An Analysis of DHCP Vulnerabilities, Attacks, and Countermeasures

AbdulAziz AbdulGhaffar
Department of Systems and Computer Engineering
Carleton University
Ottawa, Canada
abdulazizabdulghaff@cmail.carleton.ca

Sumit Kumar Paul
Department of Electrical and Computer Engineering
University of Ottawa
Ottawa, Canada
spaul058@uottawa.ca

Ashraf Matrawy
School of Information Technology
Carleton University
Ottawa, Canada
Ashraf.Matrawy@carleton.ca

***Abstract*—A large number of devices use the Dynamic Host Control Protocol (DHCP) protocol to obtain network configurations like IP address, gateway, Domain Name System (DNS) address, etc. However, the security aspect was not considered thoroughly during its design phase. As a result, it has several very lucrative vulnerabilities to many attackers. In this analysis, we discuss the major vulnerabilities of the DHCP protocol that can result in different attacks. These vulnerabilities include a lack of authentication, confidentiality, and integrity. We also explain different attacks that can be performed by exploiting these vulnerabilities, like rogue DHCP server attacks, DHCP starvation attacks, or replay attacks. Furthermore, we summarize the countermeasures proposed by the researchers to nullify and mitigate these attacks. Moreover, the advantages and drawbacks of the countermeasures are also discussed in this paper.**

***Index Terms*—DHCP, Starvation Attack, DoS Attack, Rogue server, Spoofing, Phishing attack, Replay attack, Man-in-the-middle attack**

I. Introduction

Dynamic Host Control Protocol (DHCP) is a client-server protocol that helps the network-connected hosts to obtain the necessary network configurations like Internet Protocol (IP) address, default gateway, Domain Name System (DNS) address, etc. These configurations are essential to establish network communication with the outer world. Each host in the network, who wishes to obtain network configurations without any manual intervention, executes the client component of this protocol. After receiving the requests, one or more dedicated DHCP servers in the system provide detailed network configurations to the clients. These configurations are valid for a specific period, known as lease time [1], after which the configurations must be renewed.

With the surge of the Internet of Things and smart personal devices [2], the need to assign valid network configurations and IP addresses to these devices is also increased. However, given the importance of the DHCP protocol, the protocol was not built initially with security aspects in consideration. This resulted in exposing various vulnerabilities and the attackers exploited them to launch several attacks [3].

However, this protocol has some built-in weaknesses. One major drawback of DHCP is that the DHCP server and client are not able to authenticate the identity of each other and cannot guarantee that the other node is legitimate. Furthermore, DHCP uses plaintext User Datagram Protocol (UDP) messages. These vulnerabilities in DHCP allow attackers to perform several types of attacks [4], [5]. Several previous research works studied the security of the DHCP protocol [6]–[8]. However, to the best of our knowledge, this is the first study that analyzes the vulnerabilities, attacks, and countermeasures of the DHCP protocol.

The main objective of this paper is to investigate the security aspect of the highly used DHCP protocol. The loopholes, penetrations, attacks, and existing countermeasures are studied to create a solid starting point for researchers to investigate in this direction and make the DHCP protocol robust from the security aspect.

The main contribution of this paper is that we first classify and categorize the vulnerabilities of the original DHCP protocol. Then we discuss how the attackers exploit these vulnerabilities to launch several types of attacks. We also investigate and analyze several countermeasures against these attacks, which the researchers proposed over the years. Finally, we highlight the possible drawbacks of some studies. To the best of our knowledge, no study in the literature analyzes the DHCP protocol's vulnerabilities, attacks, and countermeasures.

This paper is divided into several sections. Section II specifies the criteria for categorizing different vulnerabilities in the DHCP protocol, attack mechanisms, and countermeasures. Section III talks about the main vulnerabilities present in the original DHCP protocol. In Section IV, we analyze different attack mechanisms and grouped them according to the nature of the attack. Section V describes how researchers tried to mitigate each type of attack over the years. Finally, Section VI presents directions for future studies and concludes this analysis paper.

