Sādhanā (2020)45:33
https://doi.org/10.1007/s12046-019-1244-4

DHCPv6Auth: a mechanism to improve DHCPv6 authentication and privacy

AYMAN AL-ANI, MOHAMMED ANBAR*, AHMED K AL-ANI and
IZNAN HUSAINY HASBULLAH

National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia, Gelugor, Penang, Malaysia
e-mail: dr.ayman.khallel@gmail.com; anbarnav6@gmail.com

MS received 8 December 2018; revised 4 May 2019; accepted 7 November 2019

Abstract. Internet Protocol version 6 (IPv6) deployment continues to gain ground due to the increasing demand for IP addresses generated by the number of Internet facing devices, and it is compounded by the exhaustion of allocatable IPv4 addresses. Dynamic Host Configuration Protocol version 6 (DHCPv6) is used to allocate IPv6 addresses and distribute network configuration information to IPv6 hosts in a link-local network. However, DHCPv6 messages in transit expose identifiable information of the DHCPv6 client that could be used by malicious users to track their victims. Additionally, the lack of an authentication mechanism leaves IPv6 hosts vulnerable to rogue DHCPv6 server attacks. This paper introduces DHCPv6 Authentication (DHCPv6Auth) mechanism to prevent rogue DHCPv6 server attacks and protect the privacy of IPv6 hosts. DHCPv6Auth uses the Ed25519 digital signature algorithm for authentication and could be used in conjunction with Anonymity Profile mechanisms for privacy protection. The DHCPv6Auth mechanism was compared with other mechanisms in terms of processing time, prevention of rogue DHCPv6 server attack, and protection of users' privacy. The results show that it requires less processing time and traffic overhead than other authentication mechanisms; is able to prevent rogue DHCPv6 server attacks; and provides better privacy protection for the IPv6 host than other authentication mechanisms to which it was compared.

Keywords. Rogue DHCPv6 server; privacy; DHCPv6; IPv6; digital signature; DoS.

1. Introduction

Internet Protocol version 6 (IPv6) is the new version of Internet address protocol that is intended to replace Internet Protocol version 4 (IPv4) due to IPv4 address exhaustion [1]. The State of IPv6 Deployment 2018 Reports shows a persistent increase in IPv6 use [2]. Researchers expect that by 2025, most of the Internet traffic will use IPv6 instead of IPv4 [3]. Thus, the security of IPv6 is essential. However, IPv6 still faces several security challenges, such as fragmentation attacks and Denial of Service (DoS) [4].

IPv6 has a new protocol, Dynamic Host Configuration Protocol version 6 (DHCPv6), which is used to assign IP addresses for IPv6 hosts [5, 6]. Although IPv6 has another mechanism, the Stateless Address Autoconfiguration (SLAAC), to assign the IP address to the host, DHCPv6 gives network administrators more control of the network than the SLAAC mechanism [1]. In addition, DHCPv6 is utilized to distribute other information of the network configuration to the host in the IPv6 link-local network [7]. The Internet Engineering Task Force (IETF) organization provides over 30 documents for the services [8] that can be distributed by DHCPv6. This information includes the Domain Name System (DNS) and the Network Time Protocol (NTP) [9, 10]. Therefore, DHCPv6 is widely utilized in the IPv6 network.

In spite of many benefits of DHCPv6, attackers can exploit DHCPv6 messages to inject harmful network configuration information into hosts and divert their traffic towards rogue DNS and NTP servers or to cause a connection failure in the network [11, 12]. Researchers have proposed several security mechanisms to prevent rogue DHCPv6 server attack. However, most of these security mechanisms, such as those described in [13] and [14], lack a mechanism to deploy them in a large-scale network [15].

Furthermore, DHCPv6 messages expose critical information about IPv6 host [16–20]. Such information can include the type of device and information about the operating system, which could be leveraged by attackers with knowledge of the device or specific vulnerabilities of the software to swiftly locate possible targets in the IPv6 link-local network. In designing DHCPv6 [13], the privacy issue was overlooked. As a result, the IETF organization published a Request for Comments (RFC) 7824 to highlight DHCPv6 privacy issues [19].

This paper reviews the security vulnerabilities of DHCPv6 in IPv6 link-local networks and examines the

*For correspondence

Published online: 29 January 2020

presented authentication mechanisms that aim to secure the DHCPv6 protocol. Furthermore, it proposes a DHCPv6 authentication mechanism that can be deployed in large-scale network and also protect hosts' privacy. The proposed mechanism is named DHCPv6 Authentication (DHCPv6Auth). The remainder of the paper is organized as follows: section 2 elaborates on the background of the DHCPv6 process and its threat model. Section 3 reviews related studies. The design of DHCPv6Auth is introduced in section 4. Implementation of DHCPv6Auth is provided in section 5. The experiments, as well as the evaluation, are provided in section 6. The results of the experiments are analyzed and discussed in section 7. Finally, section 8 provides the conclusion, as well as suggestions for further studies.

2. Background

DHCPv6 is used to assign IPv6 addresses and distribute network configuration information, such as NTP server address and DNS server address [21, 22], to IPv6 hosts. DHCPv6 servers operate in two modes: stateless and stateful. The DHCPv6 server in the stateful mode is used to assign IPv6 address and distribute network configuration information. In contrast, the server in the stateless mode is only used to distribute network configuration information.

When an IPv6 host joins a new IPv6 network, the host multicasts a Router Solicitation (RS) message. A router replies with a Router Advertisement (RA) message that contains information on the DHCPv6 server mode. Based on this information, the host will configure its IPv6 address accordingly.

In the stateful mode, the host multicasts a DHCPv6 Solicit message to all DHCPv6 servers located on the link-local network. The server will then respond to the host with a DHCPv6 Advertise message that contains related configuration information. Next, the host will send a DHCPv6 Request message to confirm the configuration. Finally, the server should send a DHCPv6 Reply message to confirm the configuration [23]. Figure 1 illustrates the basic DHCPv6 server processes during the stateful mode.

On the other hand, in the stateless mode, the host should multicast an Information-request message, and the server should reply with a DHCPv6 Reply message that contains configuration information [24]. DHCPv6 is considered one of the main components in IPv6 network. However, DHCPv6 is vulnerable to attack, and the most common vulnerabilities will be discussed in the following sections.

