Denial of Services (DoS) Attack: Implementation in Wireless LAN and Countermeasures

Salman Naseera[a], Mudasar Ghafoorb[b], Sohaib bin Khalid Alvi[a], Hafiz Shaheer ul Islamc[c]

[a]Department of Information Technology, University of the Punjab Gujranwala Campus, Gujranwala Pakistan.
[b]Department of Administrative Sciences, University of the Punjab Jehlum Campus, Jehlum, Pakistan.
[c]University of Sialkot, Sialkot, Pakistan.
Corresponding address: salman@pugc.edu.pk

Abstract

There are several different networking attacks but a denial of service attack (DOS) is the one that disables a server from servicing its clients. In this area, some productive hackers and professionals have built packages, methods, and scripts that trigger DoS attacks on different network types. The key phenomenon of the DoS attack is that in an effort to knock it down, it produces many false packets to the network, which implies overflowing of a large number of invalid data packets that are sent to a specific server with a wrong IP address. We will illustrate the implementation and evaluation of DoS attacks on a virtual server (in a local network) using a powerful tool named 'Xerxes'. In addition, we analyzed the attack performance using the Wireshark tool to analyze the IP address of the attacking source. The experimental results infer that Xerxes outperforms the target machine to service its clients. In this paper will show the possible damage caused by DoS attacks and examine the effects of the damage and in last, we will discuss the ways to stay safe from this kind of attack in networking infrastructure.

Keywords: DoS attack Implementation, Denial of Service Attack, Wireless LAN, and Xerxes

Introduction

A Denial of Services (DoS) attack is a serious web server and website threat, which are using weak protocols like HTTP instead of using HTTPS (Sharma & Mittal, 2019). In the networking area, this attack has a great deal of attention as it can lead to substantial economic losses if attackers shut the site down for a significant period. Denial of service (DoS) is a form of threat in which intruders steal the

capacity of approved users by not allowing them access to any service given by either flooding or collapsing the data centers using any DoS attack. There are several different forms of DoS attacks; however, the two most common attacks flood the target with tons of invalid packets with Ping of Death and Syn Packets Flood(Mohammed & Issac, 2007). We are going to experiment with the second type of attack, which is flooding the target machine with heavy amounts of packets in order to block it by not giving services to its genuine users (Kissi & Asante, 2020),(Ohira, Desta, Arai, Inoue, & Fujikawa, 2020).

The DoS attack is of two types i.e. solitary or multi; it depends on the number of connections that are being used. Only the machine resources of a single system or a single Internet connection can be overflowed by a solitary attack, and on the other hand, an intra Distributed DoS (DDoS) attack tries to use a sufficient count of parallel computers and Internet connections to overflow a source, which makes it a serious, extensive threat (Bouyeddou, Kadri, Harrou, & Sun, 2020).

In TCP SYN Attack (DoS), if users have to communicate through the TCP transport protocol, a connection consisting of the three-way handshake must be set up:

- The host began the connection by transmitting a TCP sync packet to the recipient destination.
- The target system sends the response the originating host an acknowledging packet.
- The host will transmit a Data packets with an acknowledging message to the target host (Sandhu, Haider, Naseer, & Ateeb, 2011), (Dong, Abbas, & Jain, 2019).

The three way agreement is finished and regular exchange of data will commence. The host simply fails to complete Phase iii throughout a DoS attack, leaving the target host with an ongoing interaction link. As the TCP port of the target receives the message in phase one, in preparation for the contact to follow, it allocates buffers and enhances communication stacks. By sending a "flood" of packets and never bothering to see any response, the intruder is able to overwhelm the computational capabilities of the target (Dong et al., 2019), (Idhammad, Afdel, & Belouch, 2018).

