Fragile Frames: Wi-Fi's Fraught Fight Against FragAttacks

Siebe Devroe
DistriNet, KU Leuven
Leuven, Belgium
siebe.devroe@student.kuleuven.be

Héloïse Gollier
DistriNet, KU Leuven
Leuven, Belgium
heloise.gollier@kuleuven.be

Mathy Vanhoef
DistriNet, KU Leuven
Leuven, Belgium
Mathy.Vanhoef@kuleuven.be

Abstract

In 2021, researchers disclosed vulnerabilities in the IEEE 802.11 standard related to frame fragmentation and aggregation, also known as the FragAttacks. In this paper, we design novel methods to measure whether real-world Wi-Fi networks are still affected by these vulnerabilities. Using our methods, we conducted surveys in three cities at two points in time (2023 and 2025) and found many networks still vulnerable. Concretely, we detected 52 691 networks, found that in one city, 30% are still affected by one of the FragAttacks, and that for some ISPs, nearly all their routers are still affected. Motivated by this, we also present a design flaw in the 802.11 standard's defense against one of these vulnerabilities.

CCS Concepts

• **Security and privacy** → **Mobile and wireless security**; • **Networks** → *Protocol testing and verification.*

Keywords

IEEE 802.11, survey, wardrive, 802.11s, mesh, fragment, forge

ACM Reference Format:
Siebe Devroe, Héloïse Gollier, and Mathy Vanhoef. 2025. Fragile Frames: Wi-Fi's Fraught Fight Against FragAttacks. In *18th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '25), June 30–July 3, 2025, Arlington, VA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3734477.3734478

1 Introduction

In 2021, researchers disclosed three design flaws in the encryption protocols of the IEEE 802.11 standard that underpins Wi-Fi, along with a set of common implementation flaws [15]. These flaws affected the fragmentation and aggregation features of 802.11 and collectively called the FragAttacks. In response, the 802.11 standard was updated [4, 9], and the WPA3 specification incorporated guidance on how to avoid common implementation flaws [19].

This paper measures whether defenses against the FragAttacks have been correctly deployed. Initial evidence shows these defenses are complex and error-prone. For instance, backporting a FragAttacks patch to Linux kernels 4.4, 4.9, and 4.14 introduced memory safety issues [6], OpenBSD was initially misidentified as unaffected by one of the vulnerabilities [17], and testing defenses often requires modified drivers or firmware to simulate attacks [15]. Moreover, existing FragAttacks tests can only be executed when possessing network credentials, meaning it is currently not possible to perform more large-scale tests or surveys for FragAttacks vulnerabilities.

To measure how many networks properly deployed patches for the FragAttacks vulnerabilities, we create novel test methods that do not require network credentials and are compatible with a wider range of network cards. Because these tests actively interact with networks, we also examine the legal and ethical aspects of using them in Wi-Fi surveys. We then carry out real-world surveys in three cities at two points in time (2023 and 2025). Our findings reveal that many networks remain vulnerable, with over 30% in one city still affected by the EAPOL forwarding flaw, and more than 35% of routers from one national ISP allowing trivial packet injection. We disclosed these vulnerabilities to our national Computer Security Incident Response Team (CSIRT) and the affected ISPs.

Our surveys also reveal that almost no networks support the ideal defense against the aggregation-based A-MSDU attacks, likely because of compatibility issues. Networks that do implement protections rely on an ad-hoc mitigation that can be adopted without cooperation of other devices, but lacks strong security guarantees. Although this ad-hoc defense was included in the latest IEEE 802.11 standard [4], it provides only a practical fix without addressing the underlying design issue that the A-MSDU flag is not authenticated. We discover that this ad-hoc mitigation can be bypassed in mesh networks, validate the attack in practice, and propose, implement, and evaluate defenses. This flaw was assigned CVE-2025-27558, and we are collaborating with IEEE 802.11 to update the standard [16].

To summarize, our main contributions are:

- We develop 10 novel tests covering 6 of the 12 FragAttacks CVEs without requiring network credentials or having to perform full end-to-end attacks (Section 3).
- We conduct a Wi-Fi survey using the 5 most robust and ethical tests, discuss legality, and present our findings (Section 4).
- We show how to inject frames into a protected mesh network by abusing a design flaw in the 802.11 standard (Section 5).

We cover related work in Section 6 and conclude in Section 7. Lastly, we make our code available online at https://github.com/vanhoefm/fragattacks-survey-public and we disclosed all vulnerabilities to affected organizations and our national CSIRT.

2 Background and Motivation

This section introduces relevant aspects of the 802.11 standard and one of the key FragAttacks vulnerabilities.