2.1 *Rogue DHCPv6 server attack*

The main issue with DHCPv6 is that IPv6 host configures its IP address and other network configurations based on the DHCPv6 server message it received without verifying the source and integrity of the message [25]. Therefore, attacker on the same network can masquerade as a legitimate server by crafting and sending spoofed messages to the host when the host sends a Solicit message asking the server to reply. If there is an attacker on the network, it will respond back via an Advertise message containing incorrect configuration information. Since the host does not have any mechanism to verify the source of this message, it will accept the message and configure its IP address and other network configurations with the incorrect configuration. Hence, the host falls victim to the attack, such as DoS or man-in-the-middle attack by redirecting the user's traffic to rogue servers as shown in figure 2 [26–28].

2.2 *DHCPv6 privacy*

DHCPv6 messages may reveal crucial information about IPv6 hosts. This information can be used to fingerprint the hosts, as it can reveal the device type, vendor name, operating system type, or specific version of operating system. This information can then be exploited to monitor

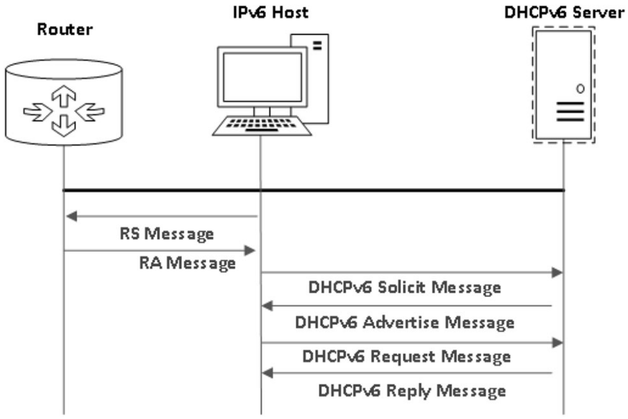


Figure 1. DHCPv6 operation in stateful mode.

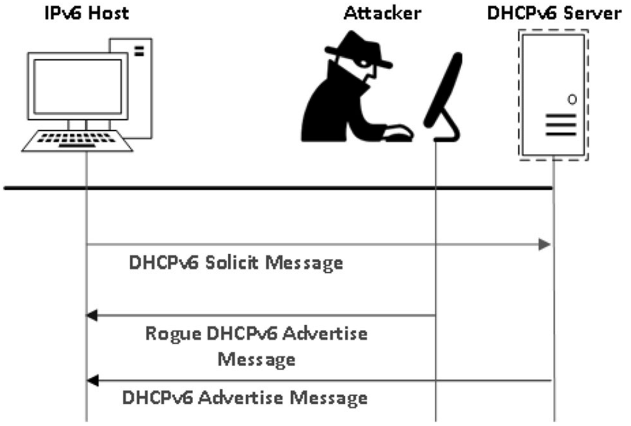


Figure 2. DHCPv6 security challenge.

the hosts and to identify the potential vulnerabilities of the device, vendor, or operating system.

Furthermore, the DHCPv6 information includes several identifiers, such as the DHCP Unique Identifier (DUID) and hostname. These identifiers can be used as a stable identity to trace the host activity over time. The stable identity is unique information that distinguishes the hosts, and it does not change over time or if so, very infrequently.

In addition, the attackers can correlate the DHCPv6 information with various other information extracted from traffic analysis to identify the device, device properties, and most likely its user. Further, the DHCPv6 message reveals the previously visited network. Whenever a host connects to a network, the host attempts to retain and reuse its old IPv6 address by sending the old IPv6 address to the DHCPv6 server. The user's old IPv6 address, if exposed, reveals the previously visited network of the user to the attacker. Therefore, the DHCPv6 information is prone to disclosure [19].

3. Related work

Many mechanisms have been proposed to provide authentication and privacy protection during the DHCPv6 process. The Delayed Authentication Mechanism (DAM), defined in RFC 3315 [13], is one such mechanism. The DAM mechanism uses Hash Message Authentication Code (HMAC) with message-digest algorithm (MD5). HMAC-MD5 is a symmetric authentication mechanism which uses a secret key to authenticate the DHCPv6 server message. The IPv6 host and server need to share this secret key. The DAM mechanism intends to prevent rogue DHCPv6 server attack and replay attack. However, DAM does not provide a mechanism to distribute the secret key between the host and the server. Thus, the secret key is distributed manually, which makes the proposed mechanism difficult to deploy and manage in large-scale networks. In addition, DAM uses a key ID which can be used as a unique identifier to trace the host and threaten its privacy.

Due to the difficulties in deploying a secret key in DAM, RFC 3315 provides another authentication mechanism, Reconfigure Key Authentication Mechanism (RKAM), which is designed to protect the host from the DHCPv6 server Reconfigure messages. RKAM is similar to DAM as it also uses HMAC-MD5, but the RKAM sends the secret key in plaintext during the initial transmitting messages to the host. Therefore, attackers can sniff the secret key and use it to authenticate their messages. Thus, RKAM cannot prevent a rogue DHCPv6 server attack [13].

Agarwal *et al* proposed an Intrusion Detection System (IDS) based on the Measurement Inconsistent Discrete Event System (MIDES) framework to detect rogue DHCP server attack by mapping the attack to a fault in the Discrete Event System (DES) model. This mechanism is easy to deploy, has low resource overhead, and requires no protocol modification [29]. However, the proposed IDS was only tested with DHCPv4; it was not tested with DHCPv6, which has a different message type and format. Therefore, this IDS needs to be modified before it can work with DHCPv6. Further, it was designed to only detect rogue DHCPv6 server attacks, but not to stop them, which means that it requires the intervention of a network administrator or operator to prevent the attack. Therefore, this mechanism may not be suitable in many cases.

Regarding privacy, Huitema *et al* suggest a mechanism called Anonymity Profile [20] to protect the privacy of DHCPv6 host. The Anonymity Profile mechanism avoids using any options, such as the Class Vendor option and the User Class option, that may reveal information about the host. Furthermore, it uses random DUID host to thwart attempts by attacker to correlate the host activities with the user. However, the Anonymity Profile mechanism cannot be used with DAM because DAM requires the use of a unique identifier for each host that could be used to fingerprint the host.

