WPA3 Connection Deprivation Attacks

Karim Lounis(✉) and Mohammad Zulkernine

Queen's Reliable Software Technology Lab, School of Computing,
Queen's University, Kingston, ON, Canada
{lounis,mzulker}@cs.queensu.ca

Abstract. After the KRACK (Key Reinstallation AttaCK) attack on WPA2 (Wi-Fi Protected Access 2) in Fall 2017, the Wi-Fi Alliance started developing WPA3 which was announced in Summer 2018. WPA3 is a certification that adds protection mechanisms to its predecessor WPA2, such as dictionary attack resistance, management frame protection, and forward secrecy. In April 2019, researchers discovered a set of vulnerabilities in WPA3. These vulnerabilities allow an attacker to perform different types of attacks, varying from denial of service to network-password cracking. This has worried the community including organizations and device vendors who have already started implementing WPA3 on their devices. In this paper, we present three possible denial of service attacks on WPA3. We start by presenting the WPA3-SAE (Simultaneous Authentication of Equals) mechanism. Then, we analyze the mechanism and show the existence of specification flaws in WPA3 protocol. An attacker exploits these flaws to generate attacks on Wi-Fi availability to deprive legitimate devices from connecting to WPA3 networks. We experimentally show the feasibility of these attacks and propose possible countermeasures to mitigate the attacks and direct device vendors to better implement security in their future devices.

Keywords: Wi-Fi security · WPA3-SAE · WPA3 security · Wi-Fi attacks

1 Introduction

Wi-Fi technology has provided a number of security mechanisms to guarantee security services, i.e., authentication, confidentiality, integrity, and availability. The first mechanism, ratified in 1999, was WEP (Wired Equivalent Privacy). It applies the RC4 (Ron's Code 4) stream cipher algorithm along with an encryption key and uses the CRC-32 (Cyclic redundancy check 32-bit) algorithm to generate a code for data integrity. Few years later, WEP was completely broken. Serious vulnerabilities were found and the security of WEP became nonsense [1–5]. The IEEE (Institute of Electrical and Electronics Engineers) started proposing the 802.11i framework [6] which promised stronger security mechanisms for authentication, encryption, and data integrity. Due to pressure from the market, the Wi-Fi Alliance rashly (in April 2003) started certifying devices

S. Kallel et al. (Eds.): CRiSIS 2019, LNCS 12026, pp. 164–176, 2020.
https://doi.org/10.1007/978-3-030-41568-6_11

based on a draft version of 802.11i under the name of WPA (Wi-Fi Protected Access). WPA uses the TKIP (Temporal Key Integrity Protocol) encryption that adopts RC4 with longer keys and the Michael algorithm for data integrity. Finally, in June 2004, the final version implementing the 802.11i specification was ratified under the name of WPA2 (Wi-Fi Protected Access 2). WPA2 uses the CCMP (CTR with CBC-MAC Protocol) encryption mechanism that adopts AES (Advanced Encryption Standard) for encryption and AES-CBC-MAC (Cipher Bloc Chaining-Message Authentication Code) algorithm for data integrity.

Despite some security vulnerabilities, mostly related to denial of service attacks [7–12], discovered on WPA2 during the last decade, the protection mechanism has provided an acceptable security level. The community has since then believed that WPA2 is the most secure mechanism that is nowadays available in the market. Nonetheless, in Fall 2017, researchers demonstrated an attack, known as KRACK (Key Reinstallation AttaCK) [13], which has broken the WPA2 secure-assumption. This has pushed the Wi-Fi Alliance to come up with a new security mechanism called WPA3, which was then announced in Summer 2018. Thus far, a 7-page specification document on WPA3 is available on the Wi-Fi Alliance website [14]. Device manufacturers are still implementing the protocol to commercialize Wi-Fi devices that are WPA3-certified. This has not prevented researchers from discovering a set of vulnerabilities, known as Dragonblood, in the WPA3 authentication protocol [15]. These vulnerabilities allow an attacker to perform different types of attacks, varying from denial of service to cracking the network password, using downgrading and side-channel attacks. Also, as part of our research, we have reported other vulnerabilities on WPA3 that can be exploited to generate denial of service attacks on WPA3 [16].