2.1 Authentication and Association

Connecting to an Access Point (AP) starts with exchanging authentication frames. In an open network, in home WPA and WPA2 networks, and in all Enterprise networks, this is only a formality: no

WiSec '25, Arlington, VA, USA

ACM ISBN 979-8-4007-1530-3/25/06
https://doi.org/10.1145/3734477.3734478

actual authentication occurs at this stage. For WPA3 personal, however, the Dragonfly handshake takes place during the authentication phase, which requires knowledge of the password. After (open) authentication, the client and AP exchange association frames, where the client informs the AP about the features that it supports. After open authentication and association, actual authentication occurs in a subsequent 4-way handshake for personal networks, or with an EAP handshake for enterprise networks. Authentication and association is performed using management frames, while the 4-way and EAP handshake use EAPOL data frames.

All combined, a password is only required to complete authentication and association for WPA3 personal, but not for WPA, WPA2, or for enterprise networks.

2.2 Frame Aggregation and its Vulnerabilities

An Aggregate MAC Service Data Unit (A-MSDU) combines multiple 802.11 frames, where this aggregated frame then contains multiple A-MSDU subframes. Each subframe starts with an 802.3 header, consisting of the packet's destination and source MAC addresses, followed by its length. Each A-MSDU header is followed by the packet itself which always starts with an 8-byte rfc1042 header, whose first 6 bytes equal AA-AA-03-00-00-00. Each subframe except the last one is padded such that its length is a multiple of 4.

The frame's plaintext header contains a flag to indicate whether the frame transports a standard frame, i.e., an MSDU or A-MSDU. Unfortunately, this plaintext flag can be modified by an adversary, tricking a victim into treating a normal frame as an A-MSDU frame, enabling arbitrary frame injection [15].

One way to prevent this attack is to enable Signaling and Payload Protected (SPP) A-MSDUs, which ensures that the A-MSDU flag also gets authenticated and therefore cannot be modified. Networks and clients can advertise and negotiate support of SPP A-MSDU by enabling the right flags in the RSN element. This element is used to advertise the security features of a device. Unfortunately, setting these flags in the RSN element can result in issues with previously deployed devices [7], where they are unable to connect with devices that enable previously used features [19, §14.1]. The 802.11 standard addressed this by defining a *new* flag in the separately transmitted RSNX element to indicate support of SPP A-MSDUs [4, 7].

Because of these compatibility issues, the next (draft) 802.11 standard also includes an alternative defense: an A-MSDU must be dropped if a subframe's destination address equals the start of an rfc1042 header, i.e., if it equals the address AA:AA:03:00:00:00 [4, 8].

3 Credential-Free Testing for FragAttacks Flaws

In this section, we design novel methods to test whether a device is affected by a subset of the FragAttacks vulnerabilities without performing invasive attacks and without relying on credentials, i.e., without needing the password of the network. Our methods are compatible with more Wi-Fi dongles and in Section 4, we will use them to perform real-world Wi-Fi surveys.

We focus on vulnerabilities where testing a device's patch status does not require sending encrypted frames, because we cannot send them as an outsider, and we do not want to manipulate encrypted frames of real users. Under these constraints, we created credential-free tests for the following 6 out of 12 FragAttacks CVEs [17]:

3.1 Fake EAPOL (CVE-2020-26144)

3.1.1 Vulnerability. Affected devices accept plaintext A-MSDU frames in a protected network as long as the first 8 bytes equal an rfc1042 header for EAPOL [15]. This vulnerability is caused by an insecure check to allow plaintext EAPOL frames while connecting to the network, but where the implementation processes the frame afterwards as an A-MSDU frame instead. The source and destination of the first A-MSDU subframe will contain invalid values, causing this subframe to be ignored. However, the subsequent subframes will still be processed, and the attacker can use them to inject arbitrary plaintext packets to the AP.

3.1.2 Test method. To test if an AP is vulnerable, an A-MSDU frame is sent after associating to the AP, whose first 8 bytes equal an rfc1042 header for EAPOL, i.e., AA-AA-03-00-00-00-88-8E. The second subframe is a plaintext ping request with a broadcast (final) destination. A vulnerable AP ignores the first subframe because it has invalid addresses but still processes the ping request in the second subframe. The AP then broadcasts this ping request as a (protected) Wi-Fi frame, which can be detected by an outsider based on the addresses in the frame and its length.

