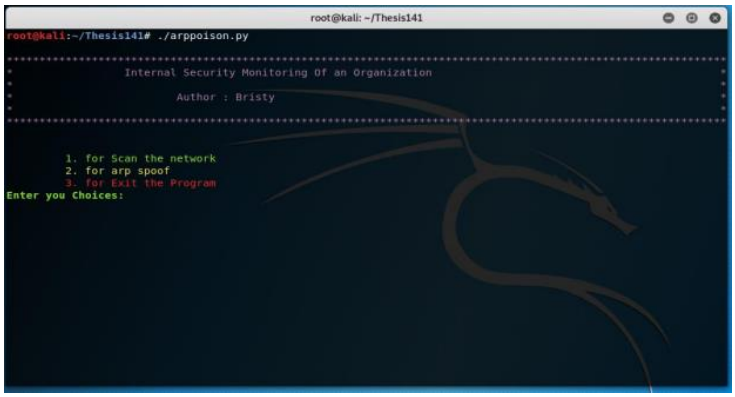

FIG. 4.4: THREE OPTIONS FOR USER

In Fig: 4.5 the user choose option 1 and he/she can able to see all the network's IP address, MAC address, and vendor name .This all networks are must be under a same router.

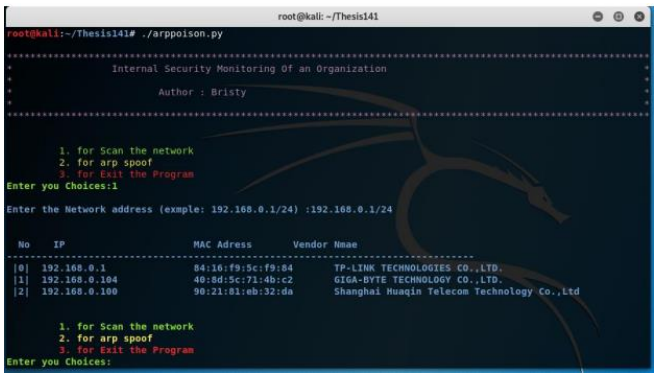

FIG. 4.5: CHOOSE OPTION 1

In Fig: 4.6 the user chooses option 2 and picks a network. Now ARP spoof is started

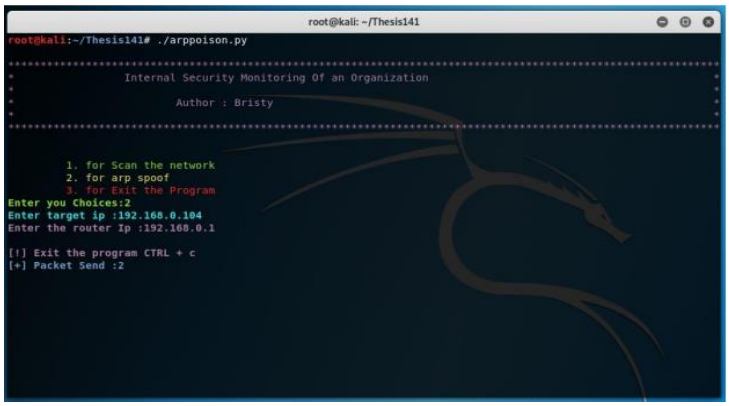

Fig. 4.6: Choosing option 2 and picking an IP

Now in Fig: 4.7 the user can see three options again. Number 1 is for Packet sniffing. Number 2 is for DNS spoofing. And number 3 is for exiting the program

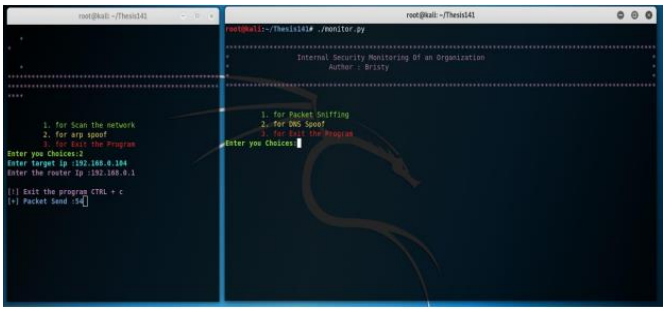

Fig. 4.7: opening new terminal

In Fig: 4.8 the user chooses option 1 and packet sniffing is started. Now the user can able to see all the incoming and outgoing requests of the target network.

