Date: Mon, 10 Jan 2022 18:08:29 +0000
From: Qualys Security Advisory <qsa@...lys.com>
To: "oss-security@...ts.openwall.com" <oss-security@...ts.openwall.com>
Subject: CVE-2021-3997: Uncontrolled recursion in systemd's systemd-tmpfiles

Hi all,

We discovered a minor denial of service (an uncontrolled recursion) in
systemd-tmpfiles, CVE-2021-3997; the Coordinated Release Date is today
(January 10, 2022), and a patch is now available at (many thanks to
Zbigniew Jedrzejewski-Szmek for working on this):

https://github.com/systemd/systemd/commit/55a89ea1b4088a6d84ba0bd3cd8e648bd51f1ebf

Below is a short write-up (which is part of a longer advisory that is
mostly unrelated to systemd and that we will publish at a later date):

```
========================================================================
CVE-2021-3997: Uncontrolled recursion in systemd's systemd-tmpfiles
========================================================================

[...]

We therefore looked into systemd-tmpfiles (which "creates, deletes, and
cleans up volatile and temporary files and directories") and discovered
a denial of service (an uncontrolled recursion): if we create thousands
of nested directories in /tmp, then "systemd-tmpfiles --remove" (when
executed as root at boot time) will call its rm_rf_children() function
recursively (on each nested directory) and will exhaust its stack and
crash. For example, on Ubuntu 21.04:

------------------------------------------------------------------------
$ cd /tmp
$ perl -e 'use strict;
for (my $i = 0; $i < (1<<15); $i++) {
mkdir "A", 0700 or die;
chdir "A" or die; }'
------------------------------------------------------------------------

Then, as root (warning: this command may delete important files and
directories in /tmp; it is normally executed at boot time only):

------------------------------------------------------------------------
# systemd-tmpfiles --remove
Segmentation fault (core dumped)
------------------------------------------------------------------------

We have not fully explored the implications of this vulnerability;
however, we noticed that:

- at boot time, systemd executes "systemd-tmpfiles --create --remove
  --boot --exclude-prefix=/dev";

- systemd-tmpfiles first enters the "remove" phase, and subsequently
  enters the "create" phase;

- but if systemd-tmpfiles crashes during the "remove" phase, then it
  never enters the "create" phase;

- and it fails to create the files and directories (specified in
  /usr/lib/tmpfiles.d/*.conf) that it should create at boot time;

- for example, on Ubuntu 21.04, systemd-tmpfiles fails to create the
  directory /run/lock/subsys; but because /run/lock is world-writable,
  attackers can create their own /run/lock/subsys; and because various
  legacy packages and daemons write into /run/lock/subsys as root, the
  attackers may create arbitrary files via symlinks in /run/lock/subsys.

Last-minute note: it seems impossible to trigger this vulnerability in
systemd-tmpfiles versions before commit e535840 ("tmpfiles: let's bump
RLIMIT_NOFILE for tmpfiles") from February 2019.

========================================================================

Thank you very much! We are at your disposal for questions, comments,
and further discussions.

With best regards,

--
the Qualys Security Advisory team
```