Literature Review

The DoS attacks have extended in consistency, complexity, and the pace at which they occur in the last few years. Due to extent of threats on large networks, the Internet needs urgent attention and effective countermeasures for DoS attacks (Naseer et al., 2012). In 2018, the biggest DoS attack on GitHub was released, which overwhelmed Internet traffic at a rate of 1.33 TBPS (Bouyeddou et al., 2020), which has shown in Fox News network. The other biggest attack was made in late 2016 on a cloud provider in France ‘Mirai botnet’ (an army of compromised systems) with 1.1 terabits

per second. Jelena and Peter, in their work, Shared features and significant characteristics of the attack scheme have been selected as attacking factors (Faouri et al., 2022), (Naseer, Liu, & Sarkar, 2019). It can also be used for problem identification and specific design of countermeasures. Based on their system architectures, the security terminology categorizes the form of current DoS defenses. The work discusses how these observations illustrate the benefits and drawbacks of the proposed arrangements. Feinstein et al. suggested strategies to identify their Availability of DoS attacks by comparing the distribution of abundance and intensity of packet traits selected (Naseer et al., 2021). To demonstrate anomalies in the characteristics of the packet attributes selected (Ahmad, Ijaz, et al., 2022), DoS attacks have been registered. For intense DoS detection, a matrix analysis approach is used by analyzing the threat strategy structure containing the threat type, selection of controllers, staff, communications, and manipulated mechanism.

Kulkarni and Bush proposed a new active router methodology that implements intruder resources to associate random flow of traffic. On the network to monitor DoS attacks. The whole methodology does have the benefit is that it does not use any unique detection rules and is therefore ideal for detection of any generic DoS threat (Agrawal & Tapaswi, 2017). Michael et al. has suggested an integrated design to explain the nature of different parts of the system in the DoS attack (Zaman-ul-Haq et al., 2022). The approach captures the effectiveness, gain, and possibility of attacks on modules. Active attacks will reflect the magnitude of concurrent threats mostly on targeted device's responsiveness. Nagy et al. have achieved a robust DoS attack tracker using FPGA (Naseer et al., 2018). The tracker is said to be able to identify most types of DoS attacks in moments (Bokhari et al., 2022).

As reported above, the DoS security mechanisms are based on the attacker's conduct and characteristics. The data and evidence of the intruder's patterns and processes will also contribute significantly to the effective development of the protective mechanism. The purpose of this paper is to give an analysis of the entire implementation of the attacking mechanism with attacking evidence of sending packet floods on the Wireshark tool, and finally, we have suggested some potential solutions to resolve this type of network attack (Kaur et al., 2022).

Experimentation And Results Of Dos Attack Implementation Using Xerxes: Dos Attacking Tool

Xerxes is a basic denial of service (DoS) application-layer mechanism that is used directly to attack servers and can be launched from a single device. It is not based on a botnet and all connections come from a single source. The tool initiates a TCP link flood to its target after execution, causing session table resource exhaustion, effectively crashing the server (Riaz et al., 2022), (Ishfaq et al., 2022). Xerxes can take down a web server without the need to create a large amount of traffic, enabling network-based defenses to go unnoticed.

3.1 Xerxes (an Attacking Tool):

It is one of the latest DoS attacking tools developed by a hacker named (CyberXCodder) in year 2019. This attack, which uses many servers, is not a typical DoS attack. It compromises 'http' by sending fake packets, which takes it down in some milliseconds. It can keel over the http server by dispatching a couple of hundred of them every second. The whole attack hits, rather than choking the channels to the database, by flooding the web server. Hence, the total bandwidth a link of a server has; is negligible. If a new server is just installed, it only increases the attack intensity by n+1 to take it down (Ali et al., 2022), (Ahmad, Cherif, et al., 2022).

A. Configure Xerxes in Kali Linux

Fire up your Kali Linux machine now, and clone or download Xerxes to your Desktop. The recommended place for Xerxes is the desktop. Open your terminal and, as it is easy to find out, go to the Desktop directory.

