Bug 1177362 - (CVE-2020-8029) VUL-0: CVE-2020-8029: skuba: Insecure handling of private key

Status: REOPENED

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

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

URL:
Whiteboard: CVSSv3.1:SUSE:CVE-2020-8029:4.0:(AV:L...
Keywords:

Depends on:
Blocks:
Show dependency tree / graph

- Create test case
- Clone This Bug

Reported: 2020-10-06 11:42 UTC by Johannes Segitz
Modified: 2021-04-16 07:45 UTC (History)
CC List: 5 users (show)

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

Flags: rfernandezlopez: needinfo? (kkaempf)

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.

Johannes Segitz 2020-10-06 11:42:25 UTC Description

```
In internal/pkg/skuba/deployments/ssh/kubelet.go

124         if err := t.target.UploadFile(keyPath,
filepath.Join(kubernetes.KubeletCertAndKeyDir, kubernetes.KubeletServerKeyName));
err != nil {
125                 return err
126         }
127         if _, _, err := t.silentSsh("chmod", "0400",
filepath.Join(kubernetes.KubeletCertAndKeyDir, kubernetes.KubeletServerKeyName));
err != nil {
128                 return err
129         }

this uploads the key world readable, then sets secure permissions.

POC:
On node that will be joined as non-root user:
sles@jseg-caasp45-worker1:/tmp> id
uid=1000(sles) gid=100(users) groups=100(users),494(cdrom)
sles@jseg-caasp45-worker1:/tmp> while true; do cat /var/lib/kubelet/pki/kubelet.key
2>/dev/null; done
<now join the node>
 -----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEA0wG5F5sI9LzKrba8bUnV8vzjGY3C1lJp3BHpsM/BIUbS3BnH
<snip>

Please have a look. As we're upstream for this I would assign a CVE from our pool
```

Johannes Segitz 2020-10-06 11:43:16 UTC Comment 1

```
This is an embargoed bug. This means that this information is not public.

Please do NOT:
- talk to other people about this unless they're involved in fixing the issue
- make this bug public
- submit this into OBS (e.g. fix Leap/Tumbleweed) until this bug becomes public
(e.g. no EMBARGOED tag on the header)

Consult with security team if you think that the issue is public and the bug is
still private (e.g. subject still contains "EMBARGOED"). Please do NOT make the bug
public yourself.

Please be aware that the SUSE:SLE-15-SP3:GA codestream is available via OBS, so do
NOT submit there before this is public.

These are the steps that are asked from you:
1, Your primary responsibility is to submit a fix for this issue. Here's a how-to
for submitting packages for maintenance releases in IBS:

https://confluence.suse.com/display/maintenance/How+to+Submit+Packages+or+Containers+
   Apart from the GA codestreams mentioned above, you can submit to IBS anytime.
This is private and allows us to start testing as soon as possible.
2, We also want to fix openSUSE if it's affected.
   $ is_maintained $PACKAGE
   will tell you if the package is inherited from SLES or if it is branched for
openSUSE. There are two cases:
   - It's coming from SLES: The update will automatically be released for openSUSE.
Nothing to do for you.
   - It's branched for openSUSE: You need to submit AFTER the bug became public, to
the current openSUSE codestreams.
   For openSUSE Factory please submit to the devel project of you package AFTER the
bug became public.

Security will then take the following steps:
```

- We wait for your submission and package them into an incident for QA testing. The QA tester might reach out to you if they find issues with the update.
- Once the coordinated release date (CRD), the date this issue should become public, is reached (or for internal findings: once we're done testing), we remove the EMBARGOED tag from this bug and publish the updates.
- Only if the bug here is public you may submit to public repositories (OBS).

You can contact us at:

* IRC: irc.suse.de #security
* RocketChat: https://chat.suse.de/channel/security
* Email: security-team@suse.de

Internal CRD: 2021-01-04 or earlier

Comment 2

Adding Jenting to the issue.

@Johannes, I think this is indeed an issue. As for the approach for mitigating it, would setting a `umask` before the file is written be enough?

Johannes Segitz 2020-10-08 08:11:34 UTC Comment 3

(In reply to Rafael Fernández López from comment #2)
Thanks for the quick reaction!