WPA3 applies SAE (Simultaneous Authentication of Equals), also known as Dragonfly [17], on top of the classical 4-way-handshake, for authentication and key establishment. The SAE key establishment protocol uses elliptic curve cryptography along with a Deffie-Hellman key exchange style and the shared network password, to allow two communication parties to establish a shared key. This key is known by PMK (Pairwise Master Key). It is employed further in the classical 4-way-handshake to derive other cryptographic keys, such as the encryption and message integrity keys. Thus, the hardness of this protocol relies on the difficulty of solving the ECDLP (Elliptic Curve Discrete Logarithm Problem)[1]. Also, WPA3 requires the use of MFP (Management Frame Protection)[2] mechanism to protect Wi-Fi users from being victims to denial of service attacks that are based on management frame spoofing, such as deauthentication attack [8].

In this paper, we present three potential connection deprivation attacks on WPA3. We start by presenting the WPA3-SAE authentication mechanism. Then,

[1] ECDLP is the problem of finding a scalar n given two elliptic points $P \in \xi(\mathbb{F}_p)$ and $Q \in \xi(\mathbb{F}_p)$ such that Q is the product of the scalar n by the point P ($Q = n.P$), where ξ is an elliptic curve defined over a finite field $\mathbb{F}_p$ and $p = q^m$ (q is prime) [18].

[2] MFP (Management Frame Protection) was introduced as part of the IEEE 802.11w amendment to add protection to management frames that are originally not authenticated and hence can be easily spoofed for denial of service attacks.

we analyze the mechanism and show the existence of specification flaws. An attacker can exploit these flaws to generate attacks on Wi-Fi availability and deprive legitimate Wi-Fi devices from connecting to WPA3 networks. We demonstrate the feasibility of these attacks and propose possible countermeasures to mitigate them.

The remainder of the paper is organized as follows: Sect. 2 presents WPA3 and its authentication mechanism. In Sect. 3, we present three possible attacks that abuse WPA3 authentication to deprive Wi-Fi users from connecting to WPA3 Wi-Fi networks. We show their practical feasibility and discuss possible countermeasures to mitigate them. Finally, Sect. 4 concludes the paper.

2 WPA3 Authentication Phases

WPA3 allows three possible operational modes. WPA3-SAE (Wi-Fi Protected Access-Simultaneous Authentication of Equals) is used when Wi-Fi devices only support WPA3. WPA3-SAE transition, also known as mixed mode, allows Wi-Fi devices that only support WPA2 to connect to a WPA3 network. WPA3-Enterprise 192-bit is used in sensitive enterprise environments, such as government and industrial networks. In the remaining part of this paper, we consider the WPA3-SAE and WPA3-SAE transition modes. In the following paragraphs, we present how authentication is performed in WPA3-SAE.

The WPA3 authentication consists of three phases: (1) The SAE (Simultaneous Authentication of Equals) handshake. (2) The association phase. (3) The 4-way handshake phase, as illustrated in the MSC[3] of Fig. 1. The first phase is also known as Dragonfly. It consists of four messages in which the supplicant[4] and the access point (authenticator) use the shared network password to derive the shared key PMK (Pairwise Master Key). This phase is illustrated with much details in the MSC(see footnote 3) of Fig. 2 and discussed in the next paragraph. The second phase consists of two messages, where the supplicant sends an association request and the access point replies back by an association response. During this phase, the supplicant indicates in the association request which security parameters (i.e., authentication, encryption, and authentication key management algorithms) it wishes to use. The access point confirms or rejects the parameters in the association response message. Finally, in the last phase, both parties use the previously derived PMK key to execute the classical 4-way handshake to derive and install the PTK (Pairwise Transient Key), which is the session key.

[3] MSC (Message Sequence Chart) is a graphical language for the description of the interaction between different components of a system. This language is standardized by the ITU (International Telecommunication Union).

[4] In 802.1X terminology, Wi-Fi users are called supplicants. They authenticate themselves to the access point, which is known by the authenticator. In the rest of the paper, we use the term Wi-Fi supplicant and Wi-Fi user interchangeably. We also use the term Wi-Fi access point and Wi-Fi authenticator interchangeably.

During the SAE-handshake phase shown in Fig. 2 (Phase 1 in Fig. 1), both parties, i.e., the supplicant and the access point, agree on a cryptographic domain, ECP (Elliptic Curve groups) or MODP (Modular Exponential groups). Depending on the cryptographic domain, they use the shared network password along with a hash-to curve or hash-to-group algorithm to transform the password into an elliptic curve point P (when ECP is used) or into a multiplicative group modulo a prime element (when MODP is used). In both cases, the output is denoted by PWE (PassWord Element). In the remaining part of this paper, we only consider the ECP domain for WPA3-SAE description and experimentation.