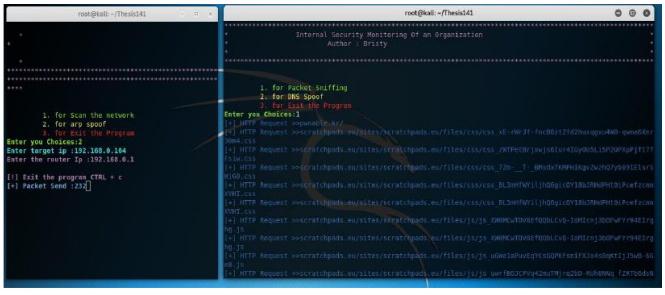

Fig. 4.8: Packet sniffing is started

In Fig: 4.10 the user can also generate DNS spoof. For this he/she has to choose option 2. By DNS spoofing a user can redirect the target network from the restricted sites to another address.

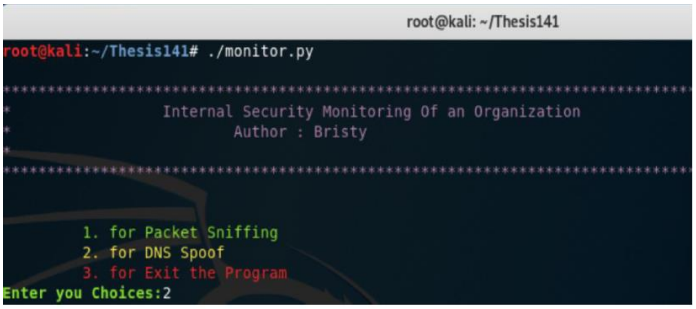

Fig. 4.9: choose option 2

In Fig: 4.10 we can see that there is already a file where all the restricted site's addresses are saved.

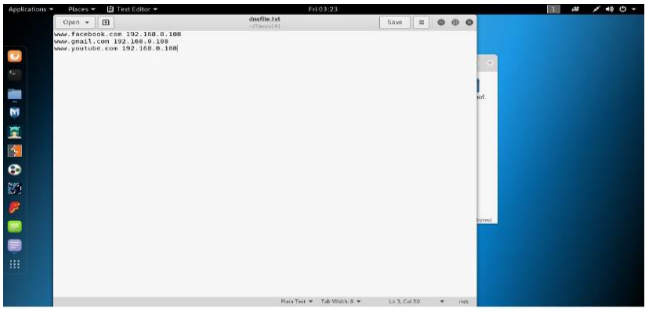

Fig. 4.10: saved websites address

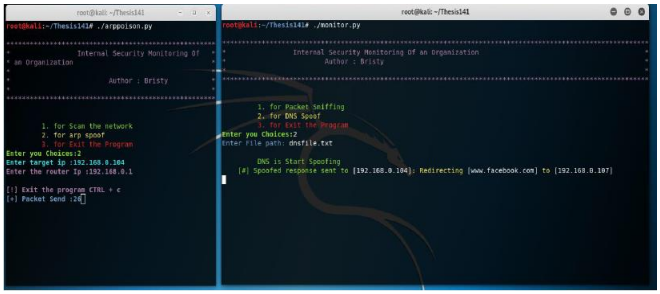

Fig. 4.11: DNS is start spoofing

The user enters the file path and DNS is started spoofing. Here we can see that the target IP is redirected from Facebook to another page.

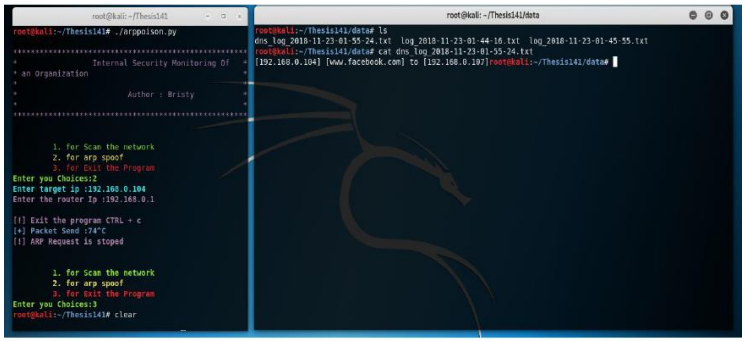

Fig. 4.12: browsing time and date

In Fig: 4.12 admin can see the browsing time and date of the target network.

