~~Bug 1157880 - (CVE-2021-35938) VUL-0: CVE-2021-35938: rpm: races with chown/chmod/capabilties calls during installation~~

Status: RESOLVED WONTFIX

Classification: Novell Products
Product: SUSE Security Incidents
Component: Incidents
Version: unspecified
Hardware: Other Other

Priority: P3 - Medium **Severity**: Normal
Target Milestone: ---
Assigned To: Michael Schröder
QA Contact: Security Team bot

URL: https://smash.suse.de/issue/248049/
Whiteboard: CVSSv3.1:SUSE:CVE-2021-35938:6.5:(AV...
Keywords:

Depends on:
Blocks:
Show dependency tree / graph

- Create test case
- Clone This Bug

Reported: 2019-11-27 11:50 UTC by Malte Kraus
Modified: 2022-09-16 08:19 UTC (History)
CC List: 9 users (show)

See Also:
Found By: ---
Services Priority:
Business Priority:
Blocker: ---

Flags: gabriele.sonnu: needinfo? (mls)

Attachments

Verbose reproducer (26.57 KB, application/x-executable) Details
2020-12-16 10:34 UTC, Johannes Segitz

More reliable, quiet reproducer (26.57 KB, application/x-executable) Details
2020-12-16 10:34 UTC, Johannes Segitz

Add an attachment (proposed patch, testcase, etc.) View All

Note
You need to log in before you can comment on or make changes to this bug.

Malte Kraus 2019-11-27 11:50:51 UTC Description

When RPM installs a file/directory, it uses path-based operations afterwards to set the desired permissions and credentials.

When the directory this is happening in is owned by an unprivileged user, that user can escalate privileges to root by exchanging the file/directory with a symbolic link to a security-critical file/directory.

(When RPM installs large files where writing takes some time, exploits are perfectly reliable. E.g. mine hasn't failed yet during installation of the pcp-testsuite package in openSUSE, which has many large files in directories not owned by root.)

Malte Kraus 2019-11-27 11:53:09 UTC Comment 2

Adding lnussel to CC since he was involved in handling the same/similar issue previously.

Also see here for the O_PATH/proc contortions I did for similar problems in chkstat:

https://build.suse.de/package/view_file/SUSE:Maintenance:13179/permissions.SUSE_SLE-15-SP1_Update/0007-chkstat-fix-privesc-CVE-2019-3690.patch?expand=1

Johannes Segitz 2020-12-02 09:34:25 UTC Comment 3

I stumbled over this too while working through issues with our %post/%pre sections. There we tell our maintainers to not use insecure shell constructs but let rpm do the heavy lifting. So it would be good if rpm actually does that in a secure way ;)

This has been idling for a long time. We nowadays have a 90 day deadline for disclosing security issues. As this has been reported a long time ago I usually would set it lower (e.g. 4 weeks), but since this is probably hard to solve since rpm can't assume to always have the capabilities of newer kernels and with the upcoming holidays I'll just set the usual 90 day deadline.

CRD: 2021-03-02
https://en.opensuse.org/openSUSE:Security_disclosure_policy

Before we make this public we should coordinate with other distros. We can either do this two weeks before this becomes public via the distros mailinglist or the rpm upstream project can coordinate this themselves.

Ludwig Nussel 2020-12-02 10:16:43 UTC Comment 4

looks like a dup of #943457

Johannes Segitz 2020-12-03 13:43:03 UTC Comment 5

(In reply to Ludwig Nussel from comment #4)
Thanks for the hint. That's a saga *sigh*

I fear this is more of the same (so not fixed in 943457) as Malte tested on Factory, which has the patches. He left the company, so I'll dig into this as the report here unfortunately misses details

Johannes Segitz 2020-12-16 10:34:00 UTC Comment 6

Created attachment 844524 [details]
Verbose reproducer

I checked the older bug and while it's similar this is different. Unfortunately Maltes report lacked some details and he's not in the company anymore, so I had to dig into rpm. I've seen this behavior already a while ago before Malte filed this issue, but didn't pursue it further since Malte started to work on this.

The main issue here is fsmSetmeta, while the earlier issue has patches for problems in fsmVerify and for the way files are created in the first place.

Here's how to reproduce:
- Create a file that belongs to root and has strict permissions, e.g.
  `cp /etc/shadow /etc/shadow2`
- Start the exploit as
  `uid=462(pcpqa) gid=458(pcpqa) groups=458(pcpqa)`
  and provide the directoy you control and rpm will operate in and the file you want to take control
  `./exploit /var/lib/pcp/testsuite/ /etc/shadow2`
  exploit_verbose gives more verbose output, but that makes winning the race harder
- As root:
  `rpm -U --force pcp-testsuite-4.3.4-7.2.x86_64.rpm`

You might need to do this more than once, but for me this works at least 50% of the time and I'm sure that this can be made much more reliable.