II. Classification Criteria

Fig. 1 shows the classification of DHCP vulnerabilities, attacks, and countermeasures used in this paper. The following section describes the criteria for the categorization.

- **Vulnerabilities:** During classification, we consider three main security vulnerabilities: lack of confidentiality, in-

978-1-6654-9380-2/23/$31.00 ©2023 IEEE

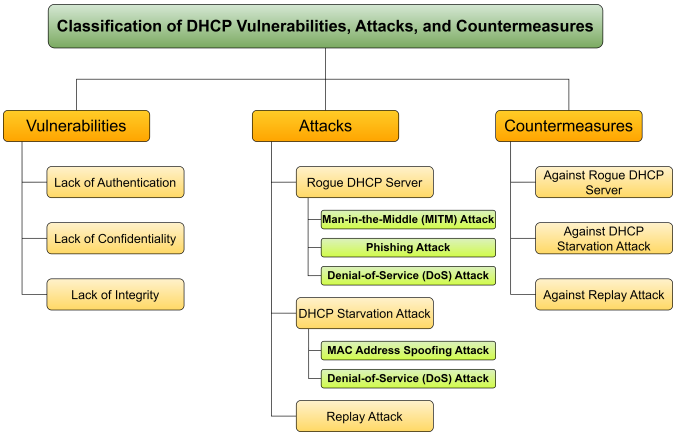

Fig. 1. Classification of DHCP Vulnerabilities, Attacks, and Countermeasures

tegrity, and authentication. These three types are directly related to the design of the DHCP protocol and we explain each of these in the upcoming section.

- **Attacks:** We divide the attacks in hierarchical order with two levels, the first level is the major attacks and the second level is the relevant attacks. The major attacks can happen on the DHCP protocol directly because of its vulnerabilities, whereas the relevant attacks are linked to the major attacks and they are the result of the major attacks. This is shown in Fig. 1, where the major attacks are shown with light-yellow blocks, and their relevant attacks are shown in the green blocks. For example, a Denial-of-Service (DoS) attack can be launched due to either a rogue DHCP server attack or a DHCP starvation attack. It is also important to mention that we did not categorize the attacks based on the vulnerabilities they target in the DHCP protocol because a single attack can be caused as a result of multiple vulnerabilities. Rogue DHCP server attacks, for instance, can happen due to a lack of authentication and confidentiality at the same time. So, we categorize the attacks as major attacks and their related attacks.
- **Countermeasures:** We divide the countermeasures based on the types of attacks they mitigate. We only focus on the major attacks that happen in the DHCP. Other attacks, like Man-in-the-middle (MITM), DoS, and MAC (Media Access Control) address spoofing, are caused as a result of the major attacks, so mitigating the major attacks will prevent these attacks in the DHCP as well.

III. DHCP Vulnerabilities

A. Lack of Authentication

Due to the lack of an authentication mechanism in the DHCP protocol, a malicious node can inject any kind of DHCP packet in the network and can claim to be a valid server or a client [5] to control the network state. The attacker can take advantage of this vulnerability and act as a legitimate server to give the clients wrong network configurations, meanwhile, the clients cannot distinguish a malicious DHCP server from the legitimate DHCP server in the network and use the provided configurations. On the other hand, a malicious client can exhaust the server's resources by sending numerous spoofed requests, whereas the DHCP server cannot detect a malicious client that is flooding the network with DHCP requests. Due to this vulnerability, network administrators often find themselves in a difficult situation to cope with attacks like DHCP starvation attacks [5], [9]–[11], DHCP spoofing [12]–[14], etc.

B. Lack of Confidentiality

Lack of confidentiality in the DHCP protocol means that the information sent to the server by the clients is vulnerable to being read by other unauthorized nodes. This vulnerability arises because the client broadcasts the DHCP packets in cleartext [4]. So, the attacker or a malicious node can silently sniff those messages and obtain precise knowledge about the current state of the network. Moreover, an attacker can control the network state by altering the packet's data on the go. This vulnerability can give rise to MITM attacks that can be a result of rogue DHCP server attacks [15]–[17].