Considering the ECP domain, each party $i \in \{S, A\}$ generates two random values, $rand_i$ and $mask_i$. These two random values are used to compute two commit values, $scal_i = (rand_i + mask_i)\ [r]$ and $elem_i = Inv(mask_i \bullet PWE)$. The value $scal_i$ is a scalar, whereas $elem_i$ is an elliptic curve point which corresponds to the inverse (Inv) of the point that results from the elliptic point multiplication '$\bullet$' of the scalar $mask_i$ by the elliptic point PWE. Once computed, each party sends to the other one an authentication message with an authentication sequence number set to `0x0001`[5]. This message is also known as the commit message and contains the tuple ($scal_i$, $elem_i$).

Upon receiving the tuple, each party verifies whether the values of $scal_i$ and $elem_i$ are within the curve definition domain or not, i.e., $scal_i \in [1, r[$ and $elem_i \in \xi(\mathbb{F}_p)$, where r is the prime order of the generator G of the finite cyclic group $\mathbb{G}$ that defines the addition operation in the elliptic curve ξ. The used elliptic curve ξ is defined over the prime finite filed $\mathbb{F}_p$ of order p (large prime number). Once verified, both parties compute a token tok. This token is the result of applying a HMAC (Keyed-Hash Message Authentication Code) over five concatenated elements (the concatenation is denoted by '|' in Fig. 2). For $i, j \in \{S, A\}$ and $i \neq j$, the first element is $F(rand_i \bullet (scal_j \bullet PWE \diamond elem_j))$, where F is a hash function and '$\diamond$' the elliptic curve point addition operator. The second element is $F(elem_i)$, the third is the scalar $scal_i \in [1, r[$, the fourth is $F(elem_j)$, and the fifth is the scalar $scal_j \in [1, r[$. Each party $i \in \{S, A\}$ sends its token tok_i to the other party in an authentication message with an authentication sequence number set to `0x0002`(see footnote 5). This message is also known as the commit message. Each party verifies the correct derivation of the token by the other party. A token constitutes a proof of knowledge of the password for a given party. If both tokens are validated, the SAE-handshake succeeds and both parties use the value $F(rand_i \bullet (scal_j \bullet PWE \diamond elem_j))$ as the shared PMK, which is used as a seed in the last phase to perform the 4-way-handshake.

[5] In the IEEE 802.11 standard, the authentication sequence number indicates the type of the authentication frame: `0x0001` is used to indicate an authentication request frame, whereas `0x0002` is used to indicate an authentication response frame.

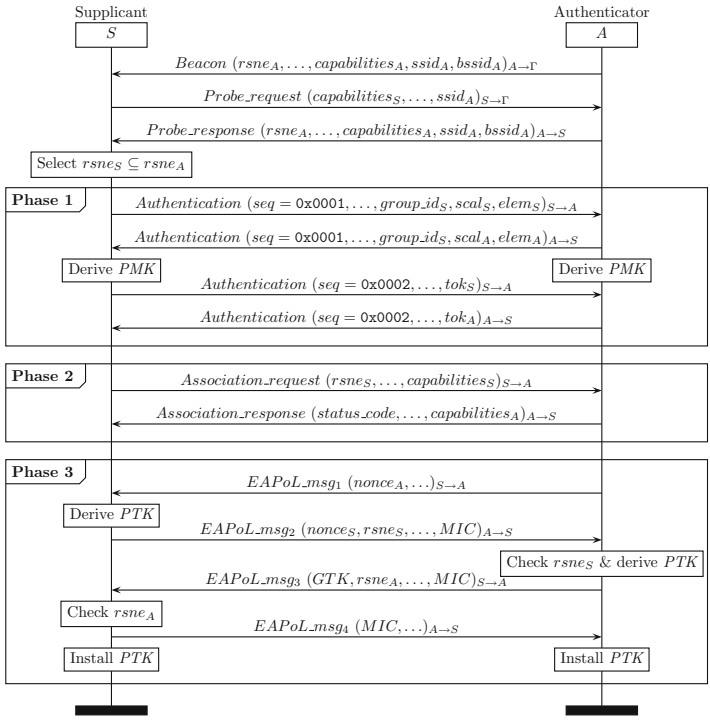

Fig. 1. WPA3-SAE authentication mechanism, where **Phase 1** is the SAE-handshake phase, **Phase 2** is the association phase, and **Phase 3** is the WPA2-4-way-handshake phase. The notation $M_{x \to y}$ indicates a message M sent from x to y. Also, E_x indicates an element E that is generated by $x \in \{S, A\}$. For $y = \Gamma$, the destination is set to the broadcast MAC address (i.e., `FF:FF:FF:FF:FF:FF`).