The Secure DHCPv6 mechanism is another authentication mechanism proposed by Li *et al*, and it provides authentication and privacy protection for DHCPv6 messages [27]. This mechanism uses Asymmetric-Key Cryptography (AKC) to provide authentication and privacy protection for DHCPv6 messages [30]. Every IPv6 host and DHCPv6 server has their own digital certificate for signing and encrypting DHCPv6 messages. However, it is difficult to deploy in large-scale network since the host and server need to be manually configured with trusted certifications. This mechanism requires two extra messages before starting the actual DHCPv6 communication. The host and server should verify the digital certificate and digital signature algorithm (DSA), as well as decrypt DHCPv6 messages. This makes the process much more complicated. The Secure DHCPv6 also puts a limit on the maximum message size due to the use of Rivest–Shamir–Adleman (RSA) algorithm which is not designed to encrypt large message size [31–33]. In addition, the Secure DHCPv6 mechanism requires the use of DSA, which may be used as a unique identifier for the host message.

Most authentication mechanisms do not provide a mechanism to distribute the key, IP address, or certification between the hosts and server; therefore, they are difficult to deploy in a large-scale network. Furthermore, the Anonymity Profile, which intends to protect the host's privacy, cannot be used to conjunction with other authentication mechanisms. In addition, the Secure DHCPv6 mechanism, which provides authentication and privacy protection, has very complex processes, fails to provide privacy and put a limit on the DHCPv6 message size.

4. Proposed DHCPv6Auth

From the standard DHCPv6 processes, the IPv6 host does not have a mechanism to verify the integrity and source of the DHCPv6 server message, which leads the host to configure itself with a rogue DHCPv6 server message. Furthermore, the DHCPv6 message exposes critical information about the host which affects the host's privacy. The current mechanism does not provide a way to distribute the verification information and protect the privacy of the host. Hence, the design goals of the DHCPv6Auth mechanism are the following:

- Provide a simple mechanism to distribute the verification key.
- Prevent rogue DHCPv6 server attacks.
- Prevent replay attacks.
- Enable use in conjunction with Anonymity Profile mechanism for privacy protection.
- Avoid modifying the DHCPv6 message format or introducing new DHCPv6 messages.

4.1 *Distribution DHCPv6 keys*

To achieve the main goals of this study, DHCPv6Auth utilizes an asymmetric authentication algorithm, DSA [34] to prevent rogue DHCPv6 server attack. DSA uses public and private keys for signing and verifying the DHCPv6 server message. DHCPv6Auth uses Ed25519 for DSA.

Ed25519 is an Edwards-curve DSA (EdDSA) signature scheme using SHA-256/512 and Curve25519, which is an elliptic curve offering 128 bits of security and is designed for use with the elliptic curve Diffie–Hellman (ECDH) key agreement scheme. Ed25519 was selected for use in DHCPv6Auth because of its high performance on a variety of platforms. Further, Ed25519 has several unique DSA properties, such as small public key size, small signature size, and deterministic nonce generation, which makes Ed25519 more efficient for security purposes than other DSAs. Ed25519, similar to other DSAs, uses a public and private key pair [35]. The DHCPv6 server signs the DHCPv6 server message by using Ed25519 with the private key, and the host verifies the DHCPv6 server message by using Ed25519 with the public key.

Distributing the public key manually to the host makes the proposed mechanism difficult to deploy and manage in large-scale networks. In order to overcome this issue, DHCPv6Auth uses RA message to distribute the public key to the hosts. When the host joins the network, it has to get network configuration information, such as a network prefix, a maximum transmission unit (MTU), and an address mode (i.e., stateless or stateful) [36, 37]. This information can only be obtained by using the RA message. Therefore, the DHCPv6Auth utilizes the RA message for distributing the public key to the hosts, rather than using a third party, which may require sending extra messages. The DHCPv6Auth requires the RA message to be secured by using third-party mechanisms such as Secure Neighbor Discovery(SEND) [35] or RA Guard (RA-Guard) [35].

The public and private keys should be generated by the DHCPv6 server that exists on the link-local network. The generated public key would then be manually deployed to each router in the local network.

DHCPv6Auth introduces two options named DPK to convey the server public key by RA message, and Signature Authentication to convey the signature by the DHCPv6 server message. The following section explains the format of these options.

4.2 *DPK option format*

To allow an RA message to convey a public key to the hosts, a new RA option has been designed, named the DHCPv6 public key (DPK) option. The DPK option format follows the DHCPv6 option format as defined in RFC 4861 [38] as shown in figure 3.

The DPK option fields are configured as follows: Option Type is set to 253, which refers to RA option type; Option Length is set to 3 which refers to the option size; Reserved is set to zero which is an unused field; and DHCPv6 Public Key is set to the server public key.

4.3 *SA option format*

To allow the DHCPv6 server message to convey the signature that will be generated by the Ed25519, the DHCPv6Auth utilizes an SA DHCPv6 authentication option. The SA option is used to convey the digital signature as shown in figure 4.

The SA option fields are configured as follows: The Option type is set to 11 which refers to option type; Option Length is set to 79 which refers to the option length; the Protocol Type field is set to 4; which refers to the authentication option used in the DHCPv6Auth mechanism; and the Algorithm field is set to 4, which refers to the mechanism using Ed25519; Furthermore, the DHCPv6Auth sets the RDM field to zero, and the Replay Detection (RD) field is set to the current time of the DHCPv6 server. The RDM

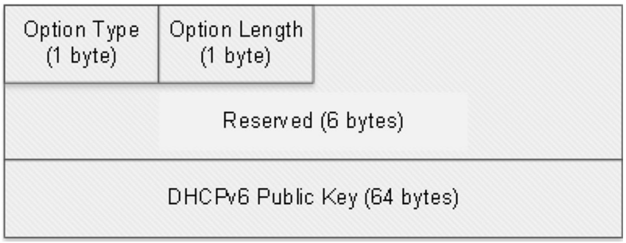


Figure 3. DPK option format.

