



ESPILON
SECURITY RESEARCH

SECURITY ADVISORY

Out-of-bounds read in the ESP-IDF DHCP server option parser

CVE-2026-45160

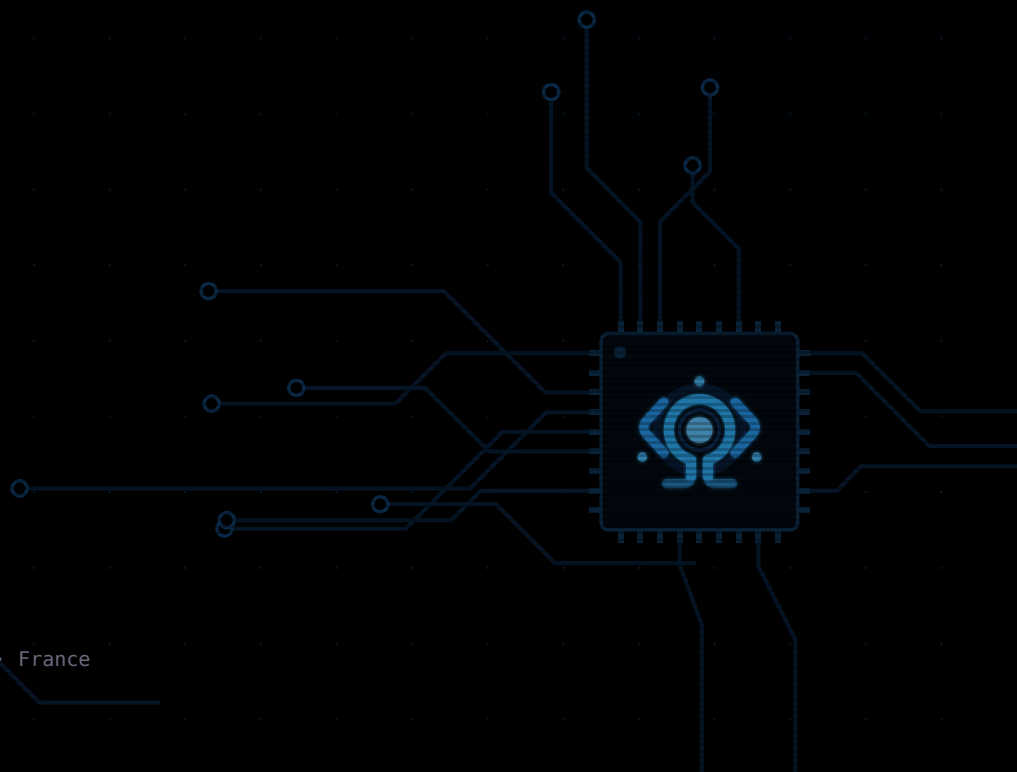
CVSS 6.5 · MEDIUM

Espressif · ESP-IDF · DHCP server

CVE PUBLISHED **10 Jun 2026**

Marouane El Moufid (Eun0us)

espilon.net · Association loi 1901 · France



01 Vulnerability

6.5

CVSS 3.1 BASE SCORE · GITHUB (CNA), GHSA-g764-gwc3-75m5

AV:A / AC:L / PR:N / UI:N / S:U / C:N / I:N / A:H

MEDIUM

CVE	CVE-2026-45160 (CNA: GitHub)
Advisory	GHSA-g764-gwc3-75m5 , published 2026-05-21
Component	ESP-IDF DHCP server, parse_options() in components/lwip/apps/dhcpserver/dhcpserver.c
Type	Out-of-bounds read (CWE-125)
Attacker	Adjacent network: a station on the same Layer-2 segment as the DHCP server, no authentication
Credit	Epsilon · Marouane El Moufid (Eun0us)

Summary

The DHCP server that ESP-IDF ships in its lwIP component directory (used by the SoftAP and by any configuration in which the device serves addresses) walks the options field of each incoming DHCP request without checking that an option's length byte, or the payload that length announces, lies inside the received packet. A single crafted DHCP datagram makes the parser read past the end of the options buffer into adjacent heap memory. Espressif rates the consequence as a denial of service of the DHCP server.