3 Connection Deprivation Attacks on WPA3-SAE

In the following subsections, we present three connection deprivation attacks on WPA3: (1) Attack on the 4-way-handshake downgrade protection. (2) Attack on SAE-handshake commit values. (3) Attack on the group/curve negotiation. The three attacks exploit specification flaws in the WPA3-SAE handshake to deprive Wi-Fi supplicants from connecting and joining WPA3 networks. We describe each attack individually, show its practical implementation, and provide countermeasures as well. As the same countermeasure can be applied to mitigate the three attacks, we discuss the countermeasure in Subsect. 3.5. First, we describe the environment used to generate all three attacks as follows.

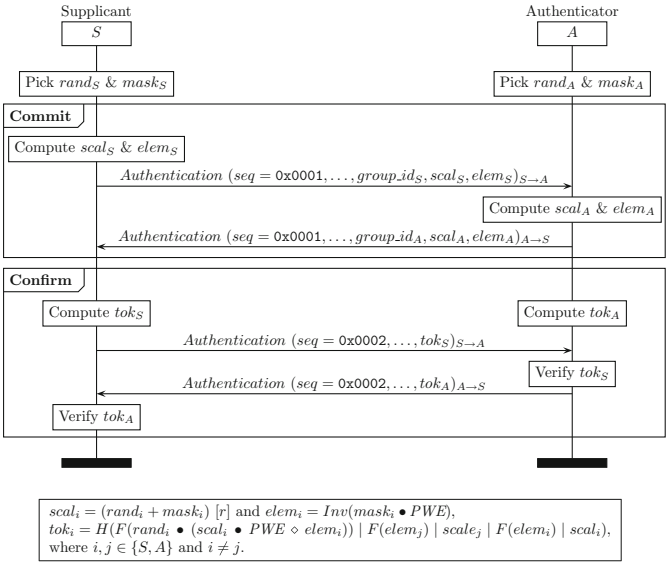

Fig. 2. Simultaneous Authentication of Equals Handshake, also known as Dragonfly. The notation $M_{x \to y}$ indicates a message M sent from x to y. Also, E_x indicates an element E that is generated by $x \in \{S, A\}$.

3.1 Attack Environment

To put the previous attacks into practice, we have used two Raspberry Pis B3+ and one laptop. The first Raspberry Pi runs *hostapd-2.7*[6] Linux utility (on Raspbian OS) to emulate a WPA3-SAE access point. The second Raspberry Pi runs *wpa_supplicant-2.7*[7] Linux utility (on Ubuntu MATE) to emulate a WPA3-SAE supplicant. The access point is configured to use WPA3 with SAE key management algorithm and AES-CCMP for encryption. It operates on channel 6 with an SSID set to `QRST_WPA3`. We have also augmented the two Raspberry Pis with a Wi-Fi interface (ODROID Wi-Fi Module 4) as the built in Wi-Fi network card does not support the WPA3-SAE as well as the monitor mode. We have also configured the supplicant with the correct network settings to be able to connect to the access point. We have run the access point and then run the supplicant which successfully got authenticated and associated to the access point. As the Wi-Fi network interfaces that were in our possession do not support MFP (Management Frame Protection), we have not enabled this option. Although MFP is mandatory in WPA3, enabling or disabling it does not affect

[6] *hostapd-2.7* is an open source package that allows to emulate access points on a computer. The version 2.7 supports the use of WPA3-PSK authentication protocol. It can be downloaded from https://w1.fi/releases/hostapd-2.7.tar.gz.

[7] *wpa_supplicant-2.7* is an open source package that allows to implement Wi-Fi supplicant on a computer. The version 2.7 supports the use of WPA3-PSK. It can be downloaded from https://w1.fi/releases/wpa_supplicant-2.7.tar.gz.