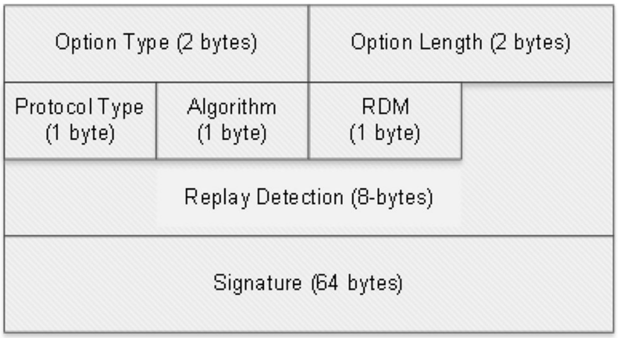


Figure 4. SA option format.

and RD fields will be used to prevent replay attack. This will be illustrated in section 4.5. Further, the Signature field will have the digital signature value to convey.

4.4 *Signature process*

The DHCPv6 server should sign the DHCPv6 server message by using Ed25519 with the private key and append the signature to the DHCPv6 message for verifying the source and the integrity of the message. After the server generates the DHCPv6 server message, the signing process is started by generating and appending the SA option to the message. The Signature field of the SA option is set to zero before the signing of the message because the signing has not yet occurred. Next, the message with the SA option is signed by using the Ed25519 with the private key. Then, the signature value is inserted into the Signature field of the SA option. Finally, the message is sent to the host, as shown in figure 5.

4.5 *Signature verification process*

The IPv6 host should verify the SA option of the DHCPv6 server messages to prevent a rogue DHCPv6 server attack. The signature verification process starts after the host receives the server message, such as Advertise and Reply messages. First, the host should set the Signature field of the SA option to zero because the message was signed without the signature. Subsequently, the host verifies the signature by using Ed25519 with the public key that is received by the RA message. If the verification succeeds, the host accepts and processes the DHCPv6 server as per RFC 8415 [39]; otherwise, the DHCPv6 server message is discarded, as shown in figure 6.

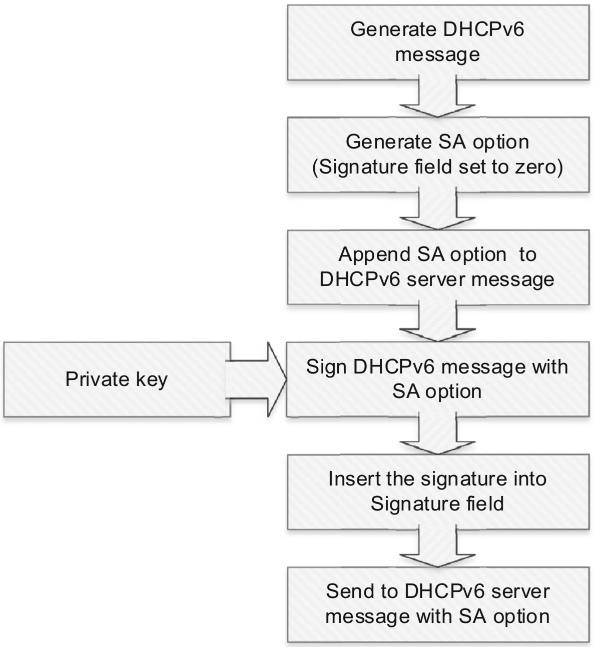

Generate DHCPv6 message
Generate SA option (Signature field set to zero)
Append SA option to DHCPv6 server message
Private key
Sign DHCPv6 message with SA option
Insert the signature into Signature field
Send to DHCPv6 server message with SA option

Figure 5. Signing process.

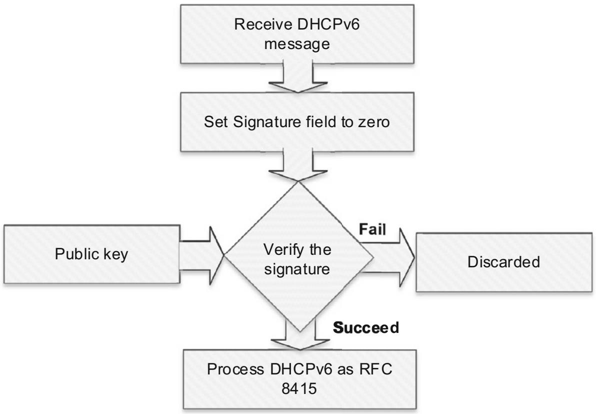


Figure 6. Verification process.

4.6 *Replay attack prevention*

A replay attack is a network attack in which attackers use an old message to send to the victim a message that was previously used in the victim's communication. This old message has a valid signature but with old configuration information, which may lead the IPv6 host to rogue services or result in DoS attack. In order to prevent replay attack, DHCPv6Auth utilizes a transaction ID and replay detection.

The transaction ID is defined by RFC 8415 [39]. It is a unique value associated with the request and reply of DHCPv6 messages. This allows the host to verify if the received server message is a response to the sent message. The transaction ID is generated randomly with the host message, and the reply server message should be set to the same value of the received host message. The host should check if the sending transaction ID equals the receiving transaction ID. If so, the message is processed normally; if not, the message is discarded, as shown in figure 7.

Moreover, the RD field of SA option should be set to a strictly increasing value, based on the current time of the DHCPv6 server. When the host receives a DHCPv6 server message, the host should check if the RD field has a value higher than the previously received RD value. If so, the message can be processed normally; otherwise, the message should be discarded as shown in figure 8.

The verification of a replay attack should be performed before the signature verification process that was described in section 4.4 because the replay verification process is considered faster than signature verification process, and in case the verify replay attack fails, the host is not required to verify the signature.

4.7 *DHCPv6Auth operations*

When a host joins the IPv6 network, it will first multicast an RS message to obtain the network configuration. The router will reply with an RA message to convey the network information and the public key held by the DPK option, as shown in figure 9. By doing so, the DHCPv6 gets the public key that can be used to verify the host.

In the stateful mode, the host multicasts a DHCPv6 Solicit message to all DHCPv6 server messages that exist in the IPv6 link-local network. The DHCPv6 server generates a DHCPv6 Advertise message and signs the DHCPv6 message by using its private key, as illustrated in section 4.4. Subsequently, the server sends a DHCPv6 Advertise message with a SA option. After the host receives the DHCPv6 message, the host first verifies the replay attack, as illustrated in section 4.6, and then verifies the DHCPv6 server message by using the public key that is received by the router, as illustrated in section 4.5. In cases where the host receives a rogue DHCPv6 server message, the host will discard this message as this message will not pass verification. Whenever the server sends a message to the host, the server must sign the message, and the host must verify the server message. If the host uses the Anonymity Profile, the server and host can continue using the DHCPv6Auth because the DHCPv6Auth was designed to do not add any unique id to DHCPv6 host's message. Therefore, the DHCPv6 mechanism can conjunction with Anonymity Profile to prevent a rogue DHCPv6 server attack and provide privacy for the host.