C. Lack of Integrity

There are no built-in mechanisms in the DHCP protocol to ensure the integrity of the messages and protect against unauthorized changes. Since the packets are broadcasted in plaintext, an attacker can capture the transmission and alter the fields of the packet. The attacker can also perform a replay attack by sending the captured packet after some time to masquerade as a legitimate client or server in the network [1], [4], [18]. Another attack that can be performed is MAC address spoofing [9], where the attacker uses fake MAC addresses to launch other attacks like the DHCP starvation attack. As mentioned in section II, these attacks are not strictly linked to the lack of integrity only as they can be caused by the lack of authentication as well. More details on these attacks are provided in the next section.

IV. DHCP Attacks

In this section, we classify several types of attacks that can be launched by exploiting the vulnerabilities of the DHCP protocol. We identify three major attacks that can happen on the DHCP protocol, including, rogue DHCP server attacks, DHCP starvation attacks, and replay attacks.

A. Rogue DHCP server

In a rogue DHCP attack, the attacker tends to set up a fake DHCP server with the intention of providing malicious network configurations to the requesting clients [17]. A malicious user may run a rogue DHCP server without explicit permission from the network administrator and may try several tricks to stay undetected by the network administrator. The attacker tries to respond to client DHCP queries before the legitimate server. If a client receives the message from the rogue server before hearing from a valid DHCP server, then the client uses

the network configuration returned by the rogue server. The rogue DHCP server can also perform several sophisticated attacks on the clients, and these include:

- **Man-in-the-Middle (MITM) Attack:** A rogue DHCP server can specify a malicious gateway while providing DHCP configurations. By doing so, the attacker ensures that all the client traffic traverses through the malicious gateway, which is controlled by the attacker. This enables the attacker to sniff all the traffic.
- **Phishing Attack:** Another attack that a rogue DHCP server can perform is a phishing attack. The rogue DHCP server can specify a malicious DNS server in response to the DHCP request. The attacker can perform various malicious actions on the user traffic, including redirecting to malicious web pages, collecting sensitive user information, etc. For example, when the user tries to visit a web page, the malicious DNS server will redirect the traffic to a lookalike web page which is under the control of the attacker. By doing so, the attacker can steal sensitive information of the user, including usernames and passwords [19].
- **Denial-of-Service (DoS) Attack:** The rogue server can also provide a bogus IP address to a valid DHCP client in response to the *DHCPREQUEST* message. It will make the client's PC unreachable from other hosts in the local network or over the internet. This will prevent the client from accessing the services and will cause a DoS attack for the client [19].

B. DHCP starvation attack

This attack is an example of a malicious client attack, where the attacker sends multiple DHCP requests to the server with spoofed or fake MAC addresses (also known as **MAC address spoofing attack**) utilizing MAC address changer software. Moreover, since the messages are exchanged in clear text, an attacker can intercept a transmission between the client and the server and modify the MAC address and other fields of the packet [17]. This directly impacts the integrity of data exchanged in the messages. The DHCP server will allocate a free IP address from its pool and respond to the request. The attacker aims to utilize the complete pool of assignable IP addresses of the DHCP server and prevent legitimate users from obtaining the IP addresses. Once the complete IP address pool is depleted, the server will not be able to serve the new DHCP requests from legitimate clients, which will cause a **DoS** attack on top of the DHCP starvation attack [20].

A new type of starvation attack in wireless networks is identified in [21]. In wireless networks, the Access Point (AP) mandates a client to register with a unique MAC address, which makes it difficult to spoof. However, the attackers have devised an alternate way to deplete the IP pool in the DHCP server [21]. Before assigning a new IP address, the server probes the network with the to-be-allocated IP address to check whether it is already in use. A malicious insider client sends a fake reply to that request. Due to this, the server marks the IP as used. So, even if that IP address is available, it never becomes usable. On the other hand, after obtaining a new IP address, the client probes to verify whether the newly allocated IP is already in use or not. Another insider malicious client may sniff for these probes and send a fake reply indicating it is already used by someone else. Due to this, the victim client can send a *DHCPDECLINE* message [10]. After listening to this, the client marks this IP as unusable. Hence, even if an IP address is available, it cannot be allocated to anyone.