An alternative to making the AP broadcast the injected ping request is to put the tester's Internet server as destination. If the ping request arrives at the server, then the AP is vulnerable. The advantage is that this approach is not impacted by client isolation and that the frame broadcasted by the AP does not have to be detected. For instance, with a Sitecom X8 AC1750, this ping test succeeded, but the broadcast test did not. We suspect packets with a broadcast final destination are handled differently from those with a unicast destination. However, the server ping test requires knowing the (private) IP addresses used in the network and exposes the network's public IP, therefore we did not use this test.

Home WPA3-only APs cannot be tested in this way, as an outsider cannot complete the Dragonfly handshake before associating. In our survey, only 0.1% of protected networks were WPA3-only.

3.1.3 Preconditions. One Wi-Fi dongle is required to have completed open authentication and association with the network. Some APs also require that a legitimate client has been connected to them at least once since the system has powered up. During the actual test, this legitimate client does not need to remain connected to the AP. For example, this is the case with the Linksys WAG320 AP.

3.2 Plaintext Injection Vulnerabilities

3.2.1 Vulnerability. Some APs accept plaintext frames (CVE-2020-26140) or *fragmented* plaintext frames (CVE-2020-26143), and some even do so before completing the 4-way handshake [15]. Certain devices also accept plaintext *broadcast fragments* (CVE-2020-26145). All three vulnerabilities trivially allow an attacker to inject frames.

3.2.2 Test method. To test an AP for these vulnerabilities, a (fragmented) plaintext ICMP ping frame is sent to an AP. The receiver address of this frame is the AP, or the broadcast address for CVE-2020-26145, the sender address is a connected client, and the final destination address the broadcast MAC address. A vulnerable AP that accepts one of these three types of plaintext frames will forward it to its final destination, meaning it will broadcast the frame, which can be detected by an outsider based on the frame's addresses

and length. The injected frame's source address must correspond to a connected client, otherwise the AP may not accept the frame [15].

Like the Fake EAPOL test, a ping request could be sent to a server, but this was not done due to its previously-detailed drawbacks.

3.2.3 Preconditions. One dongle must be associated, and this test requires identifying the MAC address of a connected client.

3.3 EAPOL Forward (CVE-2020-26139)

3.3.1 Vulnerability. As mentioned previously, APs are required to accept plaintext EAPOL frames since they are used in the 4-way handshake when connecting. Some APs, however, forward these plaintext EAPOL frames to other clients when the EAPOL frame is not destined for the AP [15]. Some APs even do this before the sending client has fully authenticated. This vulnerability can be abused for Denial-of-Service attacks and allows the exploitation of the A-MSDU design flaw to inject an arbitrary packet to a client while merely being within range of the targeted network and client [15].

3.3.2 Test method. To test an AP for this vulnerability, two Wi-Fi dongles are used that both first complete open authentication and association with the network. The first dongle then sends an EAPOL frame to the AP with the second dongle as final destination. A vulnerable AP forwards this frame to the second dongle, which can trivially be detected. A disadvantage of this approach is that some APs may not forward EAPOL frames from clients that are not fully authenticated and that some APs may not forward EAPOL frames to not fully authenticated clients. Although this can be overcome by trying to detect two fully authenticated legitimate clients and using their MAC addresses, this is tedious in practice and significantly reduces the number of networks that can be tested. Fortunately, associating with two dongles and sending EAPOL frames between them already detects a lot of vulnerable APs (see Section 4).

3.3.3 Preconditions. Two Wi-Fi dongles that completed open authentication and association with the network are required.

3.4 Spoofing A-MSDUs (CVE-2020-24588)

3.4.1 Vulnerability. The unauthenticated A-MSDU flag can be manipulated to inject any packet towards a client (recall Section 2.2).

3.4.2 Test method. There are two ways devices can prevent this vulnerability. First, devices can require SSP A-MSDUs. Determining whether a network requires SSP A-MSDU can be done by inspecting beacon frames, but this is rarely supported due to compatibility issues. Instead, in practice, most devices adopt the ad-hoc mitigation where the A-MSDU is dropped if a subframe's destination address equals the start of an rfc1042 header (see Section 2.2 and 4.3.6).

To test whether an AP adopted the above ad-hoc mitigation, we must send a *plaintext* data frame to the AP that triggers a reaction if the frame is not dropped. To the best of our knowledge, the only data frames that can be transmitted without fully authenticating with the network are EAPOL frames. However, none of the EAPOL frames sent to the AP in the 4-way handshake guarantee a reaction. Instead, we focus on Enterprise networks where EAPOL frames are used for 802.1X authentication after association (recall Section 2.1).