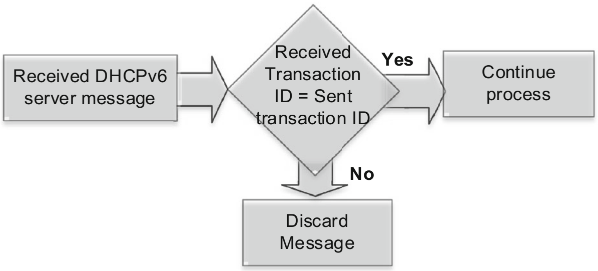

Received DHCPv6 server message
Received Transaction ID = Sent transaction ID
Yes
Continue process
No
Discard Message

Figure 7. Transaction ID verification process.

5. Implementation of the DHCPv6Auth mechanism

The DHCPv6Auth mechanism was implemented using Python language with Open Source DHCPket server software [40] and Open Source library PyNaCl for the Ed25519 [41]. In addition, the IPv6 host was modified to process the DHCPv6Auth mechanism. An RA message was simulated to carry the DPK option. figure 10 illustrates the network topology and device specifications, which consist of a DHCPv6 server, host, router, and an attacker. The attacker ran on Kali Linux, which was used for penetration testing. Wireshark and Scapy tools were used to sniff the server message [42, 43]. Furthermore, rogue RA message was prevented by using RA-Guard [44]. It should be noted that DHCPv6Auth can be deployed in any network by installing DHCPv6Auth on the routers, hosts, and DHCPv6 servers.

6. Experiments and evaluation

The DHCPv6Auth mechanism was evaluated with the Standard DHCPv6, DAM mechanism, and Secure DHCPv6 mechanism. The DHCPv6Auth mechanism was evaluated based on processing time, prevention of a rogue DHCPv6 server attack, and protection of the privacy of the IPv6 host.

6.1 *Processing time*

The aim of this experiment was to measure the total processing time for the Standard DHCPv6, DAM, Secure-DHCPv6, and DHCPv6Auth mechanisms. The total processing time for generating and verifying the DHCPv6 messages, which were Solicit, Advertise, Request, and Reply messages, was calculated. The total processing time (Pt) between host and DHCPv6 server was calculated by subtracting the ending time (Et) with the starting time (St) of the generation process (Gp) and verification process (Vp) for the DHCPv6 messages to obtain the summation of the generation process and verification process for the four DHCPv6 messages, as shown in Eq. (1):

$$PT = \sum_{DHCPv6messages}^{i=0} \left(Et_{(Gp)i} - St_{(Gp)i} + Et_{(Vp)i} - St_{(Vp)i} \right). \tag{1}$$

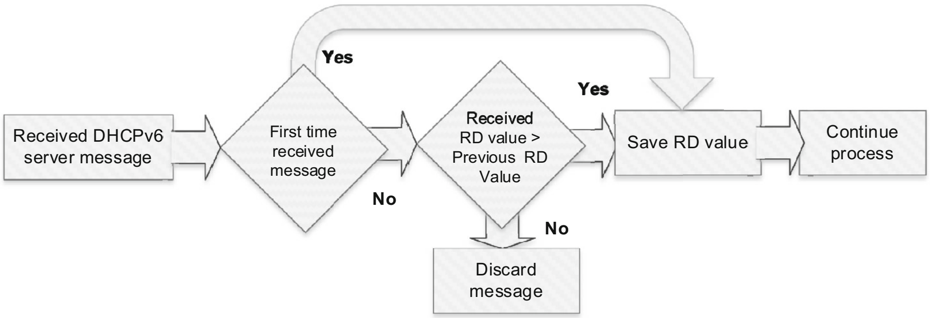

Received DHCPv6 server message
First time received message
Yes
No
Received RD value > Previous RD Value
Yes
No
Save RD value
Continue process
Discard message

Figure 8. Replay attack verification process.

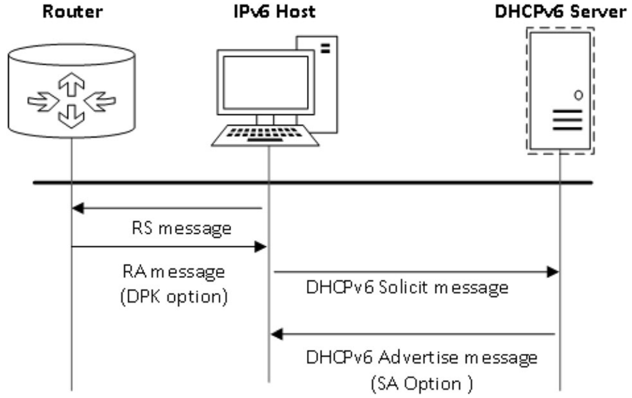
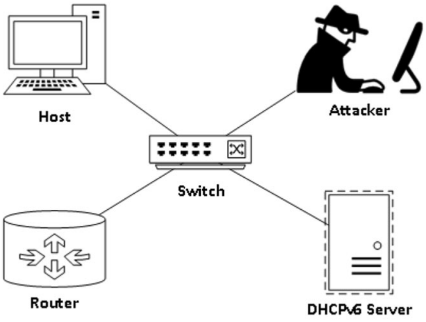


Figure 9. DHCPv6 messages exchange with DHCPv6Auth mechanism mode enabled.



Device Type	Operating System	CPU Type	Memory
DHCPv6 Server	Ubuntu 16.04	Intel Core™ i7-2640M CPU @ 2.80GHz × 4	4 Gb
Client	Ubuntu 16.04	Intel Core™ i7-3770 CPU @ 3.40GHz × 8	8Gb
Attacker	Kali Linux 2018.4	Intel Core™ i7-2640M CPU @ 2.30GHz × 4	4Gb
Router	Ubuntu 16.04	Intel Core™ i7-2640M CPU @ 2.30GHz × 4	4 Gb

Figure 10. Network topology and device specifications.