C. Replay attack

A replay attack is possible mainly due to the lack of usage of any kind of nonce. In this case, the attacker captures a packet between the client and the server during their communication. The attacker then re-sends this packet to the server or to the client later, depending on the type of the message, to masquerade as a legitimate user or a server [17]. The importance of this attack is evident from the fact that the attacker can sniff and capture the packet without being detected. Furthermore, the attacker does not require any advanced capabilities to perform such attacks. Various research works offer several countermeasures to this type of attack, which we explain in the following section.

V. Countermeasures

This section categorizes the countermeasures available in the literature based on the attacks they mitigate.

A. Against Rogue DHCP Server Attack

The need for protection against the rogue server was identified long ago. A Kerberos-based solution is drafted in [22]. However, that never came as a complete Request for Comments (RFC) due to the complexity of the infrastructure. To implement this in the real network along with DHCP servers, other components like an authentication server (AS), Ticket-Granting server (TGS), etc., are required.

After this draft, in 2017, a very similar approach was proposed in the P-DHCP protocol [5]. It relies on symmetric-key cryptography. Key distribution and authentication server (KDA) is installed in the network to authenticate both the DHCP server and the clients. Along with entity authentication, this solution performs message authentication as well.

Agarwal et al. [15] proposed a Measurement Inconsistent Discrete Event System (MIDES) framework to detect and diagnose the faults in various computational systems. Furthermore, the authors extend their proposed MIDES system to include intrusion detection capabilities. This intrusion detection system (IDS) can detect and reduce the threats of rogue DHCP server attacks, specifically in wireless networks. The proposed scheme can detect a malicious DHCP server using a dummy client. They also performed a case study involving the detection of rogue DHCP server attacks using their proposed mechanism. The authors set up a network environment in a machine running Kali Linux, and the implementation of the DHCP server was provided by the ISC-DHCP server package. The authors utilized the Scapy packet manipulation library from Python to handle network packets. The results show that

their IDS mechanism is able to achieve a detection rate of 99%. The authors carried out a small-scale and limited simulation analysis. The study is missing a detailed evaluation of the proposed mechanism for DHCP attacks.

Software-defined networking (SDN)-based [23], [24] solution is proposed in [14] called Network Flow Guard (NFG). This is an automated application used to identify and crack down on rogue DHCP servers in the network. The operation of this application is based on a predefined list of authorized DHCP servers in the network and the SDN controller uses this to identify a potential rogue DHCP server in the network. The authors implemented a Mininet-based testbed using a POX SDN controller, OpenFlow-enabled SDN switches, and ISC-DHCP server to evaluate the performance of their proposed solution. The authors provide a very brief summary of their results; however, this study is missing an in-depth analysis of their proposed system and its evaluation.

A digital certificate-based authentication can be used to verify the identity of the DHCP server [19]. The DHCP server signs each message with its private key, and the client can validate that using the corresponding public key. Digital certificates should be present on the client side beforehand; from this, the server's public key can be extracted. However, this solution has a drawback. DHCP packets cannot be fragmented, so a large amount of data like digital certificates cannot be transferred to the client using these packets. So, this solution requires the manual installation of certificates in each DHCP client. The situation becomes more problematic if some certificate gets compromised.

The solution proposed by the authors in [16] implements the functionality of the DHCP client on the actual DHCP server to find out any rogue server in the network. The client component will periodically send *DHCPDISCOVER* requests and will find out how many *DHCPOFFER* messages it receives. After parsing these responses, the malicious DHCP server can be detected because the server will know the list of trusted DHCP servers operating in the network. The authors built a prototype network using several virtual machines (VM) with the help of VirtualBox. The DHCP server runs a custom-built Python script that scans the network in the background to identify potential rogue servers. The results show that their proposed solution can identify rogue DHCP servers. However, if the malicious DHCP server knows the address of the benign DHCP server, it may not respond to its requests. If the client MAC address mentioned in the *DHCPDISCOVER* request matches the actual DHCP server's address, then the attacker can stay quiet.