The first EAPOL packet in 802.1X towards the AP is an Identity Response frame. We send this response, with the fixed username `test@test.org`, as the second subframe in an A-MSDU, where the destination of the first subframe equals the start of an rfc1042 header. When the AP is vulnerable, it will ignore the first subframe but still process the second subframe containing the Identity Response. The precise reply depends on the authentication protocol but usually indicates that a nonexistent username was used, which our tool can detect to identify a vulnerable AP.

3.4.3 Preconditions. This test requires one associated dongle and is limited to Enterprise networks which use 802.1X authentication.

4 Wi-Fi Survey on FragAttacks Patch Adoption

This section uses our novel credential-free FragAttacks tests to survey how many real-world APs are still vulnerable.

4.1 Legal and ethical aspects

4.1.1 Legality. Our survey is done in Belgium, where a law on digital whistleblowers, i.e., ethical hacking, took effect in 2023 [18]. This law protects researchers if they adhere to the following:

(1) They acted without fraudulent intent or intent to cause harm.
(2) They promptly informed the affected organization, and no later than the time of notification to the national CSIRT.
(3) They limited their actions to what was necessary and proportionate to verify the existence of the vulnerability.
(4) They did not disclose information about the discovered vulnerability without the consent of the national CSIRT.

Our goal is not to cause harm but to assess patch status and whether defenses require further improvements. We notified affected ISPs and minimized the number of frames that are sent to test for a vulnerability. Lastly, we informed our national CSIRT before writing this paper and they did not raise any concerns.

As an extra measure to show good intentions, we included the text *"This is a test, see survey.fragattacks.com"* in at least one frame of every test to help network administrators recognize our tests and find details online. Our website explains how to opt out by contacting us or by adding `_nomap` or `_optout` to the network name.

4.1.2 Ethics. Our tests are designed to be minimal, to avoid network and user impact, and to prevent sensitive data leaks. In particular, none of our tests initiate a 4-way or other authentication handshake, meaning no (indirect) info about the network's password is obtained. Additionally, both the *Fake EAPOL* and *EAPOL Forward* tests do not involve legitimate clients, and only make the AP forward a single frame to test whether it is vulnerable. For the *Plaintext Injection* test, we do not use the variant where a ping request is sent to a server to reduce the number of transmitted frames and to avoid generating outgoing traffic, which might otherwise consume a few bytes of the user's data limit and reveal the network's public IP address. Instead, we send a broadcast ping, which a vulnerable AP retransmits as a Wi-Fi frame. This test requires knowing the MAC address of a connected client, but sending the broadcast ping does not affect this client and has no impact on the network. Finally, in the *Spoofing A-MSDUs* test, the Identity Response frame uses the dummy username `test@test.org` which does not impact regular users nor the network as a whole.

We contacted our university's Institutional Review Board (IRB), but they could not assess our study because it does not directly

Table 1: Percentage of affected APs out of those that met the test preconditions, for each year, surveyed city, and for ISP APs.

Vulnerability	Leuven		Heverlee		Roeselare	Telenet		Proximus		Orange	
	2023	2025	2023	2025	2025	2023	2025	2023	2025	2023	2025
Fake EAPOL	5.74%	9.00%	8.35%	12.36%	10.98%	5.34%	8.27%	2.81%	4.52%	36.65%	35.17%
EAPOL forward	22.09%	20.09%	29.39%	25.73%	30.30%	20.50%	17.02%	88.96%	37.82%	84.51%	60.31%
Plain. full	1.44%	1.30%	2.91%	2.21%	2.10%	2.50%	2.50%	0.00%	1.51%	4.00%	0.00%
Plain. frag.	2.36%	2.12%	3.83%	3.32%	4.00%	3.75%	3.02%	1.14%	0.50%	2.00%	0.00%
Spoof. A-MSDU	1.32%	1.75%	2.79%	0.67%	2.17%	0.00%	—	—	—	—	—

involve humans. As an alternative, we gave presentations on our methodology at three different universities in the US, Europe, and West Asia. The audience consisted of cybersecurity researchers, ranging from master students to professors. Feedback included to first test our own devices, which we already did, and performing the survey in phases so that we can stop early if needed. The latter advice was followed by first performing the survey in a single city, and afterwards determining if it was useful to continue, and by contacting ISPs and our national CSIRT before doing a second survey in 2025. All combined, no major concerns were raised.