Figure 1: Desktop directory

Therefore, it will not take that long to download the Xerxes Directory from GitHub,
#git clone https://github.com/CyberXCodder/XerXes.git

Figure 2: Clone the Xerxes repository from GitHub

Now let's go to the directory of Xerxes and see what stuff we've got.
#cd Xerxes
#ls

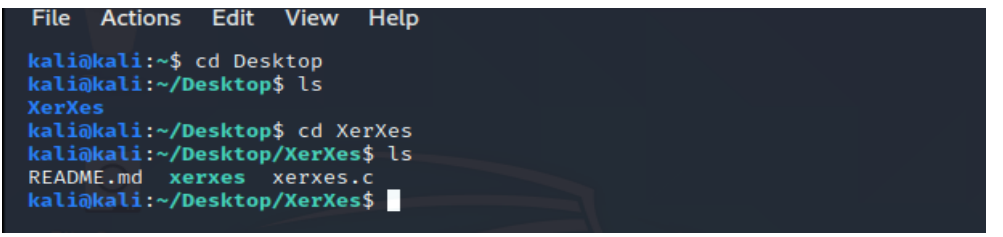

Figure 3: Get into the Xerxes folder

As in image shown, you can see that it includes several files, one of which is 'Readme' and the other is the script 'xerxes.c.' The Xerxes script extension tells us that it is written in the C language, and we have to compile it before we use it. Now, step forward and use the following command to compile it.
#gcc xerxes.c -o xerxes

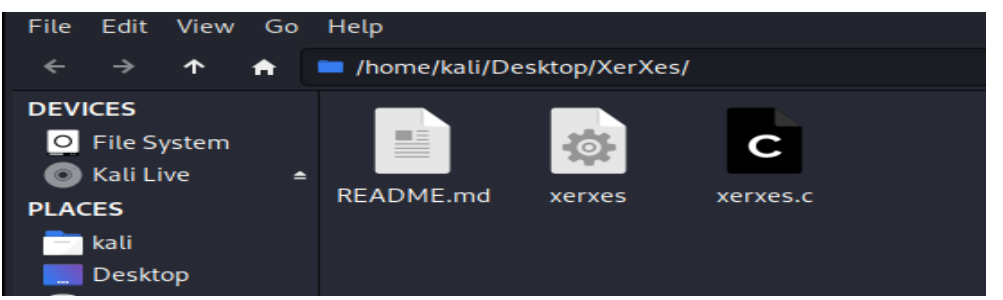

Figure 4: Inside files of XerXes

A. Targeted Virtual Machine

Here you can question yourselves which target we're going to try and strike, I recommend you to perform it with your own environment where you regulate

everything as you attempt to study these things, and one of the effective methods is to use this on your virtual environment or a Network that you want to test, now I have my Metasploitable2 server in my case.

B. Attacking the Target

Firstly, let us notice the Address of our victim machine, so the IP of my metasploitable2 server can be seen here,

Figure 5: Find the IP-address of our target server

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:0e:63:fb
          inet addr:192.168.32.129  Bcast:192.168.32.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe0e:63fb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:37 errors:0 dropped:0 overruns:0 frame:0
          TX packets:71 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4656 (4.5 KB)  TX bytes:7302 (7.1 KB)
          Interrupt:19 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:97 errors:0 dropped:0 overruns:0 frame:0
          TX packets:97 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:21529 (21.0 KB)  TX bytes:21529 (21.0 KB)