Research work done by Rietz et al. [25] proposes a solution by utilizing the properties of the SDN network to mitigate common local area network (LAN) attacks. These include a rogue DHCP server attack along with several others. The authors implement vulnerable protocols, like DHCP and Address Resolution Protocol (ARP), as network applications on top of the SDN controller. This reduces the broadcast and has fewer chances of attacks, as the client requests will be sent directly to the SDN controller through the data plane switches. Additionally, the authors also implement host authentication at switch ports using Extensible Authentication Protocol (EAP) [26]. Finally, the authors implement the proposed solution on the Mininet emulator, and the results show that their proposed mechanism blocks around 95% of the attacks compared to the 68% rate of traditional networks.

Shete et al. [17] proposed a One-time password (OTP)-based authentication mechanism, where each client will register itself with a valid server using a mobile phone number. During the execution of the DHCP protocol, the OTP will be sent to the client's mobile number, which is required to proceed further. A rogue server should not know the client and mobile number mapping, so this attack cannot occur.

B. Against DHCP Starvation Attack

Generally, the cause of the starvation attack is a malicious host who has access to the network. A shared secret-based standard solution was proposed in RFC 3118 to authenticate DHCP clients, and DHCP messages [27]. However, it is rarely used due to the problem of establishing the shared secret. Moreover, if the valid host shares the secret with a malicious host, then the malicious host can act as a valid DHCP client.

A patent was filed in 2011 by De Graaf et al. [28] for their solution to authenticate the DHCP clients by using Challenge Handshake Authentication Protocol (CHAP) [29] based challenge-response protocol. However, this solution requires additional entities like the Remote Authentication Dial-In User Service (RADIUS) server.

Issac proposed a secure DHCP protocol to protect against MAC address spoofing attacks [30]. The author proposed three variants of the DHCP protocol. The client and the server use a shared secret key during DHCP negotiation in the first variant. In addition to the secret key, the client and server also share a random number for added security in the second variant. Whereas the third proposed DHCP protocol uses a session key along with the secret key and the random number to enhance security. To evaluate the performance of their protocol, the author collects network traffic from an office network using Ethereal software. The author concludes that their proposed protocols provide higher protocol run time while providing additional security compared to the traditional DHCP protocol.

Tripathi et al. [10] proposed an anomaly detection scheme for DHCP messages. Normal DHCP traffic is observed in the training phase, and a probability distribution function of different DHCP messages is determined. Then, the observed traffic pattern is compared with the normal one in the testing phase. If the observed difference is more than the predetermined threshold, the attack is detected.

Yaibuates et al. [9] suggested the use of ICMP messages to detect requests from malicious DHCP clients. It monitors the rate of *DHCPREQUEST*, and if it is more than a threshold, then ICMP ECHO is sent to the allocated IP addresses to verify whether it was an attack. If it were an attack, then the reply would not come. In some hosts, ICMP is blocked by the firewall; for these situations, the authors suggested sending an ARP request to get the corresponding IP address [31]. And if

the allocated IP is not used, the reply will not come back and the attack can be detected.

The work done by Narwal et al. [20] proposes a game-theory-based solution to prevent DoS attacks in the OpenStack cloud environment. The solution implements a firewall or an intrusion detection system (IDS) in the network to prevent an attacker from launching the attack. However, the authors did not provide additional details on the implementation of their system and did not evaluate their solution's performance.

C. Against Replay Attack

Replay attacks can be mitigated with the help of digital certificates and some special replay detection value [19]. The authors use the current time as a replay detection value before signing the DHCP message. So, no one can replay the same message at some later point in time, as the replay detection value field will not match the current time. However, this solution requires clock synchronization among all the clients and the DHCP server.