4.2 Methodology

We implemented the tests of Section 3 in Python. We did not test for CVE-2020-26145, since APs are unlikely to be affected and the original FragAttacks research also did not find affected APs [15]. We used channel hopping over channels 1, 6, and 11, since they host most networks. We did not test APs in the 5 GHz band, because this band typically requires performing weather radar detection before transmissions are allowed [4, §11.8]. Initially, a TL-WN722N was used to transmit frames and a CSL 300 dongle, having a RT5572 chipset, to receive frames. Later on, CSL 300 dongles were exclusively used. We used a Linux kernel that properly injected fragmented Wi-Fi frames [17]. In March 2023, a first survey was done in Leuven and Heverlee in Belgium, and in February 2025, a second survey was done in the same cities and also Roeselare. In both surveys, researchers walked around for several days until sufficiently many networks were detected and tested by the created tool.

4.3 Survey results

In 2023, we detected 23 110 unique APs, and 34 727 in 2025 due to the extra survey in Roeselare, totaling 52 691 unique APs over both years. Table 1 lists the percentage of affected APs to each test. Not all the detected protected networks met the preconditions to perform each test. As a representative example, in the 2023 survey in Leuven, our tool successfully associated with 52% of protected networks using one dongle, a precondition of the *Fake EAPOL* test, and for 43% of protected networks with both dongles, a precondition of the *EAPOL forward* test. The exact number of affected and tested APs is available in our repository [1]. We believe that packet loss, and networks going out of range while walking, are the main reasons for not always successfully associating. Finally, for 26% of networks, we could find a connected client required for the *Plaintext Injection* tests. Lastly, the percentage of vulnerable networks is a lower bound: due to packet loss or client isolation, our tool might not have detected the packet that would indicate an AP is vulnerable.

4.3.1 Security impact. A notable number of networks allow trivial packet injection, e.g., via the Fake EAPOL attack. These flaws allow an attacker to inject packets but not read replies. Interesting future work is studying the impact of such *injection-only* flaws in detail.

4.3.2 Impact of region. In our surveys in 2025, networks in Roeselare appear to be more vulnerable than those in Leuven and Heverlee. Additionally, around 53% of networks in Roeselare support WPS, which is considered insecure, as opposed to 41% and 45% in Leuven and Heverlee, respectively. We conjecture that this is because Leuven and Heverlee have many student residences, where routers get updated more frequently [11].

4.3.3 Vendor analysis. By inferring the vendor from the AP's MAC address, we observed that some vendors, e.g., HP and Cisco Meraki, have few affected devices, while some other vendors, e.g., Askey, Arcadyan, and Huawei, have over 80% of devices affected by at least one vulnerability. More details are in our repository [1]. We believe that (the lack of) automatic, or at least centrally managed, AP and router updates are a driving force behind these differences.

4.3.4 ISPs. We also studied the patch status of ISP routers, i.e., ISP APs, since one might assume they receive more updates and are more secure. There are three widely-used ISPs in Belgium that we can reliably detect. In particular, the name of these ISPs is included in the Wi-Fi network name by default, making them easily recognizable. We confirmed that recognizing an ISP using network SSIDs has high accuracy by analyzing the vendor of each ISP AP, which showed that the vast majority of networks with the same ISP in their SSID come from just a few, or even a single, vendor. Overall results are shown in Table 1, where we excluded Roeselare for the ISP analysis to ensure a fair comparison between the different years. Results are similar when including the data from Roeselare. In most cities and years, we barely detected any Enterprise networks from ISP APs, and could not derive accurate percentages.

Surprisingly, ISP APs are overall more vulnerable than others. Especially, the EAPOL Forward flaw affects many ISP APs, and many of them are also vulnerable to plaintext frame injection attacks, which are arguably easier to exploit in practice.

4.3.5 Historical trends. Surprisingly, in 2025, more networks in Leuven and Heverlee were vulnerable to the Fake EAPOL attack. Examining the vendors reveals more APs from Ubiquiti Inc., a Wi-Fi AP manufacturer, that is significantly affected by the Fake EAPOL flaw. The prevalence of other flaws appears to slowly decrease.

Also notable is that Proximus had significantly fewer APs that were vulnerable to the *EAPOL forward* test. Additionally, in 2023,

Telenet offered a country-wide Enterprise hotspot that customers could connect to, which was discontinued in 2024. This provided enough Enterprise APs that could be tested for *Spoofing A-MSDU* in Leuven in 2023, but not in any other cities or years.

4.3.6 Lacking SSP A-MSDU support. During our survey in 2025, we inspected beacons and probe responses, which are sent by APs to advertise their presence and properties, and found that around 0.03% of networks supported SSP A-MSDUs. All these APs appear to be part of entertainment or navigation devices in cars. This shows that this defense is rarely adopted, possibly because it causes compatibility issues in practice, meaning most devices rely on the alternative ad-hoc mitigation discussed in Section 2.2 and 5.1.

