

Bug ~~1196556~~ (CVE-2022-29527) VUL-0: CVE-2022-29527: amazon-ssm-agent: creates world-writable sudoers file during runtime (race condition)

Status: RESOLVED FIXED

• [Create test case](#)

Classification: Novell Products

• [Clone This Bug](#)

Product: SUSE Security Incidents

Component: Audits

Reported: 2022-02-28 14:34 UTC by Matthias Gerstner

Version: unspecified

Modified: 2022-10-20 13:56 UTC ([History](#))

Hardware: Other Other

CC List: 8 users ([show](#))

Priority: P3 - Medium **Severity:** Normal

See Also:

Target Milestone: ---

Found By: ---

Assigned To: Sean Marlow

Services Priority:

QA Contact: Security Team bot

Business Priority:

URL: <https://smash.suse.de/issue/324948/>

Blocker: ---

Whiteboard: CVSSv3.1:SUSE:CVE-2022-29527:8.4:(AV:...

Flags: sean.marlow: needinfo? (adrian.glaubitz)

Keywords:

Depends on:

Blocks:

Show dependency [tree](#) / [graph](#)

Attachments

[Add an attachment](#) (proposed patch, testcase, etc.)

Note

You need to [log in](#) before you can comment on or make changes to this bug.

Matthias Gerstner 2022-02-28 14:34:36 UTC

Description

+++ This bug was initially created as a clone of [Bug #1196135](#)

Found this via traces in the changes file of amazon-ssm-agent. The agent implement in Golang creates sudoers.d rules during runtime in:

```
agent/session/utility/utility_unix.go:
```

```
...
```

```
// createSudoersFileIfNotPresent will create the sudoers file if not present.
func (u *SessionUtil) createSudoersFileIfNotPresent(log log.T) error {
```

```

// Return if the file exists
if _, err := os.Stat(sudoersFile); err == nil {
    log.Infof("File %s already exists", sudoersFile)
    _ = u.changeModeOfSudoersFile(log)
    return err
}

// Create a sudoers file for ssm-user
file, err := os.Create(sudoersFile)
if err != nil {
    log.Errorf("Failed to add %s to sudoers file: %v",
appconfig.DefaultRunAsUserName, err)
    return err
}
defer file.Close()

    if _, err := file.WriteString(fmt.Sprintf("# User rules for %s\n",
appconfig.DefaultRunAsUserName)); err != nil {
        return err
    }
    if _, err := file.WriteString(fmt.Sprintf("%s ALL=(ALL) NOPASSWD:ALL\n",
appconfig.DefaultRunAsUserName)); err != nil {
        return err
    }
}
...

```

So basically a root replacement user is created here which doesn't sound too good an idea.

Interesting in the same source file is the following piece of code:

```

...
// changeModeOfSudoersFile will change the sudoersFile mode to 0440 (read
only).
// This file is created with mode 0666 using os.Create() so needs to be
updated to read only with chmod.
func (u *SessionUtil) changeModeOfSudoersFile(log log.T) error {
    fileMode := os.FileMode(sudoersFileMode)
    if err := os.Chmod(sudoersFile, fileMode); err != nil {
        log.Errorf("Failed to change mode of %s to %d: %v",
sudoersFile, sudoersFileMode, err)
        return err
    }
    log.Infof("Successfully changed mode of %s to %d", sudoersFile,
sudoersFileMode)
    return nil
}
...

```

So this file is created with mode 0666? Hopefully the author only thought of the mode not yet modified by the process's umask, otherwise this would be a great race condition involving a local root exploit.

I looked at this package during runtime, but probably lacking the AWS environment it just pukes around and ends up in segmentation faults (yes!). So I can't really tell at the moment what the practical implications of this are.

Matthias Gerstner 2022-03-21 14:09:47 UTC

[Comment 1](#)

Assigning to aosthof, since there is currently no dedicated maintainer for amazon-ssm-agent. Can you help sorting out this issue or find someone more suitable to take care of it?

Alexander Osthof 2022-03-23 09:28:47 UTC

[Comment 2](#)

Hi Matthias, in fact we do have a maintainer set for amazon-ssm-agent which is the 'public-cloud-team' group in IBS. Is this what you were looking for?