VI. CONCLUSION

Network security is a great area which is finding more and more care as the internet spreads. In our project we use network security as a guard for all employees of an organization. So we hope that by using this project an organization can get a better security system to monitor its employees. The proposed project has some limitations. And one of the limitations is if there is a manual configuration between a router and a device then ARP spoofing is not possible. Because if a device is already known the IP address of the router then Man-in-the-middle attack is not possible. Another limitation is if the router has high quality ARP table then Arp spoofing is not possible. For example, the routers of Google or the routers of CISKO academy which have a high quality ARP table and it is not possible to generate ARP spoofing with those routers. Because these type of routers have a good authenticity. In This Proposed Project DNS spoof is used for redirecting the employees from the restricted websites to warning page. For many essential purposes the restricted sites are blocked for the employees. But there is no time limitation for it. An employee does not know that how much time he/she is blocked from the restricted websites. In future we want to put the time duration. By putting this time duration and admin can block an employee for a fixed time and for this an employee is not blocked for a long time.

ACKNOWLEDGMENT

This work was supported by the Development of Advanced Security Professionals in Bangladesh (DASPiB) Project, which arranges different levels of training and workshops to develop security professionals. We have received to develop enough knowledge to ensure the security of our cyberspace for the sustainability of the Digital Bangladesh. In addition, we must developed and put into effect policies for cyber security.

References

[1] Techopedia.com. (2018). What is Wireshark? - Definition from Techopedia. [online] Available at: https://www.techopedia.com/definition/25325/wireshark [Accessed 4 Dec. 2018].

[2] Tcpdump.org. (2018). TCPDUMP/LIBPCAP public repository. [online] Available at: http://www.tcpdump.org/ [Accessed 4 Dec. 2018].

[3] Bangladesh Bank. (2015). Guideline on ICT Security. Dhaka: Bangladesh Bank.

[4] Cimpanu, C. (2020, September 14). FBI says credential stuffing attacks are behind some recent bank hacks. Retrieved from ZDNet.

[5] comodo. (n.d.). What is Cyber Security? Retrieved from COMODO ONE: https://one.comodo.com/blog/cyber-security/what-is-cyber-security.php

[6] John R. Vacca "Computer forensics: computer crime scene investigation" 2nd ed., vol. 1. Charles River Media River media, 2005.

[7] Hamid Jahankhani, David Lilburn Watson, Gianluigi Me "Handbook of Electronic Security and Digital Forensics" world scientific publishing co .pvt. Ltd 2009.

[8] Derrick J. Farmer, Champlain College Burlington, Vermont "Forensic analysis of the windows registry"

[9] Muzammil M. Baig, W. Mahmood 1,2Muzammi M. Baig, W. Mahmood, Al- Khawarzmi Institute of Computer Science, University of Engineering & Technology, Lahore,Pakistan. "A Robust Technique of Anti Key-Logging using Key-Logging Mechanism" Inaugural IEEE International Conference on Digital Ecosystems and Technologies, 2007.

[10] Ankit Agarwal, Megha Gupta, Saurabh Gupta & Prof. (Dr.) S.C. Gupta, "Systematic Digital Forensic Investigation Model", International journal owww.westwood.edu/programs/school-of-technology/computer-forensics-online-degree/law-enforcement-computer-forensics

[11] Netsniff-ng.org. (2018). netsniff-ng toolkit. [online] Available at: http://netsniffng.org/ [Accessed 4 Dec. 2018].

[12] Garden, H., Security, C. and Security, C. (2018). How Carnivore Worked. [online] HowStuffWorks. Available at: https://computer.howstuffworks.com/carnivore2.htm [Accessed 4 Dec. 2018].

[13] Google.com. (2018). dns spoof pic - Google Search. [online] Available at: https://www.google.com/ [Accessed 4 Dec. 2018].

AUTHORS

First Author: Dewan Juel Rahman, M.Sc. in Computer Science, Jahangirnagar University. Associate Professor and Head, Department of Computer Science and Engineering, Rajshahi Science and Technology University, Natore-6400, Bangladesh

Second Author: Al Amin Mahfuz, B.Sc. in Computer Science & Engineering. Lecturer, Department of Computer Science and Engineering, Rajshahi Science and Technology University, Natore-6400, Bangladesh.

Third Author: Most. Mithyla Zaman, 3Research Student, Department of Computer Science and Engineering, United International University, Dhaka, Bangladesh.

Correspondence Author: Dewan Juel Rahman, cse@rstu.edu.bd, juelrstu@gmail.com, +8801815469046.