5 Mesh Networks: FragAttacks Defense Bypass

This section bypasses the 802.11 standard's ad-hoc fix for the FragAttacks A-MSDU flaw in mesh networks, allowing attackers to inject frames. The attack is evaluated and is assigned CVE-2025-27558.

5.1 Background on Mesh Networks

The 802.11s amendment, released in 2011, introduced support for mesh networks. In these networks, the plaintext header of data frames can have four addresses, allowing a receiver to forward frames on behalf of another client. The (encrypted) payload of mesh data frames always start with a 6-byte Mesh Control field (see the top of Figure 1). This field contains a 1-byte Flags field to indicate the usage and size of the Mesh Address Extension field, a 1-byte Time-To-Live (TTL) field to prevent routing loops, and a 4-byte Mesh Sequence Number. Depending on the two lowest-order bits of the Flags field, the Mesh Control field is followed by an empty, 6-byte, or 12-byte Mesh Address Extension field that contains extra addresses for advanced routing (see Figure 1).

Mesh networks also support A-MSDUs and rely on identical FragAttacks mitigations. Unfortunately, enabling more secure SPP A-MSDUs results in the same compatibility issues. More troublesome, with the open-source `hostap` daemon for Linux, support for SPP A-MSDUs cannot even be enabled for mesh networks. As a result, mesh clients rely on the following mitigation that was included in the latest IEEE 802.11 standard [4]:

> *"the first six octets of the first A-MSDU subframe header (the mesh DA, if the frame is from a mesh STA, or the DA in a Basic A-MSDU not from a mesh STA) shall not be AA-AA-03-00-00-00"*

This defense is implemented in Linux, where for mesh networks the Destination Address (DA) of all A-MSDU subframes is compared to `AA:AA:03:00:00:00` as indicated in the quoted instructions.

5.2 A-MSDU Defense Bypass in Mesh Networks

5.2.1 Threat model. Our target is a mesh client that adopted all FragAttacks defenses and is in radio range of the attacker. This mesh client may be part of a typical home network. For instance, Google Wi-Fi, Nest, and open-source OpenWRT routers support 802.11s mesh links to extend coverage. Similar to the original FragAttacks threat models, the adversary must be able to send IPv4 packets to the victim [15]. This can be done directly if the victim's public IP address is known, and can otherwise be accomplished by social engineering the victim into connecting with the adversary's server, e.g., by tricking the victim into opening a link, which is a simplification of the BEAST-like threat model used in many TLS attacks [15].

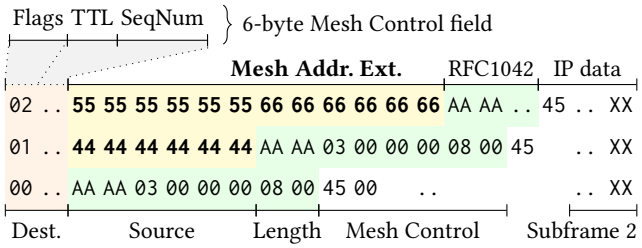

Figure 1: Three mesh frames and their fields when parsed as a single frame (top) or as an A-MSDU frame (bottom). When parsed as a single frame, the length of the (optional) Mesh Address Extension field shown in bold (yellow) depends on the two lowest-order bits of the Flags field (see Section 5).

5.2.2 Standard attack. We exploit the A-MSDU flaw in mesh networks by modifying a standard mesh frame so that, when processed as an A-MSDU, one of its subframes becomes the injected packet, while bypassing existing mitigations. We do this by sending a tailored IPv4 packet to the victim. When the Mesh Address Extension field is not used by the sender, the length field of the first A-MSDU subframe equals 8 bytes (see Figure 1), and the first byte of the Mesh Control field in the first subframe equals 45, meaning the Address Extension is 6-bytes long. This means the first A-MSDU subframe consists of the 14-byte A-MSDU header, the 6-byte Mesh Control field, the 6-byte Address Extension, and the 8-byte payload, giving a total length of 34 bytes, meaning it is followed by 2 bytes of padding. Since our crafted IPv4 packet has an IPv4 header of 20 bytes, and is preceded by a 6-byte Mesh Control and 8-byte rfc1042 field, the second A-MSDU subframe begins 2 bytes after the IPv4 header. This gives the adversary full control over the second A-MSDU subframe, enabling the injection of arbitrary layer 2 packets.

The destination addresses of the A-MSDU subframes in Figure 1 never equal AA-AA-03-00-00-00, meaning existing defenses do not prevent our attack. Additionally, the adversary can use a channel-based Machine-in-the-Middle (MitM) to flip the A-MSDU flag to change a normal frame into an A-MSDU during an attack [15].