Matthias Gerstner 2022-03-23 10:15:22 UTC

[Comment 3](#)

(In reply to [aosthof@suse.com](#) from [comment #2](#))

> Hi Matthias, in fact we do have a maintainer set for amazon-ssm-agent which is th

Ah I've only been looking in OBS since this is about Factory (at least for now). Does the public-cloud-team also have an associated email address in Bugzilla? IBS does not show one.

Alexander Osthof 2022-03-23 11:14:26 UTC

[Comment 4](#)

Yes, it's [public-cloud-maintainers@suse.de](#)

Matthias Gerstner 2022-03-24 09:17:44 UTC

[Comment 5](#)

Okay, thanks for helping out, reassigning the bug accordingly.

Alexander Osthof 2022-03-29 11:49:33 UTC

[Comment 6](#)

Sean, can you please check on an EC2 instance if the amazon-ssm-agent creates a sudoers file with mode 0666? Thank you!

Matthias Gerstner 2022-03-29 12:05:30 UTC

[Comment 7](#)

(In reply to [aosthof@suse.com](#) from [comment #6](#))

> Sean, can you please check on an EC2 instance if the amazon-ssm-agent creates a s

Note that you will probably have to use a debugger or strace to notice this. An strace of the agent when the code mentioned in [comment 0](#) runs would be most helpful to see what is really going on.

Sean Marlow 2022-03-29 13:58:26 UTC

[Comment 8](#)

The file is created in the case that a user does not already exist and cannot be created and a sudoers file does not already exist. Currently the code creates a file with default permissions (0x666):

```
...  
// Create a sudoers file for ssm-user  
file, err := os.Create(sudoersFile)  
...
```

Then the file is written to and finally the permissions are change to (0x440) after writing. That means there's a window of time where the file exists and is writable by all.

I assume this should probably be something like:

```
...  
// Create a sudoers file for ssm-user  
file, err := os.OpenFile(sudoersFile, os.O_RDWR|os.O_CREATE, 0640)  
...
```

That way the file is only writable by owner. Until the mode gets changed to readonly after writing.

Sean Marlow 2022-03-29 15:02:07 UTC

Comment 9

```
I guess this could actually be:
...
file, err := os.OpenFile(sudoersFile, os.O_WRONLY|os.O_CREATE, 0640)
...
```

Matthias Gerstner 2022-03-30 08:16:51 UTC

Comment 10

Thank you for confirming this. Can you report this (preferably privately) to upstream or should I do it?

Sean Marlow 2022-03-30 13:28:25 UTC

Comment 11

Yeah, I can reach out.

Sean Marlow 2022-04-01 15:49:35 UTC

Comment 12

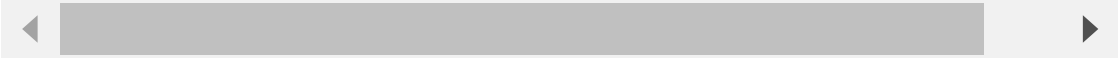
Amazon security team is working on a fix which is expected to be released in mid-April.

Matthias Gerstner 2022-04-04 08:53:42 UTC

Comment 13

(In reply to sean.marlow@suse.com from [comment #12](#))

> Amazon security team is working on a fix which is expected to be released in mid-



Super, thanks for taking care of this.

Sean Marlow 2022-04-06 13:28:17 UTC

Comment 14

The patch has been released:

<https://github.com/aws/amazon-ssm-agent/commit/0fe8ae99b2ff25649c7b86d3bc05fc037400aca7>

Is in the latest version:

<https://github.com/aws/amazon-ssm-agent/releases/tag/3.1.1208.0>

Matthias Gerstner 2022-04-06 13:45:23 UTC

Comment 15

Can you ask upstream if they intend to assign a CVE to this? It's a possible local root escalation.

We should also backport the fix to any maintained codestreams of amazon-ssm-agent. I'll ask colleagues from my team to look post information what backports to make.

Robert Schweikert 2022-04-06 13:47:17 UTC

Comment 16

We'll do a version update instead of a backport.

Matthias Gerstner 2022-04-06 13:49:27 UTC

[Comment 17](#)

If they didn't already do this upstream they should also now remove the misleading comment:

```
```
```

```
// This file is created with mode 0666 using os.Create() so needs to be updated to
read only with chmod.
```
```

This should no longer be true.

Sean Marlow 2022-04-06 14:00:35 UTC

[Comment 18](#)

The comment still exists in code so I pinged AWS security about that and if there will be a CVE to reference.