The authors in [1] propose two protocols to mitigate the lack of liveliness checks in the DHCP protocol. The first protocol uses digital certificates to enhance the security of DHCP, whereas the second protocol implements a shared secret key between the client and the server. The authors claim that their proposed protocols can prevent the network from replay attacks and MAC address spoofing with improved authentication of DHCP. The authors implemented a small-scale prototype to evaluate the performance and concluded from the results that their protocols could perform within 1.5 seconds. However, this research lacks an extensive evaluation.

Dinu et al. [32], [33] propose a DHCPAuth module to introduce authentication in the DHCP protocol. The proposed solution works as a firewall for the DHCP protocol. It scans and verifies all the communications with the DHCP server. For packets arriving at the DHCP server, the module verifies the authenticity of the messages, and if any message is not verified, it will be dropped immediately. On the other hand, packets sent by the DHCP server are authenticated and signed by the server's private key. The authors allow network administrators to choose from any of the two trust models flexibly, Public Key Infrastructure (PKI) and Pretty Good Privacy (PGP) [34], in their modules. The authors also evaluate the performance of their module and conclude from the results that the proposed module introduces a slight overhead of less than 40 ms while providing the authentication.

Jero et al. [18] identify a new type of attack in SDN networks known as Persona Hijacking, in which the attacker can break the bindings between different network layers of a target node. An example of binding, in this case, can be the binding between the client's IP address and MAC address [18]. This can result in a replay attack or a rogue DHCP server attack. The authors propose a SecureBinder algorithm that protects against such types of attacks. This system contains a list of authenticated bindings and does not allow any changes to these unless authenticated. The system also contains an authenticator that verifies the MAC address of the hosts in the network. The authors evaluated the performance of their proposed defense mechanism by implementing a network in the Mininet emulator. From their results, the authors claim that their proposed mechanism can protect against various types of network attacks with very little overhead.

Replay attacks can also be mitigated by the use of OTP-based authentication of the users [17] as the OTP will change over time, so it will not be possible to replay the captured packets at some later point in time. However, this solution requires an additional OTP generator and manual registration of the mobile numbers in the DHCP server. A similar approach is taken by Zhang et.al. [4]. Here, instead of sending the OTP to the user's mobile number, a one-time key is generated using the current time, the MAC address, and the client's secret key. That one-time key is used to authenticate the DHCP messages and clients. As time changes, no one can replay the captured packet at a later time. However, this approach requires precise synchronization among all the nodes within a network. Furthermore, the exact mechanism of communicating the one-time key between the client and the DHCP server is not clearly mentioned.

In conclusion, the DHCP protocol's security should be enhanced by adopting techniques for authentication, confidentiality, and integrity during its operation.

VI. Conclusion and Future work

Although the DHCP protocol was invented long ago, its security aspect is still something to ponder upon. Especially when billions of devices are using it, in today's wireless and IoT-driven era. The original DHCP protocol lacks confidentiality, integrity, and authenticity, and the attackers have exploited these vulnerabilities over the years.

In this paper, we first outline the major vulnerabilities in the design of the DHCP protocol. We further investigate how the attackers exploit these vulnerabilities and launch several types of attacks. Finally, we discuss the countermeasures proposed by the researchers over the years. We have also analyzed them and pointed out their positive and negative aspects. However, another important concept that is worth mentioning is DHCP snooping [12]. DHCP snooping is a layer two port security technique used by layer two switches to mitigate DHCP attacks. The network administrator identifies the trusted and untrusted interfaces on their switch. Then, the legitimate or trusted DHCP server is connected to the port that is labeled as trusted. While the hosts in the network can be connected to untrusted interfaces. The main aim of this technique is to accept the server-related DHCP messages only from the trusted interface while dropping these packets on the untrusted ports. The server-side DHCP packets include *DHCPOFFER*, *DHCPACK*, and *DHCPNAK* messages issued by the DHCP server. The switch can also perform additional functionalities like performing packet inspections and limiting the rate of DHCP traffic that can be sent on the interfaces. The switch keeps track of the information it receives with the help of a database. This technique can prevent some attacks mentioned in our study.