If the Mesh Address Extension field is 6 bytes long, the length field of the first A-MSDU subframe equals the bytes AA-AA, which is a length that is too long to exploit. If the Mesh Address Extension field is 12 bytes long, the length field of the first A-MSDU subframe equals the first two bytes of the end destination's MAC address, meaning exploitability depends on the value of this address.

We tested our attack with `wpa_supplicant` 2.11 on Linux 6.12. To facilitate reproducibility, we used the `mac80211_hwsim` driver to create virtual Wi-Fi network cards and assumed that no Mesh Address Extension field was used. Under this setup, we successfully performed the attack and injected arbitrary frames to the victim.

5.2.3 Exploiting non-standard A-MSDUs. The endianness of the length field in a mesh A-MSDU is not explicitly defined, and therefore uses the default little-endian encoding of the 802.11 standard. However, the non-mesh A-MSDU format is defined as equivalent to Ethernet frames, meaning its length field is big-endian [4, §9.3.2.2.2].

These subtleties caused confusion and compatibility issues, where some implementations (wrongly) parsed the length field in A-MSDU mesh frames as big-endian [2]. Such clients can still be exploited if a 12-byte mesh extension field is used, or if no extension field is used and one can send TCP/IP packets larger than 2048 bytes.

5.2.4 Abusing EAPOL forwarding. If the network has a mesh client that forwards EAPOL frames, which our survey showed is a widespread flaw, the adversary only needs to be in radio range to perform A-MSDU attacks. In particular, the adversary can inject a plaintext mesh frame with a 12-byte Mesh Address Extension field, followed by an rfc1042 header indicating that the payload contains an EAPOL frame. This frame is accepted and forwarded with encryption to the victim. The adversary then sets the A-MSDU flag in the header of this encrypted forwarded frame. Since the adversary controls the full Mesh Address Extension field, and all data after the rfc1042 header, they can craft these values and use the second A-MSDU subframe to inject arbitrary frames to the victim. All combined, this allows an adversary to inject frames without needing to send IP packets to the victim, i.e., no social engineering is required anymore.

5.2.5 Mesh Control Present flag. The 802.11 header also has a *Mesh Control Present* flag which we were unable to abuse in practice.

5.3 Defenses and mitigations

Ideally, SPP A-MSDUs are enabled, ensuring the A-MSDU flag in the 802.11 header is authenticated [4, §10.11]. However, using this feature leads to compatibility issues and therefore requires updates to the sender and receiver before it can be enabled (recall Section 5).

Alternatively, a mesh client can drop A-MSDU frames that, if parsed as a non-aggregated MSDU frame, start with an rfc1042 header. Concretely, let n be the two lowest-order bits of the first byte of the MSDU, then drop the frame if $n \neq 3$ and the 6 bytes at offset $(6 + 6 \cdot n)$ equal AA-AA-03-00-00-00. We confirmed this defense on Linux kernel 6.1.110 and proposed this defense to the IEEE 802.11 [16]. However, novel attacks may remain possible.

6 Related Work

Our work is based on that of Vanhoef [15], where we extend their methods to test for most FragAttacks vulnerabilities in a credential-free manner without having to perform end-to-end attacks, and while ensuring the tests are compatible with more network cards. Our tests allow surveying how many networks remain vulnerable.

Several works survey Wi-Fi networks [5, 10, 12, 14] and Schepers et al. give best practices for doing so [11]. These works passively capture beacon and probe requests and analyze advertised network properties. In contrast, Hoorvitch actively transmitted authentication and association frames to check if a network's password can be brute-forced from a leaked PMKID [3]. They did not discuss ethical aspects. We are the first to interact with networks to test for FragAttacks vulnerabilities, while considering ethics and legality, showing that many networks are still affected by FragAttacks flaws.

To inject custom frames properly, changes to Linux or firmware may be needed [15]. Others extended Radiotap to facilitate this and used an open Wi-Fi stack on top of software defined radios [17].

Compared to infrastructure Wi-Fi networks, there has been less focus on mesh Wi-Fi security. Nevertheless, researchers studied the security of proprietary mesh protocols [20], and studied mesh networks without focusing on Wi-Fi specifically [13]. In contrast, we analyze standardized 802.11 mesh protocols and found that the mitigation to prevent abuse of the A-MSDU flag can be bypassed.

7 Conclusion

Our credential-free test methods showed that a significant fraction of Wi-Fi networks remain vulnerable to some of the FragAttacks, especially to flaws like EAPOL forwarding. We also discovered a design flaw in 802.11's A-MSDU mitigation for mesh networks, and we proposed, implemented, and evaluated an updated defense.