Matthias Gerstner 2022-04-07 07:54:38 UTC

[Comment 19](#)

Thinking some more about this it will not be a local root exploit on many distributions, because /etc/sudoers.d is often hardened to be not world-readable.

A quick look around different distributions shows that only Debian Linux has 755 permissions for /etc/sudoers.d and would be affected.

We should backport / update with the fix anyway, because creating a world-writable file in there is bad style and a security hazard in any case.

John Paul Adrian Glaubit 2022-04-19 11:10:43 UTC

[Comment 20](#)

(In reply to Robert Schweikert from [comment #16](#))

> We'll do a version update instead of a backport.

I have submitted the latest upstream version which includes the fix to Factory now:

> <https://build.opensuse.org/request/show/970747>

Shall I also submit it to SLE? Do we need an ECO for that?

Matthias Gerstner 2022-04-19 12:29:24 UTC

[Comment 21](#)

(In reply to adrian.glaubit@use.com from [comment #20](#))

> I have submitted the latest upstream version which includes the fix to Factory nc



Great, thanks!

> Shall I also submit it to SLE? Do we need an ECO for that?

I consider this important enough to update SLE. An ECO would not be needed, the bug reference is enough.

Is there a CVE available yet? Reference the CVE would be important for updating SLE.

I'll involve colleagues from reactive security to help with the maintenance process.

Marcus Meissner 2022-04-19 12:38:56 UTC

Comment 22

I filed a CVE request with Mitre.

John Paul Adrian Glaubit 2022-04-20 09:01:31 UTC

Comment 23

(In reply to Marcus Meissner from [comment #22](#))

> I filed a CVE request with Mitre.

OK, thanks. Let me know once we have a CVE ID, then I'll add it to the changelog.

OBSbugzilla Bot 2022-04-20 14:40:03 UTC

Comment 24

This is an autogenerated message for OBS integration:
This bug (1196556) was mentioned in
<https://build.opensuse.org/request/show/971130> Factory / amazon-ssm-agent

Swamp Workflow Management 2022-05-03 19:16:52 UTC

Comment 28

SUSE-SU-2022:1510-1: An update that fixes one vulnerability is now available.

Category: security (important)

Bug References: 1196556

CVE References: CVE-2022-29527

JIRA References:

Sources used:

openSUSE Leap 15.4 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

openSUSE Leap 15.3 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

SUSE Linux Enterprise Module for Public Cloud 15-SP4 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

SUSE Linux Enterprise Module for Public Cloud 15-SP3 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

SUSE Linux Enterprise Module for Public Cloud 15-SP2 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

SUSE Linux Enterprise Module for Public Cloud 15-SP1 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

SUSE Linux Enterprise Module for Public Cloud 15 (src): amazon-ssm-agent-3.1.1260.0-150000.5.9.2

NOTE: This line indicates an update has been released for the listed product(s). At times this might be only a partial fix. If you have questions please reach out to maintenance coordination.

Matthias Gerstner 2022-07-05 11:48:58 UTC

Comment 30

Sean, can you please also provide the fix for SLE-12 as pointed out in [comment 29](#)?

Sean Marlow 2022-09-09 11:10:16 UTC

Comment 34

Not sure for SLE12. It requires golang >= 1.15.12. Adrian does SLE12 have a sufficient version of golang?

Marcus Meissner 2022-09-09 11:17:25 UTC

Comment 35

we have go 1.19 even.

Sean Marlow 2022-09-09 13:49:58 UTC

[Comment 36](#)

Seems good to push the same version to SLE12 then.

Swamp Workflow Management 2022-10-19 13:20:33 UTC

[Comment 40](#)

SUSE-SU-2022:3654-1: An update that fixes one vulnerability is now available.

Category: security (important)

Bug References: 1196556

CVE References: CVE-2022-29527

JIRA References:

Sources used:

SUSE Linux Enterprise Module for Public Cloud 12 (src): amazon-ssm-agent-3.1.1260.0-4.27.2

NOTE: This line indicates an update has been released for the listed product(s). At times this might be only a partial fix. If you have questions please reach out to maintenance coordination.

Alexander Osthof 2022-10-20 13:56:25 UTC

[Comment 41](#)

Released.

[First](#) [Last](#) [Prev](#) [Next](#) *This bug is not in your last search results.*

[Format For Printing](#) - [XML](#) - [Clone This Bug](#) - [Top of page](#)