msfadmin@metasploitable:~$
```

Currently we know that metasplotiable2 is a server, let's attempt to access it just here, launch the kali window, and type metasplotiable2's Destination ip.

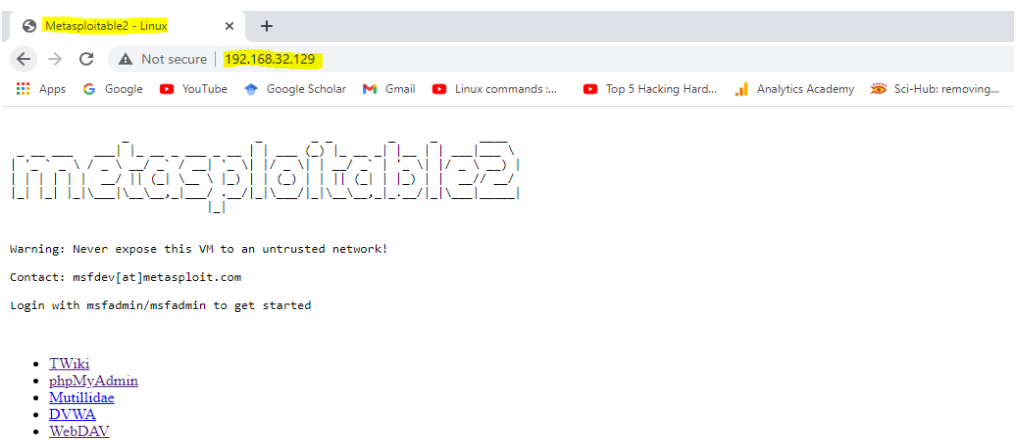

Figure 6: Launch your Kali window and enter in the IP address of metasplotiable2

Type this command in your console now, and we need to define the target and port.
#. /xerxes [Target_IP] [Port]
#. /xerxes 192.168.32.129 80

```
kali@kali:~/Desktop/XerXes$ gcc xerxes.c -o xerxes
kali@kali:~/Desktop/XerXes$ ./xerxes 192.168.32.129 80
```

Figure 7: Attacking the Target

The volley (false packets) is going to be sent. Now let us see if we can down the server.