If you want to make sure this succeeds run rpm in gdb and break on fsmSetmeta to step through:

in rpmPackageFilesInstall

```
            /* Set permissions, timestamps etc for non-hardlink entries */
            if (!rc && setmeta) {
                rc = fsmSetmeta(fpath, fi, plugins, action, &sb, nofcaps);
            }
```

We call fsmSetmeta with that path where we want to change permissions. sb is a struct stat, but its' based on the information from the rpm file

Depending on which race you win you can get to two results:
- fsmSetmeta:

  ```
      if (!rc && !S_ISLNK(st->st_mode)) {
          rc = fsmChmod(path, st->st_mode);
      }
  ```

  Since st is based on the information in the rpm it doesn't matter that we switched path to a link.

  Output of the verbose exploit

  ```
  [+] watching /var/lib/pcp/testsuite/
  [+] back from read
  [+] read 32
  [+] Got name: 000 len 16
  [+] added link to /etc/shadow2
  [+] back from read
  [+] read 32
  [+] Got name: 000 len 16
  [+] skipping link 000
  [+] back from read
  [+] read 48
  [+] Got name: 000.out;5fd9dbe6 len 32
  [+] added link to /etc/shadow2
  [+] /etc/shadow2 changed, have a look :)
  ```

  If we go down this road the file we specified is now readable:

  ```
  -rwxr-xr-x  1 root root   ?                                 1.3K Dec 16 11:04
/etc/shadow2
  ```

- Other case in fsmSetmeta

  ```
       if (!rc && !getuid()) {
          rc = fsmChown(path, st->st_mode, st->st_uid, st->st_gid);
       }
  ```

  in fsmChown there's also a check for a link, but again this is from the rpm

  ```
  static int fsmChown(const char *path, mode_t mode, uid_t uid, gid_t gid)
  {
      int rc = S_ISLNK(mode) ? lchown(path, uid, gid) : chown(path, uid, gid);
  ```

  Output of the verbose exploit

  ```
  (venv) pcpqa@linux-v0tl:/tmp> ./exploit_verbose /var/lib/pcp/testsuite/
/etc/shadow2
  [+] watching /var/lib/pcp/testsuite/
  [+] back from read
  [+] read 32
  [+] Got name: 001;5fd9dbe6 len 16
  [+] added link to /etc/shadow2
  [+] /etc/shadow2 changed, have a look :)
  ```

  Now we're owner of the file we specified:

  ```
-rw-r--r--  1 pcpqa pcpqa  ?                                 1.3K Dec 16 11:04
/etc/shadow2
  ```

This will not be easy to fix at this path-based operations are insecure if the user controls a component of the path. One way would be to (carefully) get a FD and then operate on this

Johannes Segitz 2020-12-16 10:34:27 UTC Comment 7

Created attachment 844525 [details]
More reliable, quiet reproducer

Johannes Segitz 2021-03-01 15:29:48 UTC Comment 8

contacted upstream about this, will make it public this or next week

Panu Matilainen 2021-03-02 11:08:14 UTC Comment 9

Any chance for the source of the exploit(s)? Just to make it easier to see and play with?

Johannes Segitz 2021-03-02 13:04:25 UTC Comment 10

(In reply to Panu Matilainen from comment #9)
I'll check if I still have it. I lost a disk with some disposable VMs that was not included into my backup rotation and I fear the source was on there. But it wasn't much more than setting up inotify and then acting as fast as possible once this triggers.

Panu is looking into this, moving
CRD: 2021-03-16
preliminary to prevent the bot from freaking out starting today

Johannes Segitz 2021-03-03 09:57:35 UTC Comment 11

(In reply to Johannes Segitz from comment #10)
Unfortunately the source went down with the VM :(

Johannes Segitz 2021-04-01 08:41:59 UTC Comment 12

Upstream maintainer is looking into this. Because if this I restart the
CRD: 2021-06-30
to have a reasonable chance to fix this

Johannes Segitz 2021-04-28 13:19:38 UTC Comment 13

reminder ping :)

Johannes Segitz 2021-06-16 09:13:45 UTC Comment 14

Any progress on this? We're getting close to the CRD. Thanks

Johannes Segitz 2021-06-30 12:14:16 UTC Comment 15

CRD reached, making it public to give the community a chance to work on this

Gabriele Sonnu 2021-12-24 08:33:23 UTC Comment 16

Hi, any update on this?

Michael Schröder 2022-08-29 15:02:29 UTC Comment 19

IIRC upstream rpm has fixed this by rewriting most of the rpm unpacking machinery, but I don't see how we can backport this.

Panu Matilainen 2022-08-30 05:56:38 UTC Comment 20

Yup. FWIW, this is the bulk of the upstream fix for this set of symlink vulnerabilities: https://github.com/rpm-software-management/rpm/pull/1919

Stoyan Manolov 2022-09-16 08:19:00 UTC Comment 21

This fix cannot be easily backported. The upstream fixes are scheduled for the next rpm major release and they are currently in beta phase. We will come back to this upon releasing the next rpm major version.

First Last Prev Next This bug is not in your last search results.

Format For Printing - XML - Clone This Bug - Top of page