Snap policy module fails to identify snaps if SCM_CREDENTIALS are missing from PA_COMMAND_AUTH request

Bug #1895928 reported by James Henstridge on 2020-09-17

This bug affects 1 person 264

Affects	Status	Importance	Assigned to	Milestone
pulseaudio (Ubuntu)	Fix Released	Undecided	Avital Ostromich	

Bug Description

This bug was discovered while debugging the non-deterministic behaviour of the example program attached to bug 1886854.

The snap policy module currently uses the credentials passed in an SCM_CREDENTIALS control message attached to the PA_COMMAND_AUTH request sent by the client. Credentials will only be attached to the message if at least one end of the connection has set the SO_PASSCRED socket option.

In normal operation, both the client and server set SO_PASSCRED on their sockets, so this functions normally. The test program on the other bug used an alternative client library that didn't set SO_PASSCRED, which leads to a race between the client sending the PA_COMMAND_AUTH request and the server calling setsockopt().

If the client wins, the server will receive a message with an empty SCM_CREDENTIALS control message (pid=0, uid=65534, gid=65534). When the snap policy module gets these empty credentials, it would try to look up the confinement of pid 0. As there is no such process, the module decides that the client is not a snap.

As any lookup via process ID is inherently racy, a better solution would be to use aa_getpeercon() to retrieve the client's security label in pa_native_protocol_connect(), and store it in the pa_client struct. We can then look up this in the policy module when it comes time to do the check.

See original description

CVE References

2020-16123

James Henstridge (jamesh) on 2020-09-17

description:updated

John Johansen (jjohansen) wrote on 2020-09-19: #1

I should note that aa_getpeercon() is currently a wrapper around

```
 rc = getsockopt(fd, SOL_SOCKET, SO_PEERSEC, buf, &optlen);
```

it checks that apparmor is enabled before hitting the interface (other LSMs could be using it), and then splits the context that is returned into a label and mode.

Once the LSM stacking replacement interface is properly defined it will use that if available instead.

You can hit the interface directly, but if you do you should perform similar checks and processing.

James Henstridge (jamesh) wrote on 2020-09-22: #2

I had a look through how upstream Pulse Audio uses SCM_CREDENTIALS, and I don't believe this bug extends to anything there: all uses grant privileges based on matching uid or gid, so this attack would result in less privilege.

It's a problem for us because we were reducing privilege on a match rather than increasing it.

James Henstridge (jamesh) wrote on 2020-09-22: #3

pa-race_0.1_amd64.snap (1.4 MiB, application/octet-stream)

Attached is a test snap based on the example program from bug 1886854. I've turned it into a standalone snap with strict confinement an audio-playback plug, and a launcher script that sets environment variables to let it find the Pulse Audio socket and cookie file.

The program attempts to load the "module-null-sink" plugin, and then remove it. The expected output is something like this:

```
    2020/09/22 15:46:21 PulseAudio connection created successfully
    2020/09/22 15:46:21 Couldn't load module, error message: PulseAudio
error: commandLoadModule -> Access denied
```

However, on repeated runs, it will occasionally produce output like the following:

```
    2020/09/22 15:46:33 PulseAudio connection created successfully
    Loaded Module sucessfully at index: 27
```

... indicating that it has not been detected as a snap. These occasions will be paired with Pulse Audio logging "[pulseaudio] module-snap-policy.c: AppArmor profile could not be retrieved."

James Henstridge (jamesh) wrote on 2020-09-22: #4

pa-race.tar.gz (1.1 KiB, application/x-tar)

Snapcraft project for pa-race snap.

James Henstridge (jamesh) wrote on 2020-09-22: #5

pulsaudio_13.99.1-1ubuntu3.6_13.99.1-1ubuntu3.7.diff (obsolete) (18.9 KiB, text/plain)

Here is a draft fix for the bug as a patch against 20.04's Pulse Audio. If this looks okay, it should be fairly easy to port to xenial, bionic, and groovy.

With these changes, we use aa_getpeercon() to retrieve the peer's AppArmor label at connection time and store it in the pa_client struct. The policy module then uses this label rather than the "SCM_CREDENTIALS -> process ID -> aa_gettaskcon" method it currently does.

The pa-race test snap consistently returns access denied with these changes applied.

