

社区首页 > 专栏 > Docker Rootless 在非特权模式下运行 Docker

Docker Rootless 在非特权模式下运行 Docker

发布于 2021-12-31 08:23:48 0 0 0

文章被收录于专栏：Se7en的架构笔记

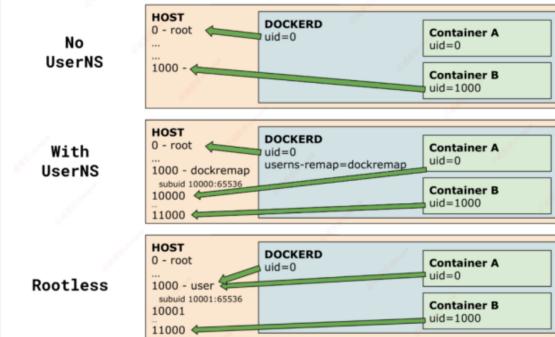
举报

Docker Rootless 基本概念

Rootless 模式允许以非 root 用户身份运行 Docker^{守护进程}（dockerd）和 容器⁴，以缓解 Docker 守护进程和容器运行时潜在的漏洞。Rootless 模式是 Docker v19.03 版本作为实验功能引入的，在 Docker v20.10 版本 GA。

Rootless 模式目前对 Cgroups 资源控制、AppArmor 安全配置、Overlay 网络、存储驱动器还有一定的限制，暂时还不能完全取代“Rootful” Docker。关于 Docker Rootless 的详细信息参见 Docker 官方文档 [Run the Docker daemon as a non-root user (Rootless mode)](https://docs.docker.com/engine/security/rootless/#limiting-resources)

Rootless 模式利用 user namespaces 将容器中的 root 用户和 Docker 守护进程（dockerd）用户映射到宿主机上的非特权用户范围内，Docker 此前已经提供了 —usersns-remap 标志支持了相关能力，提升了容器的安全隔离性。Rootless 模式在此之上，让 Docker 守护进程也运行在映射的用户名空间中。



实践验证

环境准备

本文使用 CentOS^{7.5} 操作系统的虚拟机进行实验。

代码语言: javascript

```
1 [root@demo ~]# cat /etc/redhat-release
2 CentOS Linux release 7.5.1804 (Core)
```

Se7en258



文章 95

获赞 267

专栏 1

作者相关精选

换一批

WebAssembly 在云原生中的实践指南

使用 ClusterResourceSet 为 Cluster API...

Elastic Stack 实战教程 3：快照备份与恢复

目录

环境准备

创建用户

安装依赖

启动 Docker 守护进程

添加站长进交流群

领取 10 元无门槛券，专享最新干货技术

[社区公告] 云开发开箱挑战赛 查看详情 >

相关产品与服务

容器镜像服务

容器镜像服务 (Container Container Registry, SCR) 为您提供便捷统一、高性能的容器镜像仓库分发...

产品介绍 产品文档

HOT 精选特惠 用云无忧

创建用户

代码语言: javascript

```
1 useradd rootless
2 echo 123456 | passwd rootless --stdin
```

安装依赖

Rootless 模式可以在没有 root 权限的情况下运行 Docker 守护进程和容器，但是需要安装 newuidmap 和 newgidmap 工具，以便在用户命名空间下创建从属(subordinate)用户组的映射(remapping)。通过以下命令安装 newuidmap 和 newgidmap 工具。

代码语言: javascript

```
1 cat <<EOF | sudo sh -
2 curl -o /etc/yum.repos.d/vbatts-shadow-utils-newuidmap-el7.repo https://copr.fedorainfracloud.org/coprs/vbatts/
3 yum install -y shadow-utils46-newuidmap
4 cat <<EOF >/etc/systemctl.conf
5 user_max_user_namespaces = 28633
6 EOF
7 systemctl --system
8 EOF
```

UID/GID 映射配置

从属用户和组的映射由两个配置文件来控制，分别是 /etc/subuid 和 /etc/subgid。使用以下命令为 rootless 用户设置 65536 个从属用户和组的映射。