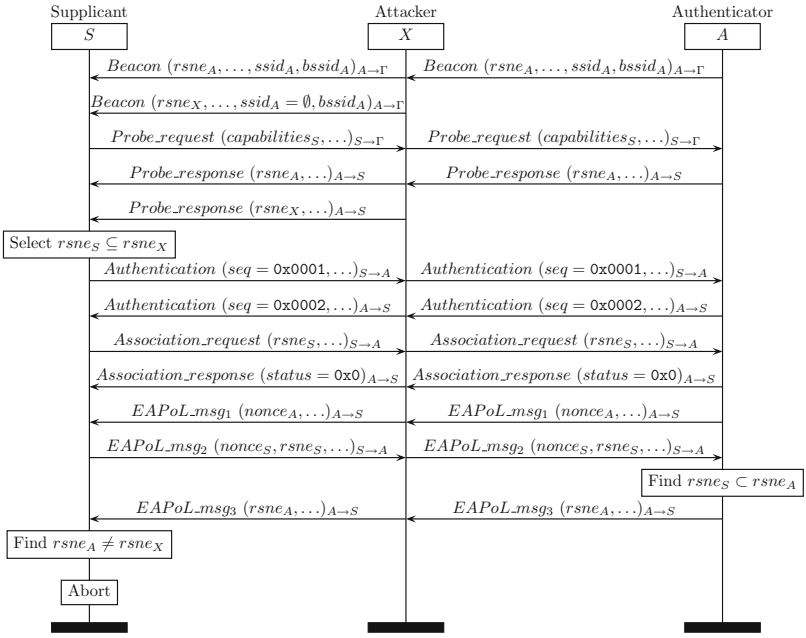

Fig. 3. Man-in-the-middle attack on WPA3-SAE downgrade protection. The notation $M_{x \to y}$ indicates a message M sent from x to y. Also, E_x indicates an element E that is generated by $x \in \{S, A\}$. For $y = \Gamma$, the destination is set to the broadcast MAC address (i.e., `FF:FF:FF:FF:FF:FF`).

the discussed attacks. In fact, it is infeasible to protect the management frames that are sent before the 4-way handshake (those are sent prior key establishment) and hence the discussed attacks are still feasible. Finally, the attacker uses the laptop (HP Probook 6560b) that runs *hostapd-2.7* on Linux (Ubuntu 16.04 LTS) to emulate an evil twin of the legitimate access point. We have set the attacker's security mechanism to be WPA2-PSK or WPA3-SAE depending on the attack scenario and set its SSID to be `QRST_WPA3` and hidden. This allows the attacker to be as passive as possible. In fact, only one SSID=`QRST_WPA3` will appear on the supplicant's device screen when scanning for Wi-Fi networks.

3.2 Attack on the 4-Way Handshake Downgrade Protection

Observation. In a Wi-Fi network that adopts the infrastructure mode, the access point periodically broadcasts management frames called beacons. These beacons reveal information about the network settings, such as synchronization information, BSSID (Basic Service Set IDentifier), SSID (Service Set IDentifier), and security information. The security information are revealed in an elementary structure called RSNE (Robust Security Network Element), which informs Wi-Fi supplicants that are interested in connecting to the network, about the supported security mechanisms (in a cipher-suite). The cipher-suite indicates which

authentication, encryption, and authentication key management algorithms are supported by the access point. Wi-Fi supplicants can then choose the highest security mechanism that they can support from the received cipher-suite.

In the WPA3 authentication, the supplicant and the access point go through three phases as illustrated in the MSC(see footnote 3) of Fig. 1. Specifically, during the 4-way-handshake, both the supplicant and access point verify whether the RSNE that the other party wishes to use is still the same and has not been modified by a third party. The access point checks whether the $rsne_S$ of the supplicant is supported. The supplicant however, checks whether the indicated RSNE in the beacon frames and probe responses (i.e., $rsne_A$) is still the same. If the supplicant detects an RSNE mismatch, it passively aborts the handshake. In case of the access point, the latter sends a rejection message that could be a deauthentication frame. In a nutshell, this prevents an attacker from spoofing beacon or probe response frames and announcing weaker RSNE to trick supplicants into choosing a weaker cipher-suite rather than a more secure one [11].

Attack Generation. An attacker exploits this abortion behavior and sets up a MITM (Man In The Middle) attack as illustrated in the MSC(see footnote 3) of Fig. 3. By spoofing the legitimate access point and broadcasting beacon frames that announce weaker cipher-suite, such as WPA2-PSK ($rsne_X$ in Fig. 3) instead of WPA3-SAE ($rsne_A$ in Fig. 3). The supplicant may choose WPA2-PSK over WPA3-SAE and start the authentication with the legitimate access point. At this point, the attacker stays idle and watches the scene. The supplicant will detect that the access point is actually supporting WPA3 in addition to WPA2 when it receives the third message of the 4-way handshake (viz., last message in Fig. 3). The supplicant detects a mismatch and aborts the connection. The attacker repeats this scenario again and again to deprive the supplicant from connecting to the Wi-Fi network. The attacker just has to send beacon frames at a higher rate[8] and rapidly reply to supplicant's probe requests.