Moreover, the processing time was affected by other operations of the operating system. Thus, to ensure the reliability of the results, the experiment was repeated 20 times to get the average processing time during the stateful mode. figure 11 shows a line chart for the processing time.

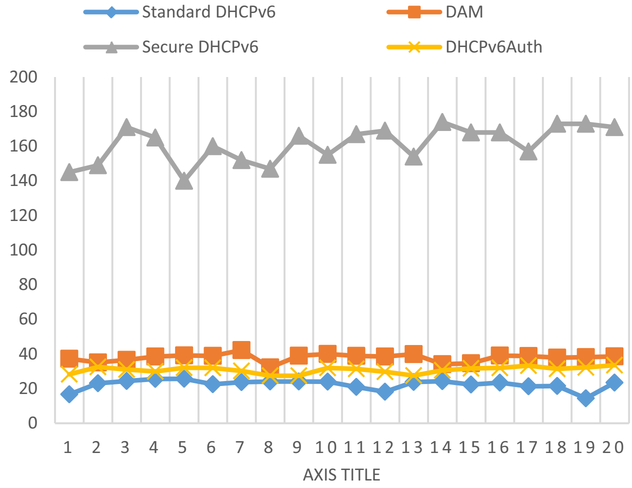


Figure 11. The total processing time of various mechanisms (in milliseconds).

Based on the experiment results, the processing time of DHCPv6Auth was 81% and 19% shorter than that of Secure-DHCPv6 and DAM, respectively. This is because DHCPv6Auth uses Ed25519 to provide authentication. Additionally, the Standard DHCPv6 had a 38% shorter process time than DHCPv6Auth because Standard DHCPv6 does not provide any verification mechanism. Table 1 shows the min, max, mean, standard deviation (SD) and overhead for the total processing time of the DHCPv6 messages. The overhead is calculated by using the Standard DHCPv6 message average processing time as a baseline.

As shown in table 1, the DHCPv6Auth increased the processing time overhead by 38%, whereas DAM and Secure DHCPv6 increased the processing time by 69% and 619%, respectively. The overall results showed that the proposed mechanism clearly reduces the complexity issue and solve the main issue of the distributed key by using the RA message to distribute the public key.

Table 1. Total processing time comparison (in milliseconds).

	Standard DHCPv6	DAM	Secure DHCPv6	DHCPv6Auth
Min	14	32	140	27
Max	26	42	174	34
Mean	22	38	161	31
STD	3	2	11	2
Overhead	Baseline	15	139	8
Overhead (%)	Baseline	69%	619%	38%

6.2 *Traffic overhead*

This section will illustrate the traffic overhead that was generated by the Standard DHCPv6, DAM, Secure-DHCPv6, and DHCPv6Auth mechanisms. The traffic overhead was calculated by measuring the total message size during the DHCPv6 stateful mode to obtain the IPv6 address. The messages which were measured include RA, RS, Solicit, Advertise, Request, and Reply.

Table 2 shows the message size and the traffic overhead of the mechanisms. The traffic overhead of the mechanisms was calculated by subtracting the total message size of the mechanisms (i.e., DAM, Secure DHCPv6, or DHCPv6Auth) from the total message size of Standard DHCPv6. DAM and Secure DHCPv6 had 220 and 1,935 bytes of traffic overhead, respectively. In contrast, the traffic overhead of DHCPv6Auth was only 198 bytes. This is because DAM requires authentication of all DHCPv6 messages as opposed to DHCPv6Auth that only authenticates the server's messages. Moreover, Secure DHCPv6 used RSA DSA for encryption, which adds an additional 256 bytes to the messages, whereas DHCPv6Auth used Ed25519 which adds only 79 bytes.

6.3 *Rogue DHCPv6 server attack*

The aim of this experiment was to measure the ability of different mechanisms to prevent rogue DHCPv6 server attacks. In the experiment, Scapy tool was used to launch a rogue DHCPv6 server attack. The attack was counted as a success if the host configured itself with the information within the rogue DHCPv6 server message; otherwise, the attack was counted as a failure. Further, the attack was attempted five times to ensure the ability of prevention, and attack prevention success rate (APSR) was then measured for all mechanisms. The APSR was calculated by using formula 2:

$$APSR = 1 - s/n \tag{2}$$

where s is the number of successful attack attempts, and n is the total number of attempt, which is 5.

Table 3 shows that the Standard DHCPv6 is vulnerable to rogue DHCPv6 server attack because it does not have a mechanism to prevent the attack. However, DAM, Secure DHCPv6, and DHCPv6Auth successfully prevented rogue DHCPv6 server attack. Thus, this experiment proved that DHCPv6Auth mechanism is able to prevent rogue DHCPv6 server attack.

6.4 *Privacy protection*

The aim of this experiment was to test the ability of proposed authentication mechanisms to provide privacy protection for IPv6 hosts in terms of finding a stable identity to trace the hosts in the network. DHCPv6Auth utilizes the Anonymity Profile mechanism to provide privacy protection. In this experiment, Wireshark was used to monitor the

Table 2. Message size and traffic overhead (in bytes).

Message Name	Standard DHCPv6	DAM	Secure DHCPv6	DHCPv6Auth
RS	70	70	70	70
RA	150	150	150	190
Solicit	145	200	1,499	145
Advertise	148	203	346	227
Request	161	216	346	161
Reply	148	203	346	227
Total	822	1,042	2,757	1,020
Traffic overhead	Baseline	220	1,935	198
Traffic overhead (%)	Baseline	27%	235%	24%

Table 3. Comparison of the rogue DHCPv6 server attack on the various mechanisms.

Mechanism name	APSR	Prevent Rogue DHCPv6 Attack
Standard DHCPv6	0	No
DAM	1	Yes
Secure DHCPv6	1	Yes
DHCPv6Auth	1	Yes

Table 4. Privacy consideration comparison.