Although we have found several vulnerabilities and related attacks in this analysis, this investigation is not exhaustive. This study can be thought of as the starting point for further studies and research related to the security of the DHCP protocol. One important research area is securing the communication within the DHCP protocol. Since DHCP uses UDP protocol to exchange messages between client and server, secure mechanisms must be proposed to overcome the current vulnerabilities of the DHCP protocol. A secure DHCP protocol must provide authentication along with protecting the confidentiality and integrity of the exchanged messages. Along with security, there are several privacy-related implications as well. DHCP servers can monitor which MAC address is asking for a new IP address and when. From this information, the DHCP server may infer several private information and characteristics of the connected devices. Moreover, in the last few years, networking has changed a lot due to the introduction of SDN, IPv6, etc. Therefore, finding out additional countermeasures against possible attacks in those areas is one of the important future research directions.

REFERENCES

[1] S. Duangphasuk, S. Kungpisdan, and S. Hankla, "Design and implementation of improved security protocols for DHCP using digital certificates," in *2011 17th IEEE International Conference on Networks*. IEEE, 2011, pp. 287–292.

[2] A. AbdulGhaffar, S. M. Mostafa, A. Alsaleh, T. Sheltami, and E. M. Shakshuki, "Internet of things based multiple disease monitoring and health improvement system," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 3, pp. 1021–1029, 2020.

[3] N. Hubballi and N. Tripathi, "A closer look into DHCP starvation attack in wireless networks," *Computers & Security*, vol. 65, pp. 387–404, 2017.

[4] F. Zhang and L. Chen, "OTP_SAM: DHCP security authentication model based on OTP," in *2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, 2016, pp. 346–350.

[5] O. S. Younes, "Securing ARP and DHCP for mitigating link layer attacks," *Sādhanā*, vol. 42, no. 12, pp. 2041–2053, 2017.

[6] M. Aldaoud, D. Al-Abri, A. Al Maashri, and F. Kausar, "Dhcp attacking tools: an analysis," *Journal of Computer Virology and Hacking Techniques*, vol. 17, pp. 119–129, 2021.

[7] M. S. Tok and M. Demirci, "Security analysis of sdn controller-based dhcp services and attack mitigation with dhcpguard," *Computers & Security*, vol. 109, p. 102394, 2021.

[8] B. Prabadevi and N. Jeyanthi, "A review on various sniffing attacks and its mitigation techniques," *Indones. J. Electr. Eng. Comput. Sci*, vol. 12, no. 3, pp. 1117–1125, 2018.

[9] M. Yaibuates and R. Chaisricharoen, "ICMP based malicious attack identification method for DHCP," in *The 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (JICTEE)*. IEEE, 2014, pp. 1–5.

[10] N. Tripathi and N. Hubballi, "A probabilistic anomaly detection scheme to detect DHCP starvation attacks," in *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. IEEE, 2016, pp. 1–6.

[11] C. Toprak, C. Turker, and A. T. Erman, "Detection of DHCP starvation attacks in software defined networks: A case study," in *2018 3rd International Conference on Computer Science and Engineering (UBMK)*. IEEE, 2018, pp. 636–641.

[12] Y. Tong and S. Akashi, "A feasible method for realizing leakage of DHCP transactions under the implementation of DHCP snooping: To what extent can DHCP snooping protect clients from the cyberattack based on DHCP spoofing," in *Proceedings of the 2019 2nd International Conference on Data Science and Information Technology*, 2019, pp. 267–272.

[13] J.-L. Wang and Y.-C. Chen, "An SDN-based defensive solution against DHCP attacks in the virtualization environment," in *2017 IEEE Conference on Dependable and Secure Computing*. IEEE, 2017, pp. 529–530.

