How to disable the private tmp?

Some application not able to read from system tmp any more.
The main reason for that is, that with debian 9, some processes which uses systemd forced the private tmp directories. Of course, I understand the security part behind that idea. And also the man page of systemd gives some more explanation:

man systemd.exec
NAME
    systemd.exec - Execution environment configuration

SYNOPSIS
    service.service, socket.socket, mount.mount, swap.swap

PrivateTmp=

> *Takes a boolean argument. If true, sets up a new file system namespace for the executed processes and mounts private /tmp and /var/tmp directories inside it that is not shared by processes outside of the namespace. This is useful to secure access to temporary files of the process, but makes sharing between processes via /tmp or /var/tmp impossible. If this is enabled, all temporary files created by a service in these directories will be removed after the service is stopped. Defaults to false. It is possible to run two or more units within the same private /tmp and /var/tmp namespace by using the JoinsNamespaceOf= directive, see systemd.unit(5) for details.*
> *This setting is implied if DynamicUser= is set. For this setting the same restrictions regarding mount propagation and privileges apply as for ReadOnlyPaths= and related calls, see above.*

To disable it, I copy the systemd service file to /etc/systemd/system/, change the line PrivateTmp=true to false and restart the process.

I use apache2 as an example:

```
$ cp /lib/systemd/system/apache2.service /etc/systemd/system/
```

Now we edit the PrivateTmp=true to PrivateTmp=false:

```
$ grep PrivateTmp /etc/systemd/system/apache2.service
PrivateTmp=true

$ vi /etc/systemd/system/apache2.service
.....

$ grep PrivateTmp /etc/systemd/system/apache2.service
PrivateTmp=false
```

And at the end, restart the apache2 process:

```
$ systemctl restart apache2.service
```