Mechanism name	PPSR	Privacy protection
Standard DHCPv6	0	No
DAM	0	No
Secure DHCPv6	0	No
DHCPv6Auth	1	Yes

DHCPv6 traffic. The privacy protection was counted as a success if the DHCPv6 message did not expose any information that could be used to trace hosts in the network; otherwise, the mechanism was counted as a failure to provide privacy protection. Further, five random messages were captured and analyzed to ensure the reliability of the experiment results. The privacy protection success rate (PPSR) was calculated by using Formula (3):

$$PPSR = s/n \tag{3}$$

Where s is the number of successful privacy protections, and n is the total number of attempt, which is 5.

Table 4 shows the results of the Standard DHCPv6, Secure-DHCPv6, and DAM failures to provide privacy protection. These results were expected because Standard DHCPv6 uses multi-unique identifiers, such as DUID and hostname. Furthermore, DAM requires the use of Key ID for the authentication option [13], and Secure-DHCPv6 mechanism uses digital certification to allow the DHCPv6 server to verify the host message [27]. The attackers can use Key ID or the digital certification to trace the hosts in different networks. In contrast, DHCPv6Auth does not add any extra information to the host's message, which makes it impossible for the attacker to correlate the activities with the user.

7. Discussion

The experiment's summary and comparisons are shown in table 5. The experiment result showed that DHCPv6Auth required less processing time and had fewer traffic overheads compared to Secure DHCPv6 and DAM; therefore, it requires fewer hardware resources and less power consumption.

Also, the experiment result showed that the Standard DHCPv6 was vulnerable to rogue DHCPv6 server attack. In contrast, DAM, Secure DHCPv6, and DHCPv6Auth had the ability to prevent rogue DHCPv6 server attack. Although Secure DHCPv6 and DAM prevented a rogue DHCPv6 server attack, these mechanisms are difficult to deploy and manage in a large-scale network and in public networks, such as those in coffee shops and airports, because the keys and digital certification have to be distributed manually. On the other hand, DHCPv6Auth mechanism is easier to deploy as it provides a key distribution mechanism.

Furthermore, the experiment results showed that DHCPv6Auth outperformed Secure DHCPv6 and DAM to provide privacy protection and authentication for the hosts. This is because DHCPv6Auth has been hybridized with an Anonymity Profile to provide privacy protection and authentication.

Table 5 also shows that the Standard DHCPv6 does not have deployment requirement since it does not have any security mechanism to prevent rogue DHCPv6 server attack. In contrast, DAM and Secure DHCPv6 required manual distribution of keys and trust certificates. Also, Secure DHCPv6 mechanism required two extra messages to be exchanged between the host and DHCPv6 server before starting communication, which led to extra processing time. On the other hand, DHCPv6Auth utilized an

Table 5. Experiment summary and comparisons.

Mechanism name	Deployment requirement	Processing time overhead (%)	Traffic overhead (%)	Prevent rogue DHCPv6 server	Privacy protection
Standard DHCPv6	None	Baseline	Baseline	No	No
DAM	Distributing key manually	69%	27%	Yes	No
Secure DHCPv6	Distributing certification manually Exchanged two extra messages	307%	235%	Yes	No
DHCPv6Auth	Securing RA message	51%	24%	Yes	Yes

RA message to distribute the public key to the hosts. The RA message should be secured by a third-party mechanism, as illustrated in section 5. Thus, the deployment and management of DHCPv6Sec are considered easier than that of DAM and Secure DHCPv6. In addition, DHCPv6Auth does not require extra messages, but it does require extra processing time.

8. Conclusion and future work

In this paper, the researchers have analyzed rogue DHCPv6 server attack and explored the privacy issue of DHCPv6 host. Furthermore, a comprehensive review of mechanisms to prevent rogue DHCPv6 server attack and protect the privacy of hosts is presented. This paper also identifies the problems for the current authentication mechanisms that limit these mechanisms' use in IPv6 networks. The problems are that the current authentication mechanisms do not provide a key distribution mechanism, and they also do not protect the privacy of the host.

This paper proposes DHCPv6Auth mechanism to prevent rogue DHCPv6 server attack and to protect the privacy of the host. The DHCPv6Auth mechanism uses digital signature to overcome these problems. DHCPv6Auth is compared with related works in terms of processing time, traffic overhead, preventing rogue DHCPv6 server attack, and protecting privacy.

The results indicate that DHCPv6Auth requires less processing time and incur fewer traffic overheads than other authentication mechanisms. Even though DHCPv6Auth successfully prevents rogue DHCPv6 server attack similar to other authentication mechanisms, it is easier to be deployed in large-scale networks and public networks, such as those used in airports and commercial outlets. The results also indicate that DHCPv6Auth is the only mechanism able to fully protect the privacy of the host.

Future work includes to further develop DHCPv6Auth to become a framework for various digital signatures since the current mechanism only works with Ed25519.

Acknowledgements

This research is supported by Fundamental Research Grant Scheme (FRGS), Ministry of Higher Education, Malaysia No: FRGS/1/2019/ICT03/USM/02/3.

References

[1] Groat S, Dunlop M, Urbanksi W, Marchany R and Tront J 2012 Using an IPv6 moving target defense to protect the smart grid. In: *2012 IEEE PES Innovative Smart Grid Technologies, ISGT 2012*. IEEE, pp 1–7

[2] Internet Society 2018 *State of IPv6 Deployment 2018 | Internet Society*. https://www.internetsociety.org/resources/2018/state-of-ipv6-deployment-2018/

[3] Beeharry J and Nowbutsing B 2016 Forecasting IPv4 exhaustion and IPv6 migration. In: *IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies, EmergiTech 2016*. pp 336–340

[4] Elejla O E, Anbar M and Belaton B 2017 ICMPv6-based DoS and DDoS attacks and defense mechanisms: review *IETE Tech. Rev. (Institution Electron. Telecommun. Eng. India)* 34: 390–407

[5] Ruiz J M V, Cardenas C S and Tapia J L M 2017 Implementation and testing of IPv6 transition mechanisms. In*: IEEE 9th Latin-American Conference on Communications, LATINCOM 2017* vol 2017-January. IEEE, pp 1–6

[6] Yousheng G, Lingyun Y and Lijing H 2018 Addressing scheme based on three-dimensional space over 6LoWPAN for internet of things ICEMI 2017. In: *Proceedings of IEEE 13th International Conference on Electronic Measurement and Instruments* vol. 2018-January. IEEE, pp 59–64