```
kali@kali:~/Desktop/XerXes$ gcc xerxes.c -o xerxes
kali@kali:~/Desktop/XerXes$ ./xerxes 192.168.32.129 80
[Connected → 192.168.32.129:80]
[0: Voly Sent]
[Connected → 192.168.32.129:80]
[0: Voly Sent]
[Connected → 192.168.32.129:80]
[0: Voly Sent]
[Connected → 192.168.32.129:80]
[0: Voly Sent]
```

Figure 8 Sending false packets to Victim

All right, Xerxes started sending botnets, and if we refresh the webserver of Metasploitable 2, Xerxes seems to have taken it down.

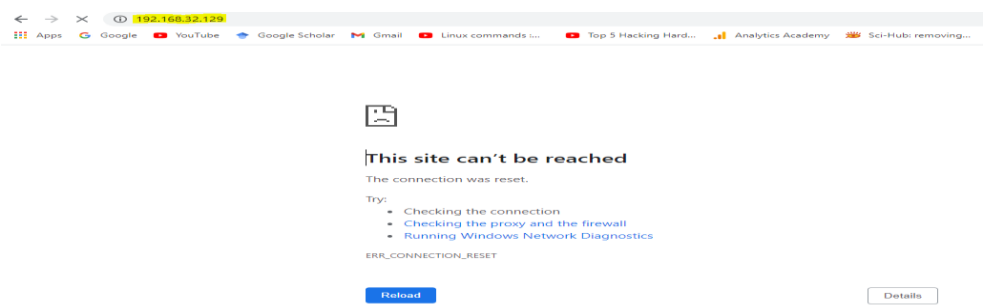

Figure 9 Server is down/attacked

C. Analyze DoS attack packet flood Using Wireshark Tool

Wireshark is traffic-monitoring tool, which is, monitors all the incoming and outgoing traffic in any network. If you want to examine something unique, such as

the traffic a program sends while calling home, it helps to shut down all other network applications so that the traffic can be minimized. Nevertheless, you will probably have a huge number of packets to pass through. That is where the filters from Wireshark come in (Ahmad, Manzoor, Naseer, Ghaffar, & Hussein, 2021). By typing it into the filter box at the top of the window and clicking Apply, the most basic way to apply a filter is (or pressing Enter). For instance, type 'TCP.PORT=80' and you will only see TCP packets. When you begin typing, Wireshark will help you auto-complete your filter.

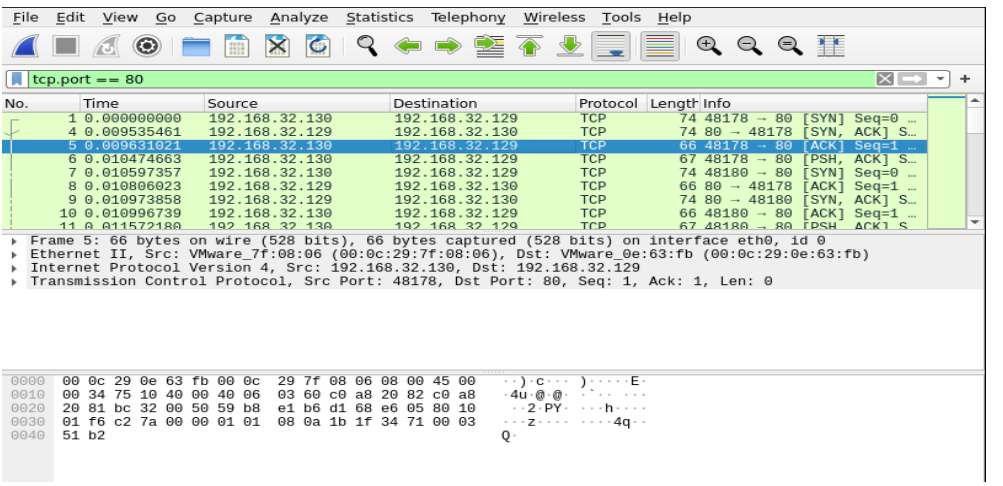

Figure 10 Analyze DoS attack packet flood Using Wireshark Tool

4. Discussion and Analysis

Network safety is a serious problem for customers in the technological modern world. The network helps users mainly to connect and obtain services conveniently. The internet needs all consumers to serve as a universal source of information, so the internet's usefulness is necessary (Abrar et al., 2021). Due to the obvious benefits, it provides towards customers, keeping them unavailable is the primary objective of attacks. At first, when a device or the whole network connection is completely sabotaged after an attacker exploits vital resources like mails, web sites, and other online communication resources, users are stopped from getting access (Benkirane et al.). Hackers will raise the DoS level of the attack by (DDoS) by executing these exploits in a parallel environment called a Distributed Denial of Service (DDoS) attack (Cetinkaya, Ishii, & Hayakawa, 2019). In this attack, a group of computer systems transmits organized strikes against a single target. The sender generates a unique type of attack where most UDP and TCP frames are being sent with varying volume bytes to the targeted device at that time

(Elleithy, Blagovic, Cheng, & Sideleau, 2005). Therefore, the software effectively monitors and collects network traffic sent by an intruder. The listed protocols flooding exploits that are typically stronger in destroying victim's resources; as they consume most channel capacity on the victim's network channel by not giving access to approved users (Zhang, Shen, Zhou, Dong, & Yu, 2020), (Tomar, Jeena, Mishra, & Bisht, 2020).

5. Suggested Countermeasures

- By implementing the appropriate countermeasures in the correct spots, the effectiveness of the network protection of an enterprise can be improved. For DoS attacks, several such countermeasures are available.
- To detect any anomalous activity, ensure that the software and protocols used are up-to-date and search the devices thoroughly.
- Improved routing protocols, particularly for the WMN multi-hop, are desirable.
- Configure the Port forwarding and open the ports only at the time of communication.
- Disable resources that are unused and vulnerable.
- To block the traffic from the reflection servers, block all inbound packets originating from the service ports.
- Preventing the distribution at the ISP stage of fraudulently addressed packets.