To experiment the attack on the 4-way-handshake downgrade protection, we have configured the WPA3 supplicant in a way so that it can connect to WPA2-PSK or WPA3-SAE access points. We have configured the attacker access point to behave as an evil twin of the legitimate access point but using WPA2-PSK instead of WPA3-SAE. We have started both access points and then executed the supplicant. We have observed (using *Wireshark*) that the supplicant has chosen to operate the WPA2-PSK instead of WPA3-SAE. In fact, after receiving probe responses from both access points (indicating the supported cipher-suite in the RSNE), the supplicant has replied back by sending an authentication frame (seq=`0x0001`) indicating the authentication algorithm `0x0`, i.e., Open System, to be used. Interestingly, both access points have replied with an authentication frame (seq=`0x0002`). The supplicant has proceeded by sending an association request in which it has indicated the selected RSNE (i.e., WPA-PSK-CCMP). The legitimate access point replied first by sending an association response indicating a rejection message with a status code `0x002b`. This code carries the

[8] Typically, beacons are sent every 100 time units (beacon interval), where a time unit is 1.024 ms. The attacker can change the beacon interval to be 15 instead of 100.

message "Invalid AKMP", which indicates invalid authentication key management protocol. The supplicant has also received the association response from the attacker (with success massage `0x0000`), but it was ignored as the supplicant has already aborted the authentication right after the rejection. We can see that an attacker can easily trick a supplicant into choosing WPA2-PSK instead of WPA3-SAE, which is the first goal in this attack. We were able to repeat this attack scenario and deprive the legitimate supplicant from connecting to the right access point. Even if the legitimate access point was configured to advertise the capability of operating both WPA2-PSK and WPA3-SAE (which is not possible in *hostapd-2.7*), the authentication would have happened through the Open System authentication. Then, during the 4-way-handshake, in particular, after receiving the third EAPoL[9] message, the supplicant would have aborted the authentication due to RSNE mismatch and have had restarted the authentication again.

3.3 Attack on WPA3-SAE's Commit Values

Observation. As described in Sect. 2, the WPA3-SAE handshake runs through two subphases (viz., Fig. 2): commit and confirm. Specifically, during the first subphase both the supplicant and the authenticator generate a tuple ($scal_i$, $elem_i$) and send it to the other party using a commit message (an authentication message with seq=`0x0001`). Each party $i \in \{S, A\}$ verifies whether the tuple ($scal_{j\neq i}$, $elem_{j\neq i}$) contains values that are within a predefined range. If one of the parties finds out that the received tuple is out of the predefined range, i.e., $scal_{j\neq i} \notin [1, r[$ or $elem_{j\neq i} \notin \xi(\mathbb{F}_p)$, the handshake is aborted.

Attack Generation. An attacker exploits this value-range checking operation to cause a denial of service on the supplicant. When the supplicant sends its first commit message (containing commit values $scal_S$ and $elem_S$), the attacker in a race condition with the legitimate authenticator, replies with a rejection message as if the generated commit values ($scal_S$ and $elem_S$) were out of the range. The attacker just has to spoof the authenticator and reply first to the supplicant with a crafted commit message that carries an "out of range" error information. The supplicant receives the latter message and aborts the handshake as illustrated in the MSC(see footnote 3) of Fig. 4 (consider $m_2 = m_4$ and $error\_code_1$ to be "Invalid commit values"). The attacker performs this injection repeatedly, at specific instants of time, and prevents legitimate supplicants from connecting to the network.

To implement this attack, we have modified the source code of *hostapd-2.7*[10] in such as way so that the attacker's access point replies to the supplicant's commit message with a commit message that contains the rejection status code `0x0001` (stating "Unspecified failure"). Next, we have run the two access points followed by the supplicant. We have observed that the supplicant has sent its

[9] EAPoL (Extensible Authentication Protocol over LAN) is a network protocol used in 802.1X for authentication. It uses EAP protocol over Ethernet.

[10] We have modified the code located in `/hostapd-2.7/src/ap/ieee802_11.c`.

commit message and then received a first commit message from the attacker access point. As the message contained the rejection message, the supplicant has straightforwardly aborted the authentication. Although it has received the second commit message from the legitimate access point, the latter message got ignored. The legitimate access point has re-transmitted the commit message many times before aborting the authentication process.