I had a go at trying to slot in a fix for bug 1886854 (allow classic snaps to load modules), but I'm getting protocol errors when switching those hooks to async mode. So this is just a fix for the SCM_CREDENTIALS security issue.

James Henstridge (jamesh) wrote on 2020-09-23: #6

pulsaudio_13.99.1-1ubuntu3.6_13.99.1-1ubuntu3.7.diff (28.5 KiB, text/plain)

The problems with the protocol error problems were due to me dropping too much of the 0409 patch: in addition to adding the pa_creds fields to the pa_client struct, it was also fixing a bug in the command argument parsing after continuing from an asynchronous hook invocation.

This version includes the fix for bug 1886854, to allow classic snaps to invoke module and daemon control related commands again.

Alex Murray (alexmurray) wrote on 2020-09-24: #7

This has been assigned CVE-2020-16123.

Alex Murray (alexmurray) wrote on 2020-09-24: #8

The patch in comment #6 looks good - we will just need to edit the changelog entry to list the actual assigned CVE ID from above and then backport to xenial, bionic and groovy.

James Henstridge (jamesh) wrote on 2020-09-29: #9

As far as testing goes, here are a few things to help test the updated package:

1. After installing the test snap, running "pa-race" repeatedly should always fail. This is in comparison with the old package where it will occasionally succeed.

2. Running "/snap/pa-race/current/bin/bug" directly (i.e. the same program outside of confinement) should always succeed.

3. Running "snap run --shell some-classic-snap" and then "/snap/pa-race/current/bin/bug" from that shell should always succeed.

Some of the tests from the plan in bug 1781428 would be worth verifying too.

Avital Ostromich (avital) wrote on 2020-10-28: #10

Hello James,

Thank you so much for the patch, test snap and testing information! There are issues applying the patch in Xenial because there are some remaining references to the creds and creds_valid variables (which are removed from the pa_client struct) in src/modules/trust-store/module-trust-store.c, which contains functionality for Ubuntu Touch. Should the changes to remove the variables be backported to that file as well?

Thank you,
Avital

James Henstridge (jamesh) wrote on 2020-11-03: #11

This is a bit tricky due to divergence in the package after we removed Trust Store post-xenial. I _think_ this is the best way to integrate it:

1. Keep xenial's 0409 patch as is: the current version in focal and from my debdiff represents the parts of that original patch that were still needed after removing the Trust Store specific parts.

2. Add the 0410 patch from my debdiff. This is a new patch rather than a replacement for xenial's 0410, so may need renumbering. I don't think it should conflict with the other truststore patches.

3. Take the 0700 patch from my debdiff as a replacement for xenial's 0450 patch.

Alex Murray (alexmurray) on 2020-11-12

Changed in pulseaudio (Ubuntu):
status:New → In Progress
assignee:James Henstridge (jamesh) → Avital Ostromich (avital)

Launchpad Janitor (janitor) wrote on 2020-11-23: #12

This bug was fixed in the package pulseaudio - 1:13.99.2-1ubuntu2.1

```
---------------
pulseaudio (1:13.99.2-1ubuntu2.1) groovy-security; urgency=medium

  * SECURITY UPDATE: don't rely on SCM_CREDENTIALS to detect snap confined
    clients (LP: #1895928)
    - d/p/0409-pa-client-peer-credentials.patch: drop patch
    - d/p/0409-fix-arg-parsing-after-async-hook.patch: remains of old 0409
      patch not related to pa_creds.
    - d/p/0410-pa-client-peer-apparmor-label.patch: new patch, records
      AppArmor label in pa_client struct for native connections using
      aa_getpeercon.
    - d/p/0702-add-snappy-policy-module.patch: use the AppArmor
      label in the pa_client rather than looking it up via the process ID
      from SCM_CREDENTIALS.
    - CVE-2020-16123
   * Don't block classic snaps from module loading/unloading (LP:
#1886854)
    - d/p/0702-add-snappy-policy-module.patch: replace
      deny_to_snaps_hook with a version that allows classic snaps.

 -- James Henstridge <email address hidden> Thu, 05 Nov 2020 16:46:59
-0500

Changed in pulseaudio (Ubuntu):
status:In Progress → Fix Released
```

Avital Ostromich (avital) on 2020-12-03

```
information type:Private Security → Public Security
```

See full activity log

To post a comment you must log in.