[7] Tirkkonen L 2016 *Utilising configuration management node data for network infrastructure management* (Aalto University)

[8] Dong Wei, Jeremy Kerr, Joseph Shifflett, Samer El-Haj-Mahmoud T H and V M 2013 Dynamic Host Configuration Protocol for IPv6 (DHCPv6). https://www.iana.org/

[9] Brzozowski J and de Velde G Van 2017 *Unique IPv6 Prefix per Host* (RFC Editor)

[10] Horley E and Horley E 2014 IPv6 and DHCP. In: *Practical IPv6 for Windows Administrators* (Apress, Berkeley, CA: Springer), pp 191–207

[11] Sarma S 2014 Securing IPv6's Neighbour and Router Discovery, using Locally Authentication Process. *IOSR J. Comput. Eng.* 16: 22–31

[12] Naidu S 2013 IPv6: threats posed by multicast packets, extension headers and their counter measures. *IOSR J. Comput. Eng.* 15: 66–75

[13] Droms R, Bound J, Volz B, Lemon T, Perkins C and Carney M 2003 *Dynamic host configuration protocol for IPv6 (DHCPv6)* (RFC Editor)

[14] Su Z, Ma H, Zhang X and Zhang B 2011 Secure DHCPv6 that uses RSA authentication integrated with self-certified address. In: *Proceedings - 2011 3rd Int. Work. Cybersp. Saf. Secur.CSS 2011*. pp 39–44

[15] Li L, Jiang S, Cui Y, Jinmei T, Lemon T and Zhang D 2017 *Secure DHCPv6 draft-ietf-dhc-sedhcpv6-21* (RFC Editor)

[16] Gont F and Liu W 2016 *A Method for Generating Semantically Opaque Interface Identifiers (IIDs) with the Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*

[17] Groat S, Dunlop M, Marchany R and Tront J 2010 The privacy implications of stateless IPv6 addressing. In: *ACM International Conference Proceeding Series*. ACM, p 52

[18] Tront J, Groat S, Dunlop M and Marchany R 2011 Security and privacy produced by DHCP unique identifiers. *Proceedings - 16th North-East Asia Symposium on Nano, Information Technology and Reliability, NASNIT 2011*. IEEE, pp 170–9

[19] Krishnan S, Mrugalski T and Jiang S 2016 *Privacy Considerations for DHCPv6* (RFC Editor)

[20] Huitema C, Mrugalski T and Krishnan S 2016 *Anonymity Profiles for DHCP Clients* (RFC Editor)

[21] Kaltio J 2016 *IPv6 in SoHo Environment: A Study of Basic Functionality* (Metropolia Ammattikorkeakoulu)

[22] Kharche M P S and Jawandhiya P M 2016 A case study of IPv4 and IPv6. *National Conference "CONVERGENCE,* p 6

[23] R. Droms W A 2001 *Authentication for DHCP Messages* (RFC Editor)

[24] Droms R 2004 *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6 Status* (RFC Editor)

[25] Shen S, Lee X, Sun Z and Jiang S 2011 Enhance IPv6 dynamic host configuration with cryptographically generated addresses. In: *Proceedings - 2011 5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2011*. pp 487–490

[26] Gont F and Liu W 2016 *DHCPv6-Shield: Protecting Against Rogue DHCPv6 Servers - draft-gont-opsec-dhcpv6-shield-01* (RFC Editor)

[27] Li L, Ren G, Liu Y and Wu J 2018 Secure DHCPv6 mechanism for DHCPv6 security and privacy protection. *Tsinghua Sci. Technol.* 23: 13–21

[28] Alangar V and Swaminathan A 2013 IPv6 security: issue of anonymity. *J. Eng. Comput. Sci.* 2: 2486–2493

[29] Agarwal M, Biswas S and Nandi S 2019 Discrete event system framework for fault diagnosis with measurement inconsistency: Case study of rogue DHCP attack. *IEEE/CAA J. Autom. Sin.* 6: 789–806

[30] Fangfang W, Huazhong W, Dongqing C and Yong P 2013 Substation communication security research based on hybrid encryption of des and RSA In: *Proceedings - 2013 9th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP 2013*. pp 437–441

[31] Rahouma K H 2016 Securing software programs by applying security services with microsoft VB net programming. *Am. J. Inf. Sci. Comput. Eng.* 2: 79–90

[32] Rahouma K H 2017 Reviewing and applying security services with non-english letter coding to secure software applications in light of software trade-offs. *Int. J. Softw. Eng. Comput. Syst.* 3: 71–87

[33] Asaduzzaman A, Gummadi D and Waichal P 2015 A promising parallel algorithm to manage the RSA decryption complexity. In: *Conference Proceedings - IEEE SOUTHEASTCON* vol 2015-June. IEEE, pp 1–5

[34] Dinu D D and Togan M 2014 DHCP server authentication using digital certificates In: *IEEE International Conference on Communications*. IEEE, pp 1–6

[35] Josefsson S and Liusvaara I 2017 *edwards-curve digital signature algorithm (EdDSA)(RFC 8032–*

[36] Podermanski T, Grégr M and Švéda M 2012 Deploying IPv6-practical problems from the campus perspective. In: *Terena Networking Conference*

[37] Atlasis A and Rey E 2015 *IPv6 Router Advertisement Flags, RDNSS and DHCPv6 Conflicting Configurations Operational & Security Implications* (Enno Rey Netzwerke (ERNW) providing Security)

[38] Narten T, Nordmark E and Simpson W 2007 *Neighbor Discovery for IP Version 6 (IPv6)* vol 6

[39] Mrugalski T, Siodelski M, Volz B, Yourtchenko A, Richardson M, Jiang S, Lemon T and Winters T 2018 *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)* (RFC Editor)

[40] Steffann S 2018 DHCPKit *github*

[41] Paul K and Heidelberger V 2018 PyNaCl: Python binding to the libsodium library *Github*

[42] Montante R 2018 Using scapy in teaching network header formats: Programmingnetwork headers for non-programmers, In: *Proceedings of the 49th ACM TechnicalSymposium on Computer Science Education*. ACM, p 1106

[43] Sanders C 2017 *Practical packet analysis: Using Wireshark to solve real-world network problems* (No Starch Press)

[44] Gont F 2014 *Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard)* (RFC Editor)