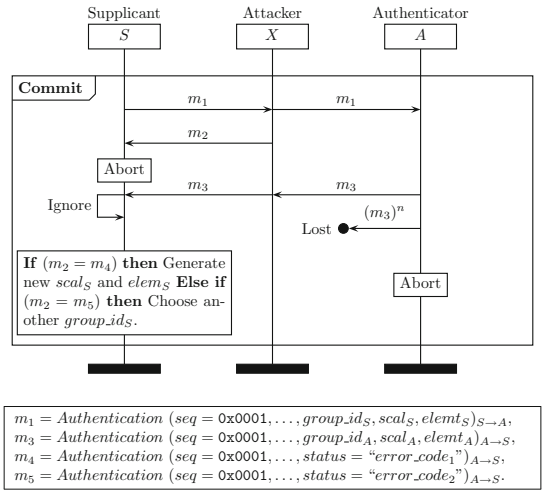

Fig. 4. Man-in-the-middle attack on WPA3-SAE commit subphase: in the case where $m_2 = m_4$, this MSC depicts the commit values attack. Otherwise, when $m_2 = m_5$ this MSC depicts the group/curve negotiation attack. The notation $M_{x \to y}$ indicates a message M sent from x to y. Also, E_x indicates an element E that is generated by $x \in \{S, A\}$ and $(m_{v \in \mathbb{N}})^n$ a message $m_{v \in \mathbb{N}}$ sent $n \in \mathbb{N}$ times.

3.4 Attack on WPA3-SAE's Group/Curve Negotiation

Observation. During the SAE handshake, the supplicant sends to the authenticator a commit message indicating which elliptic curve or multiplicative group (denoted by $group\_id_S$) it wishes to use along with the tuple ($scal_S$, $elem_S$). If the authenticator does not support the desired elliptic curve or multiplicative group, it sends a commit message to the supplicant to inform it that the access point does not support the desired multiplicative group or elliptic curve group.

Attack Generation. The attacker exploits this protocol behavior and sets up a MITM attack between the supplicant and the authenticator. It waits for a supplicant to send a commit message and quickly replies with a forged commit message informing the supplicant that the authenticator does not support the desired $group\_id_S$ before the supplicant receives the commit message from the

legitimate authenticator. The attacker repeats this attack each time the supplicant proposes whatever cryptographic option and prevents the supplicant from connecting to the network as illustrated in the MSC(see footnote 3) of Fig. 4 (consider $m_2 = m_5$ and $error\_code_2$ to be "Unsupported Deffie-Hellman-group").

To implement this attack, we have modified the same file (i.e., `ieee802_11.c`) in such a way so that the attacker's access point replies to commit messages with a rejection message that contains the status code `0x004d` (stating "Authentication is rejected because the offered finite cyclic group is not supported"). We have run the attack and have observed that each time the supplicant tried to initiate the authentication, it receives the rejection message from the attacker's access point first. We have run this attack during 30 min and have observed that the supplicant has performed 23 authentication attempts and all of them failed. The supplicant has tried to authenticate using the ECP finite cyclic groups 19, 20, 21, 25, and 26 (then repeating from 19) and have failed in each of them due to the attack. In this way, we have successfully managed to deprive the supplicant from getting connected to the right access point.

3.5 Countermeasure

The three attacks discussed in Subsects. 3.2, 3.3 and 3.4 have a common vulnerability nature, which consists of foolishly replying back to messages upon their reception. In the attack on the 4-way-handshake downgrade protection (discussed in Subsect. 3.2), the supplicant has received a probe response from the attacker and then has taken a straight decision to apply the security mechanism that the attacker proposes based on the information contained inside the received message (i.e., RSNE in probe response) and the supplicant's local network configurations. In the case of the attack on the commit values (discussed in Subsect. 3.3) and the attack on the group/curve negotiation (discussed in Subsect. 3.4), the supplicant has taken the decision to abort the handshake upon the reception of the first "rejection" message. Overall, the behavior taken by the supplicant is coherent with the specification. However, we believe that future supplicants should be smarter. Instead of taking a decision based on one message, supplicants should take a decision based on a group of quasi-similar messages. Therefore, to mitigate the previous three attacks, the supplicants should be implemented in such a way so that they do not take a decision and reply upon the reception of the first probe, authentication, or association response. The supplicant would take a longer time than usual but with a guarantee of not being fooled. In this way, we can prevent these attacks from occurring in future Wi-Fi networks that will be supporting WPA3.