We hope our work encourages (discussions on) safe interactions with wireless networks to survey their security. In this light, the safeguards we used can serve as a foundation for future more detailed surveys, including for other wireless protocols.

Acknowledgments

This research is partially funded by the Research Fund KU Leuven and the Cybersecurity Research Programme Flanders.

References

[1] 2025. https://github.com/vanhoefm/fragattacks-survey-public.
[2] Felix Fietkau. 2023. wifi: mac80211: implement support for yet another mesh A-MSDU format. Linux commit `fe4a6d2db3ba`.
[3] Ido Hoorvitch. 2021. Cracking WiFi at Scale with One Simple Trick. https://www.cyberark.com/resources/threat-research-blog/cracking-wifi-at-scale-with-one-simple-trick. Accessed 3 March 2025.
[4] IEEE Std 802.11. 2024. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.*
[5] Kipp Jones and Ling Liu. 2007. What where wi: An analysis of millions of wi-fi access points. In *IEEE International Conference on Portable Information Devices.*
[6] Davis Mosenkovs. 2016. mac80211: fix memory corruption in EAPOL handling. lore.kernel.org/all/YPAiNH03VHTgDwho@kroah.com/T. Accessed 25 Feb. 2025.
[7] Emily Qi, Johnanes Berg, Ido Ouzieli, Mark Rison, Jouni Malinen, and Mark Hamilton. 2022. Proposed Resolution for SPP A-MSDU Support. https://mentor.ieee.org/802.11/dcn/22/11-22-0398-02-000m-proposed-resolution-for-spp-a-msdu-support.docx. Accessed 19 February 2025.
[8] Mark Rison, Mathy Vanhoef, Mark Hamilton, and Jouni Malinen. 2021. On A-MSDU addressing. https://mentor.ieee.org/802.11/dcn/21/11-21-0816-03-000m-on-a-msdu-addressing.docx. Accessed 4 February 2024.
[9] Mark Rison, Mathy Vanhoef, Mark Hamilton, and Jouni Malinen. 2022. On FrAttacks and related matters. https://mentor.ieee.org/802.11/dcn/21/11-21-1128-06-000m-on-frattacks-and-related-matters.docx. Accessed 4 March 2024.
[10] Amirali Sanatinia, Sashank Narain, and Guevara Noubir. 2013. Wireless spreading of WiFi APs infections using WPS flaws: An epidemiological and experimental study. In *Conference on Communications and Network Security (CNS).*
[11] Domien Schepers, Aanjhan Ranganathan, and Mathy Vanhoef. 2021. Let numbers tell the tale: measuring security trends in Wi-Fi networks and best practices. In *ACM WiSec.*
[12] A Sebbar, SE Boulahya, G Mezzour, and M Boulmalf. 2016. An empirical study of wifi security and performance in morocco-wardriving in rabat. In *ICEIT.*
[13] Aggeliki Sgora, Dimitrios D Vergados, and Periklis Chatzimisios. 2016. A survey on security and privacy issues in wireless mesh networks. *SCN* (2016).
[14] Hristo Valchanov, Jan Edikyan, and Veneta Aleksieva. 2019. An Empirical Study of Wireless Security in City Environment. In *Balkan Conference on Informatics.*
[15] Mathy Vanhoef. 2021. Fragment and Forge: Breaking Wi-Fi Through Frame Aggregation and Fragmentation. In *USENIX Security Symposium.*
[16] Mathy Vanhoef. 2025. Protecting against A-MSDU Attacks in Mesh Networks. https://mentor.ieee.org/802.11/dcn/25/11-25-0949-00-000m-a-msdu-mesh-spoof-protection.docx. Accessed 14 May 2025.
[17] Mathy Vanhoef, Xianjun Jiao, Wei Liu, and Ingrid Moerman. 2023. Testing and Improving the Correctness of Wi-Fi Frame Injection. In *ACM WiSec.*
[18] Belgisch Staatsblad. 2022. Wet 15 december 2022 betreffende de bescherming van melders van inbreuken op het Unie- of nationale recht vastgesteld binnen een juridische entiteit in de private sector. Artikel 50..
[19] Wi-Fi Alliance. 2024. WPA3 Specification Version 3.4. https://www.wi-fi.org/file/wpa3-specification. Accessed 19 February 2025.
[20] Xin'an Zhou, Qing Deng, Juefei Pu, Keyu Man, Zhiyun Qian, and Srikanth V Krishnamurthy. 2024. Untangling the Knot: Breaking Access Control in Home Wireless Mesh Networks. In *ACM CCS.*