This server is Espressif's own code (dhcpserver.c, copyright Espressif Systems), not part of the lwIP project: upstream lwIP has no DHCP server and is not affected. The advisory and the CVE record name the component "lwIP DHCP Server (ESP-IDF)" after the directory it sits in.

Affected and fixed versions

The advisory lists the last vulnerable and the first patched release per branch; we checked each tag on 2026-10-28 for the fix. Station-only devices, which never run the DHCP server, are not affected.

BRANCH	AFFECTED	PATCHED	COMMIT	TAG CHECK (2026-10-28)
v5.2	v5.2.7	v5.2.8	2da2db43	v5.2.8 carries the fix
v5.3	v5.3.5	v5.3.6	9f713dbc	v5.3.6 carries the fix
v5.4	v5.4.4	v5.4.5	2bf4dd12	v5.4.5 not tagged yet; fix on release/v5.4
v5.5	v5.5.4	v5.5.5	d51b1076	v5.5.5 carries the fix
v6.0	v6.0.1	v6.0.2	fba5f995	v6.0.2 carries the fix
master			8b4b5d53	fixed on 2026-04-09

Timeline

DATE	EVENT
2026-04-09	Fix committed to ESP-IDF master (8b4b5d53)
2026-05-08	CVE-2026-45160 reserved by the GitHub CNA
2026-05-12	Espressif bug bounty programme confirms report #440 as valid
2026-05-21	Advisory GHSA-g764-gwc3-75m5 published
2026-06-10	CVE record published

02 Technical analysis

Vulnerable code

`parse_options(dhcps_t *dhcps, u8_t *optptr, s16_t len)` (line 917) receives a pointer to the first option and the number of option bytes, and derives `end` from them. Each iteration dereferences the option code, then the length byte at `optptr[1]`, then the payload at `optptr + 2`, and finally advances by `optptr[1] + 2`. The only bound is `optptr < end`, which covers the code byte and nothing after it:

• `components/lwip/apps/dhcpserver/dhcpserver.c, parse_options()` at 508ba7ae, parent of the master fix (abridged)

```
925     u8_t *end = optptr + len;
926     u16_t type = 0;
927
928     s.state = DHCP_STATE_IDLE;
929
930     while (optptr < end) {
931         switch ((s16_t) *optptr) {
932             case DHCP_OPTION_MSG_TYPE: //53
933                 type = *(optptr + 2);
934                 break;
935
936             case DHCP_OPTION_REQ_IPADDR://50
937                 if (memcmp((char *) &client.addr, (char *) optptr + 2, 4) == 0) {
938
939                     optptr += optptr[1] + 2;
940                 }
941         }
```

Root cause

When the last option starts at `end - 1`, line 993 reads its length byte one byte past `end`. A message-type option (53) placed there makes line 938 read `optptr[2]`, and a requested-address option (50) makes the `memcmp` at line 969 read `optptr[2]` to `optptr[5]`, that is `end[1]` to `end[4]`. Nothing compares the declared length with the option's real size either, so a short option 50 is compared against whatever follows it. With `CONFIG_LWIP_DHCP_REPORT_CLIENT_HOSTNAME` enabled, the host-name handler (lines 941 to 966) has the same gap: it copies up to `CONFIG_LWIP_DHCP_MAX_HOSTNAME_LEN - 1` bytes from `optptr + 2`, filtered to a restricted character set, into the server's lease record, not into a reply.

Where the buffer ends

`handle_dhcp()`, the UDP receive callback, copies the datagram into a heap block whose size is the larger of `sizeof(struct dhcp_msg)` (548 bytes) and the payload length, then hands `parse_msg()` the payload length minus the 240-byte fixed header:

• `components/lwip/apps/dhcpserver/dhcpserver.c, handle_dhcp() and parse_msg()` at 508ba7ae (abridged)

```
1234     malloc_len = sizeof(struct dhcp_msg);
1235
1236     if (malloc_len < p->tot_len) {
1237         malloc_len = p->tot_len;
1238     }
1239
1240     pmsg_dhcp = (struct dhcp_msg *)mem_malloc(1, malloc_len);
1241
1242     state = parse_msg(dhcp, pmsg_dhcp, tlen - 240);
1243
1244     s16_t ret = parse_options(dhcp, &m->options[4], len);
```

The options start 240 bytes into the message (`&m->options[4]`, after the magic cookie), so `end` points just past the last byte of the datagram. For a datagram close to or larger than 548 bytes the heap block ends there as well, and the bytes read past `end` belong to the next heap object. For a shorter datagram they fall in the zero-filled tail of the 548-byte block.