4 Conclusion

The WPA3 has been recently announced as the next generation of Wi-Fi security. This new security standard promises higher security and aims to completely replace the WPA2 mechanism. In April 2019, WPA3 has been shown to contain

some vulnerabilities that could affect the availability of the Wi-Fi network and the security of the whole network by cracking the network password. In this paper, we have analyzed the WPA3-SAE authentication protocol and presented three possible attacks. These attacks aim to affect Wi-Fi network availability by depriving legitimate users from connecting to the network. We have shown the practical feasibility of these attacks and proposed countermeasures to mitigate the attacks. We claim that if the next generation access points (i.e., those implementing WPA3) do not apply the countermeasures and follow the recommendations, the presented attacks will still be possible.

Acknowledgment. This work is partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canada Research Chairs (CRC).

References

1. Tews, E., Beck, M.: Practical attacks against WEP and WPA. In: Proceedings of the Second ACM Conference on Wireless Network Security, pp. 79–86 (2009)
2. AlFardan, N., Bernstein, D.J., Paterson, K.G., Poettering, B., Schuldt, J.C.N.: On the security of RC4 in TLS. In: Presented as part of the 22nd USENIX Security Symposium, pp. 305–320. USENIX (2013)
3. Stubblefield, A., Ioannidis, J., Rubin, A.D.: Using the Fluhrer, Mantin, and Shamir attack to break WEP. In: Proceedings of the Network and Distributed System Security Symposium (2002)
4. Fluhrer, S., Mantin, I., Shamir, A.: Weaknesses in the key scheduling algorithm of RC4. In: Vaudenay, S., Youssef, A.M. (eds.) SAC 2001. LNCS, vol. 2259, pp. 1–24. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45537-X_1
5. Borisov, N., Goldberg, I., Wagner, D.: Intercepting mobile communications: the insecurity of 802.11. In: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, pp. 180–189. ACM (2001)
6. IEEE: "IEEE STD 802.11i" amendment 6: medium access control security enhancement (2004)
7. Paterson, K.G., Poettering, B., Schuldt, J.C.N.: Plaintext recovery attacks against WPA/TKIP. In: Cid, C., Rechberger, C. (eds.) FSE 2014. LNCS, vol. 8540, pp. 325–349. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46706-0_17
8. Bellardo, J., Savage, S.: 802.11 denial-of-service attacks: real vulnerabilities and practical solutions. In: Proceedings of the 12th Conference on USENIX Security Symposium, vol. 12, pp. 15–27. USENIX Association (2003)
9. Alabdulatif, A., Ma, X., Nolle, L.: Analysing and attacking the 4-way handshake of IEEE 802.11i Standard. In: The IEEE 8th International Conference for Internet Technology and Secured Transactions, pp. 382–387 (2013)
10. Singh, R., Sharma, T.P.: On the IEEE 802.11i security: denial-of-service perspective. In: Security and Communication Networks, pp. 1378–1407 (2014)
11. Vanhoef, M., Piessens, F.: Denial-of-service attacks against the 4-way Wi-Fi handshake (2017). https://papers.mathyvanhoef.com/ncs2017.pdf
12. Bai, Z., Bai, Y.: 4-way handshake solutions to avoid denial of service attack in ultra wideband networks. In: The 3rd International Symposium on Intelligent Information Technology Application, vol. 3, pp. 232–235 (2009)

13. Vanhoef, M., Piessens, F.: Key reinstallation attacks: forcing nonce reuse in WPA2. In: The Proceedings of the ACM Conference on Computer and Communications Security, pp. 1313–1328 (2017)
14. Wi-Fi-Alliance. WPA3 specification version 1.0 (2018). https://www.wi-fi.org/
15. Vanhoef, M., Ronen, E.: Dragonblood: a security analysis of WPA3's SAE handshake, April 2019. https://papers.mathyvanhoef.com/dragonblood.pdf
16. Lounis, K., Zulkernine, M.: Bad-token: a denial of service attack on WPA3. In: Proceedings of the 12th International Conference on Security of Information and Networks, Sochi, Russia, 12–15 September 2019
17. Harkins, D.: Simultaneous authentication of equals: a secure, password-based key exchange for mesh networks. In: Second International Conference on Sensor Technologies and Applications, pp. 839–844 (2008)
18. Hankerson, D., Menezes, A., Vanstone, S.: Guide to Elliptic Curve Cryptography. Springer, New York (2004). https://doi.org/10.1007/b97644