Umask: You would need to try this. If this happens in one session than this should work. I would prefer an approach where the file is explicitly created with secure permissions first (e.g. 0600), written and then set to 0400 afterwards. This makes it implicit and has less potential to break.

Please use CVE-2020-8029 for tracking

Rafael Fernández López 2020-10-26 11:02:24 UTC Comment 4

Adding David Ko.

jenting hsiao 2020-10-27 02:49:21 UTC Comment 5

(In reply to Johannes Segitz from comment #3)

> (In reply to Rafael Fernández López from comment #2)
> Thanks for the quick reaction!
>
> Umask: You would need to try this. If this happens in one session than this
> should work. I would prefer an approach where the file is explicitly created
> with secure permissions first (e.g. 0600), written and then set to 0400
> afterwards. This makes it implicit and has less potential to break.
>
> Please use CVE-2020-8029 for tracking

```
124         if err := t.target.UploadFile(keyPath,
filepath.Join(kubernetes.KubeletCertAndKeyDir, kubernetes.KubeletServerKeyName));
err != nil {
125                 return err
126         }
127         if _, _, err := t.silentSsh("chmod", "0400",
filepath.Join(kubernetes.KubeletCertAndKeyDir, kubernetes.KubeletServerKeyName));
err != nil {
128                 return err
129         }
```

The above code upload the local key to the remote machine with file permission set to 0644 first and then set to 0400. So, the secure way is to create file permission "0600" and then set to "0400", or to create file permission "0400" directly. Am I unstandstand it right?

Johannes Segitz 2020-10-30 13:42:50 UTC Comment 6

(In reply to jenting hsiao from comment #5)
The secure way is to set permissions on the file first, yes. If you need to write it later go for 0600

Rafael Fernández López 2020-10-30 15:04:45 UTC Comment 7

(In reply to Johannes Segitz from comment #6)

> (In reply to jenting hsiao from comment #5)
> The secure way is to set permissions on the file first, yes. If you need to
> write it later go for 0600

I think we can check whether we can add a mode parameter to `UploadFile` (and to `UploadFileContents`). Since `UploadFileContents` writes the contents in the remote system by doing a `echo <contents> | base64 -d -w0 > /target/file`, I think we could add the umask in this very same command so it impacts the redirection in case the redirection ends up creating fhe file because it didn't exist.

What do you think? Would this be enough?

jenting hsiao 2020-11-06 01:04:33 UTC Comment 8

Change to use `install` command.
1. generate the file with specific permission mode `install -m <mode> /dev/null /target/file`
2. write the content by `echo <contents> | base64 -d -w0 > /target/file`

In the private key case, it'd be
1. `install -m 600 /dev/null /var/lib/kubelet/pki/kubelet.key
2. `echo <contents> | base64 -d -w0 > /var/lib/kubelet/pki/kubelet.key

v4.2 PR: https://github.com/SUSE/skuba/pull/1428
v4.5 PR: https://github.com/SUSE/skuba/pull/1416

code merged, waits to be released.

Swamp Workflow Management 2020-12-11 17:16:39 UTC Comment 14

SUSE-SU-2020:3761-1: An update that solves four vulnerabilities and has 11 fixes is now available.

Category: security (important)
Bug References:
1172270,1173055,1173165,1174219,1174951,1175352,1176225,1176578,1176903,1176904,11773
CVE References: CVE-2020-15106,CVE-2020-8029,CVE-2020-8564,CVE-2020-8565
JIRA References:
Sources used:
SUSE CaaS Platform 4.5 (src): caasp-release-4.5.2-1.8.2, cri-o-1.18-1.18.4-4.3.2, etcd-3.4.13-3.3.1, helm2-2.16.12-3.3.1, helm3-3.3.3-3.8.1, kubernetes-1.18-1.18.10-4.3.1, patterns-caasp-Management-4.5-3.3.1, skuba-2.1.11-3.10.1, velero-1.4.2-3.3.1

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.

Johannes Segitz 2021-04-16 07:45:32 UTC Comment 22

If that's correct then we can close this. Thanks for the effort