[14] J. H. Cox Jr, R. J. Clark, and H. L. Owen III, "Leveraging SDN to improve the security of DHCP," in *Proceedings of the 2016 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, 2016, pp. 35–38.

[15] M. Agarwal, S. Biswas, and S. Nandi, "Discrete event system framework for fault diagnosis with measurement inconsistency: case study of rogue DHCP attack," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 789–806, 2017.

[16] M. S. Makarova and A. A. Maksutov, "Methods of detecting and neutralizing potential DHCP rogue servers," in *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*. IEEE, 2021, pp. 522–525.

[17] A. Shete, A. Lahade, T. Patil, and R. Pawar, "DHCP protocol using OTP based two-factor authentication," in *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*. IEEE, 2018, pp. 136–141.

[18] S. Jero, W. Koch, R. Skowyra, H. Okhravi, C. Nita-Rotaru, and D. Bigelow, "Identifier binding attacks and defenses in software-defined networks," in *26th USENIX Security Symposium*, 2017, pp. 415–432.

[19] D. D. Dinu and M. Togan, "DHCP server authentication using digital certificates," in *2014 10th International Conference on Communications (COMM)*. IEEE, 2014, pp. 1–6.

[20] P. Narwal, S. N. Singh, and D. Kumar, "Game-theory based detection and prevention of DoS attacks on networking node in open stack private cloud," in *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions)(ICTUS)*. IEEE, 2017, pp. 481–486.

[21] N. Tripathi and N. Hubballi, "Exploiting DHCP server-side IP address conflict detection: A DHCP starvation attack," in *2015 IEEE International Conference on Advanced Networks and Telecommuncations Systems (ANTS)*. IEEE, 2015, pp. 1–3.

[22] K. Hornstein, T. Lemon, B. Aboba, and J. Trostle, "DHCP authentication via Kerberos V," *IETF DHC Working Group*, 2001.

[23] A. Abdulghaffar, A. Mahmoud, M. Abu-Amara, and T. Sheltami, "Modeling and evaluation of software defined networking based 5G core network architecture," *IEEE Access*, vol. 9, pp. 10 179–10 198, 2021.

[24] A. A. Abdul Ghaffar, A. Mahmoud, T. Sheltami, and M. Abu-Amara, "A Survey on Software-Defined Networking-Based 5G Mobile Core Architectures," *Arabian Journal for Science and Engineering*, pp. 1–18, 2022.

[25] R. Rietz, R. Cwalinski, H. König, and A. Brinner, "An SDN-based approach to ward off LAN attacks," *Journal of Computer Networks and Communications*, vol. 2018, pp. 1–12, 2018.

[26] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "RFC 3748: Extensible Authentication Protocol (EAP)," 2004.

[27] R. Droms and W. Arbaugh, "RFC3118: Authentication for DHCP messages," 2001.

[28] K. De Graaf, J. Liddy, P. Raison, J. C. Scano, and S. Wadhwa, "Dynamic host configuration protocol (DHCP) authentication using challenge handshake authentication protocol (CHAP) challenge," Oct. 8 2013, US Patent 8,555,347.

[29] W. Simpson, "PPP challenge handshake authentication protocol (CHAP)," 1996.

[30] B. Issac, "Secure ARP and secure DHCP protocols to mitigate security attacks," *arXiv preprint arXiv:1410.4398*, 2014.

[31] M. Yaibuates and R. Chaisricharoen, "A combination of ICMP and ARP for DHCP malicious attack identification," in *2020 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*. IEEE, 2020, pp. 15–19.

[32] D. D. Dinu, M. Togan, and I. Bica, "On DHCP security," *Proceedings of the Romanian Academy Series A: Mathematics, Physics, Technical Sciences, Information Science*, vol. 18, pp. 403–412, 2017.

[33] D. D. Dinu and M. Togan, "DHCPAuth—a DHCP message authentication module," in *2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics*. IEEE, 2015, pp. 405–410.

[34] D. Atkins, W. Stallings, and P. Zimmermann, "PGP message exchange formats," Tech. Rep., 1996.