Reachability

The callback is registered with `udp_recv()` on the server's UDP socket and runs for every datagram it receives; `parse_options()` is reached once the magic cookie matches. No lease, prior exchange or credential is needed. The attacker is any host on the same Layer-2 segment: in practice a station associated with the device's SoftAP. On an open SoftAP that is anyone in radio range; on a WPA2/WPA3 SoftAP it is anyone holding the passphrase.

Impact

The out-of-bounds bytes are read, never written. They decide the parsed message type, the ACK/NAK decision for option 50, and how far the walker advances, but they are not copied into any DHCP reply. Espressif's assessment, which sets the official score, is that the read can crash the lwIP/DHCP server task or corrupt parser state, denying address assignment to every client of the SoftAP (A:H, no confidentiality or integrity impact). We do not claim more than that.

Evidence

Reported to the Espressif bug bounty programme as report #440 and confirmed by Espressif. No sanitizer trace or harness from that report is kept in our archive, so none is shown: the defect rests on the vulnerable source above and on the vendor's fix. The trigger the code implies is a DHCP datagram with a valid magic cookie, close to or larger than 548 bytes, whose last option starts on its final byte.

03 Fix and recommendation

Upstream fix

Commit 8b4b5d53 (David Cermak, Espressif, 2026-04-09, "fix(lwip): Fix DHCP server OOB read issue") skips PAD options, stops on END, and refuses to read the length byte or the payload unless both lie inside the buffer; the message-type and requested-address handlers also check the declared length. The core of the change is reproduced verbatim below. Elided: the copyright-year hunk, the new DHCP_OPTION_PAD define, two debug-log context lines, the removal of the `is_dhcp_parse_end` flag and of the old END case, and the option 50 branch, now wrapped in `if (opt_len >= 4)`.

• espressif/esp-idf commit 8b4b5d5301815198d177974ffc24848f47748248, dhcpserver.c (abridged)

```
@@ -932,10 +932,31 @@ static u8_t parse_options(dhcps_t *dhcps, u8_t *optptr, s16_t len)
[...]
```

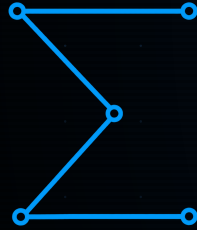
```
+     if (*optptr == DHCP_OPTION_PAD) {
+         optptr++;
+         continue;
+     }
+
+     if (*optptr == DHCP_OPTION_END) {
+         break;
+     }
+
+     if (optptr + 1 >= end) {
+         break;
+     }
+
+     u8_t opt_len = optptr[1];
+
+     if (optptr + 2 + opt_len > end) {
+         break;
+     }
+
+     switch ((s16_t) *optptr) {
+
+         case DHCP_OPTION_MSG_TYPE: //53
+             type = *(optptr + 2);
+             if (opt_len >= 1) {
+                 type = optptr[2];
+             }
+             break;
+
+         ...
+         optptr += optptr[1] + 2;
+         optptr += opt_len + 2;
+     }
}
```

Recommendation

Update to v5.2.8, v5.3.6, v5.5.5 or v6.0.2 (on v5.4, the head of `release/v5.4` until v5.4.5 is tagged), or apply the commit for your branch from section 01. Per Espressif, no reliable workaround exists short of disabling the DHCP server; a strong WPA2/WPA3 passphrase narrows who can reach the parser but does not remove the flaw.

04 Credit and references

- Credit: Marouane El Moufid (EunOus), Espilon; the advisory credits "EunOus / Espilon Worker" as reporter
- Advisory: github.com/espressif/esp-idf/security/advisories/GHSA-g764-gwc3-75m5
- CVE record: www.cve.org/CVERecord?id=CVE-2026-45160
- Fix (master): github.com/espressif/esp-idf/commit/8b4b5d5301815198d177974ffc24848f47748248



ESPILON

SECURITY RESEARCH

Embedded, IoT and OT security research

espilon.net

eun0us@espilon.net

Association loi 1901 • France