To manage the jamming and scrambling type of threats, deploy cognitive radios in the physical layer. Configure the firewall to prevent traffic access to the external Internet Control Message Protocol (ICMP). You can also avoid this attack by enabling cloud fare service on your website/Machine

6. Conclusion and Future Work

Internet service is essential because it serves all users with worldwide sources of information, and the internet has become the target of attackers because of these advantages. One of the main problems is recognition of the DoS attack in the client network topology. We performed DoS attacks using the Xerxes tool in this analysis, by overflowing the target system with inappropriate traffic. Xerxes provides excellent performance and is very easy to use and install. For a student, this is a positive skill to understand these sorts of things. This is because there are still some websites that do not have DOS security nowadays. Xerxes can do significant damage to them. However, it is not enough for larger websites to carry

out an attack from a single device to put them down as they can accommodate a large amount of traffic or have higher bandwidth. Nevertheless, this tool may be a serious issue for other small websites. We are planning to generalize this research into three fields as part of future work. Next, we intend to choose a broader range of products, like routers. Routers tend to be among the most price risk that can be abused to effectively obstruct legal access to the all infrastructure via an effective attack. Furthermore, we plan to build a threat models in which the ransomware attack operator is remotely located underneath a network or storage provider's infrastructure. Ultimately, we aim to examine techniques, which can detect DoS attacks reliably and effectively. This will likely include using machine learning to learn a simple safety profile for each device.

References

Abrar, U., Yousaf, A., Jaffri, N. R., Rehman, A. U., Ahmad, A., Gardezi, A. A., . . . Choi, J.-G. (2021). Analysis of Complex Solid-Gas Flow under the Influence of Gravity through Inclined Channel and Comparison with Real-Time Dual-Sensor System. Electronics, 10(22), 2849.

Agrawal, N., & Tapaswi, S. (2017). Defense schemes for variants of distributed denial-of-service (DDoS) attacks in cloud computing: A survey. Information Security Journal: A Global Perspective, 26(2), 61-73.

Ahmad, S., Cherif, N., Naseer, S., Ijaz, U., Faouri, Y. S., Ghaffar, A., & Hussein, M. (2022). A wideband circularly polarized CPW-fed substrate integrated waveguide based antenna array for ISM band applications. Heliyon, 8(8), e10058.

Ahmad, S., Ijaz, U., Naseer, S., Ghaffar, A., Qasim, M. A., Abrar, F., . . . Abd-Alhameed, R. (2022). A jug-shaped CPW-fed ultra-wideband printed monopole antenna for wireless communications networks. Applied Sciences, 12(2), 821.

Ahmad, S., Manzoor, B., Naseer, S., Ghaffar, A., & Hussein, M. (2021). A Flexible Broadband CPW-Fed Circularly Polarized Biomedical Implantable Antenna With Enhanced Axial Ratio Bandwidth.

Ali, H., Batool, K., Yousaf, M., Islam Satti, M., Naseer, S., Zahid, S., . . . Choi, J.-G. (2022). Security Hardened and Privacy Preserved Android Malware Detection Using Fuzzy Hash of Reverse Engineered Source Code. Security & Communication Networks.

Benkirane, S., Guezzaz, A., Azrour, M., Gardezi, A. A., Ahmad, S., Sayed, A. E., . . . Shafiq, M. Adapted Speed System in a Road Bend Situation in VANET Environment.

Bokhari, S. A., Saqib, Z., Amir, S., Naseer, S., Shafiq, M., Ali, A., . . . Hamam, H. (2022). Assessing Land Cover Transformation for Urban Environmental Sustainability through Satellite Sensing. Sustainability, 14(5), 2810.

Bouyeddou, B., Kadri, B., Harrou, F., & Sun, Y. (2020). DDOS-attacks detection using an efficient measurement-based statistical mechanism. Engineering Science and Technology, an International Journal, 23(4), 870-878.

Cetinkaya, A., Ishii, H., & Hayakawa, T. (2019). An overview on denial-of-service attacks in control systems: Attack models and security analyses. Entropy, 21(2), 210.

