利用 CVE-2022-0847 实现容器逃逸

CVE-2022-0847 漏洞分析与利用

网络与系统安全实验室

2022 年 6 月

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目录

CVE-2022-0847

- 2022 年 03 月 7 日，安全研究员 Max Kellermann 披露了一个 Linux 内核本地提权漏洞 CVE-2022-0847，命名为 Dirty Pipe[1]
- 由于 pipe_buffer 结构体未正确初始化，攻击者可利用此漏洞向只读文件的页缓存写入数据
- 本报告将利用 CVE-2022-0847 实现容器逃逸

splice() 系统调用

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

splice() API

```
1 ssize_t splice(int fd_in, off64_t *off_in, int fd_out, off64_t *off_out,
      size_t len, unsigned int flags);
```

- splice() 用于在两个文件描述符间移动数据，而无需内核态和用户态的内存拷贝，但需要借助管道实现
- Linux 的管道实现为 pipe_buffer 环，每个 pipe_buffer 都引用一个内存页，如果最后一次写入没填满一页，那么后续的写会继续添加到该页
- 而如果 splice() 文件到管道，内核首先加载数据到页缓存（page cache），然后创建一个指向页缓存的 pipe_buffer，若该页未满，额外的写入数据不会继续在该页写入，因为该页不属于管道

CVE-2022-0847 漏洞成因

- Linux 5.8 为 pipe_buffer 的 flags 引入新的 flag PIPE_BUF_FLAG_CAN_MERGE，用来表示该页能否合并（即后续写入是否会继续在该页添加）
- 而 pipe_buffer 的 flags未进行正确初始化，如此一来，splice() 用到的 pipe_buffer 若 PIPE_BUF_FLAG_CAN_MERGE 在此之前被置位则不会被清零，攻击者可将任意数据添加到该 pipe_buffer 指向的页缓存的剩余空间，实现覆写只读文件的效果

CAP_DAC_READ_SEARCH

- 用于打开文件的系统调用 openat() 可以拆分为两个系统调用[1]
  name_to_handle_at() 用于得到目标文件的句柄
  open_by_handle_at() 根据传入的文件句柄打开目标文件，得到文件描述符
- CAP_DAC_READ_SEARCH 的权限包括绕过读取文件和文件夹的权限检查、执行 open_by_handle_at() 系统调用
- 若容器被赋予 CAP_DAC_READ_SEARCH，则能绕过容器隔离，遍历宿主机文件系统并读取任意文件

[1] https://man7.org/linux/man-pages/man2/open_by_handle_at.2.html

CAP_DAC_READ_SEARCH 容器逃逸

文件句柄

在 64 位系统中，文件句柄长度为 8 个字节，其中前 4 个字节为文件的 inode 号

- 若容器被赋予 CAP_DAC_READ_SEARCH，攻击者可在容器内部遍历宿主机 inode，找到目标文件 inode 号（占据句柄前 4 字节）[2]
- 随后暴力破解文件句柄后 4 字节，即可找到指向该文件的句柄
- 最后调用 open_by_handle_at() 即可通过该句柄打开目标文件，得到文件描述符

结合 CVE-2022-0847 容器逃逸

- CVE-2022-0847 利用代码能覆写只读文件，但由于容器的隔离机制，容器内部只能访问容器的文件系统，无法覆写宿主机的文件
- CAP_DAC_READ_SEARCH 容器逃逸能读取宿主机上的任意文件，但不能写入宿主机文件

两者结合

将 CVE-2022-0847 利用代码与 CAP_DAC_READ_SEARCH 容器逃逸结合，首先通过遍历宿主机文件系统拿到目标文件的句柄，调用 open_by_handle_at() 得到目标文件的文件描述符，将该文件的部分内容通过 splice() 系统调用发送到管道，最后向管道继续写入内容即可覆写宿主机文件，代码见：🔗。

演示

```
root@vagrant:/home/vagrant# echo "hello world" > flag.txt
root@vagrant:/home/vagrant# chmod 0400 flag.txt
root@vagrant:/home/vagrant# ll flag.txt
-r-------- 1 root root 12 Jun  8 13:13 flag.txt
root@vagrant:/home/vagrant# cat flag.txt
hello world
root@vagrant:/home/vagrant#
```

图: 创建 root 用户只读文件

演示

```
root@vagrant:/home/vagrant# docker run --rm -it -v $(pwd):/exp --cap-add=CAP_DAC_READ_SEARCH ubuntu
root@85a087798003:/# /exp/dp /home/vagrant/flag.txt
Dumping /home/vagrant/flag.txt successfully, content:

hello world

root@85a087798003:/# /exp/dp /home/vagrant/flag.txt 1 abcdefghij
Dumping /home/vagrant/flag.txt successfully, content:

hello world

Overwrite /home/vagrant/flag.txt successfully.
root@85a087798003:/# /exp/dp /home/vagrant/flag.txt
Dumping /home/vagrant/flag.txt successfully, content:

habcdefghij

root@85a087798003:/#
exit
root@vagrant:/home/vagrant# cat flag.txt
habcdefghij
root@vagrant:/home/vagrant# ll flag.txt
-r-------- 1 root root 12 Jun  8 13:18 flag.txt
root@vagrant:/home/vagrant#
```

图: 容器内部覆写宿主机只读文件

总结

- 将 CVE-2022-0847 漏洞与已有 CAP_DAC_READ_SEARCH 逃逸路径结合，容器内部能覆写宿主机上的只读文件
- 仅能写入页缓存，若缓存没有写回，则不能实现持久化，缓存失效/重启后文件恢复

参考文献

[1] KELLERMANN M. The Dirty Pipe Vulnerability[EB/OL]. (2022-03-07). https://dirtypipe.cm4all.com/.

[2] KRAHMER S. Shocker: docker PoC VMM-container breakout[EB/OL]. (2014-06-12). http://stealth.openwall.net/xSports/shocker.c.

谢 谢