Dong, S., Abbas, K., & Jain, R. (2019). A survey on distributed denial of service (DDoS) attacks in SDN and cloud computing environments. IEEE Access, 7, 80813-80828.

Elleithy, K. M., Blagovic, D., Cheng, W. K., & Sideleau, P. (2005). Denial of service attack techniques: analysis, implementation and comparison.

Faouri, Y., Ahmad, S., Naseer, S., Alhammami, K., Awad, N., Ghaffar, A., & Hussein, M. I. (2022). Compact Super Wideband Frequency Diversity Hexagonal Shaped Monopole Antenna with Switchable Rejection Band. IEEE Access, 10, 42321-42333.

Idhammad, M., Afdel, K., & Belouch, M. (2018). Semi-supervised machine learning approach for DDoS detection. Applied Intelligence, 48(10), 3193-3208.

Ishfaq, U., Shabbir, D., Khan, J., Khan, H. U., Naseer, S., Irshad, A., . . . Hamam, H. (2022). Empirical Analysis of Machine Learning Algorithms for Multiclass Prediction. Wireless Communications and Mobile Computing, 2022.

Kaur, P., Nand, P., Naseer, S., Gardezi, A. A., Alassery, F., Hamam, H., . . . Shafiq, M. (2022). Ontology-Based Semantic Search Framework for Disparate Datasets. Intell. Autom. Soft Comput, 32, 1717-1728.

Kissi, M. K., & Asante, M. (2020). Penetration testing of IEEE 802.11 encryption protocols using Kali Linux hacking tools. International Journal of Computer Applications, 975, 8887.

Mohammed, L. A., & Issac, B. (2007). Detailed DoS attacks in wireless networks and countermeasures. Int. J. Ad Hoc Ubiquitous Comput., 2(3), 157-166.

Naseer, S., Ghafoor, M. M., bin Khalid Alvi, S., Kiran, A., Rahmand, S. U., Murtazae, G., & Murtaza, G. (2021). Named Entity Recognition (NER) in NLP Techniques, Tools Accuracy and Performance. Pakistan Journal of Multidisciplinary Research, 2(2), 293-308.

Naseer, S., Hussain, S., Raza, I., Chaudry, S., Mirza, J., & Raza, M. (2012). Mobile ad-hoc network routing protocols: A simulation and performance analysis using multimedia traffic. Journal of Basic and Applied Scientific Research, 2(10), 9925-9930.

Naseer, S., Liu, W., & Sarkar, N. I. (2019). Energy-efficient massive data dissemination through vehicle mobility in smart cities. Sensors, 19(21), 4735.

Naseer, S., Liu, W., Sarkar, N. I., Chong, P. H. J., Lai, E., Ma, M., . . . Qadir, J. (2018). A sustainable marriage of telcos and transp in the era of big data: Are we ready? Paper presented at the International Conference on Smart Grid Inspired Future Technologies.

Ohira, S., Desta, A. K., Arai, I., Inoue, H., & Fujikawa, K. (2020). Normal and malicious sliding windows similarity analysis method for fast and accurate IDS against DoS attacks on in-vehicle networks. IEEE Access, 8, 42422-42435.

Riaz, A. R., Gilani, S. M. M., Naseer, S., Alshmrany, S., Shafiq, M., & Choi, J.-G. (2022). Applying Adaptive Security Techniques for Risk Analysis of Internet of Things (IoT)-Based Smart Agriculture. Sustainability, 14(17), 10964.

Sandhu, U. A., Haider, S., Naseer, S., & Ateeb, O. U. (2011). A survey of intrusion detection & prevention techniques. Paper presented at the 2011 International Conference on Information Communication and Management, IPCSIT.

Sharma, S., & Mittal, M. (2019). Detection and prevention of de-authentication attack in real-time scenario.

Tomar, A., Jeena, D., Mishra, P., & Bisht, R. (2020). Docker security: A threat model, attack taxonomy and real-time attack scenario of dos. Paper presented at the 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence).

Zaman-ul-Haq, M., Saqib, Z., Kanwal, A., Naseer, S., Shafiq, M., Akhtar, N., . . . Hamam, H. (2022). The Trajectories, Trends, and Opportunities for Assessing Urban Ecosystem Services: A Systematic Review of Geospatial Methods. Sustainability, 14(3), 1471.

Zhang, D., Shen, Y.-P., Zhou, S.-Q., Dong, X.-W., & Yu, L. (2020). Distributed secure platoon control of connected vehicles subject to DoS attack: Theory and application. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 51(11), 7269-7278.