代码语言: javascript

```
1 echo "rootless:100000:65536" | tee /etc/subuid
2 echo "rootless:100000:65536" | tee /etc/subgid
```

对于 subuid，这一行记录的含义是：用户 rootless，在当前的 user namespace 中具有 65536 个从属用户，用户 ID 为 100000-165535，在一个子 user namespace 中，这些从属用户被映射成 ID 为 0-65535 的用户，subgid 的含义和 subuid 相同。

比如让用户 rootless 在宿主机上只是一个具有普通权限的用户，我们可以把他的一个从属 ID (比如 100000) 分配给容器所属的 user namespace，并把 ID 100000 映射到该 user namespace 中的 uid 0。此时即便容器中的进程具有 root 权限，但也仅是在容器所在的 user namespace 中，一旦到了宿主机中，顶多也就只有 rootless 用户的权限而已。

安装 Rootless Docker

切换到 rootless 用户。

代码语言: javascript

```
1 su - rootless
```

执行以下命令安装 Rootless Docker。

代码语言: javascript

```
1 curl -sSL https://get.docker.com/rootless | sh
```

安装成功后显示如下内容。

```
+ PATH=/home/rootless/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/sbin:/home/rootless/.local/bin:/home/rootless/bin
+ /home/rootless/bin/dockerd-rootless-setup.sh install
[INFO] System not detected, dockerd-rootless.sh needs to be started manually:
PATH=/home/rootless/bin:/bin:/usr/sbin:$PATH dockerd-rootless.sh
[INFO] Creating CLT context "rootless"
[INFO] Successfully created context "rootless"
[INFO] Make sure the following environment variables are set (or add them to ~/.bashrc):
# WARNING: systemd not found. You have to remove _XDG_RUNTIME_DIR_ manually on every logout.
export _XDG_RUNTIME_DIR=/home/rootless/.docker/run
export PATH=/home/rootless/bin:$PATH
export DOCKER_HOST=tcp://home/rootless/.docker/run/docker.sock
```

将以下内容添加到 ~/.bashrc 文件中，添加完以后使用 source ~/.bashrc 命令使环境变量生效。

代码语言: javascript

```
1 export _XDG_RUNTIME_DIR=/home/rootless/.docker/run
2 export PATH=/home/rootless/bin:$PATH
3 export DOCKER_HOST=tcp://home/rootless/.docker/run/docker.sock
```

启动 Docker 守护进程

使用以下命令启动 Docker 守护进程。

代码语言: javascript

```
1 | dockerd-rootless.sh
```

运行容器

使用以下命令启动一个 nginx 容器，并将 80 端口映射到宿主机的 8080 端口。

代码语言: javascript

```
1 | docker run -d -p 8080:80 nginx
```

查看容器。

代码语言: javascript

```
1 | [rootless@demo ~]$ docker ps
2 | CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
3 | f3b204c97e84 nginx "/docker-entrypoint..." 9 minutes ago Up 9 minutes 0.0.0.0:8080->80/tcp, :::8080->
```

访问容器。

代码语言: javascript

```
1 | [rootless@demo ~]$ curl http://localhost:8080
2 |
3 | # 返回结果 Nginx 欢迎界面
4 | <!DOCTYPE html>
5 | <html>
6 | <head>
7 | <title>Welcome to nginx!</title>
8 | <style>
9 | html { color scheme: light dark; }
10 | body { width: 35em; margin: 0 auto; }
11 | font-family: Tahoma, Verdana, Arial, sans-serif; }
12 | </style>
13 | </head>
14 | <body>
15 | <h1>Welcome to nginx!</h1>
16 | <p>If you see this page, the nginx web server is successfully installed and
17 | working. Further configuration is required.</p>
18 |
19 | <p>For online documentation and support please refer to
20 | <a href="http://nginx.org/">http://nginx.org/http://nginx.com/</p>
23 |
24 | <p>Thank you for using nginx.</p>
25 |
26 | </html>
```

参考阅读

- [【 首页安全 #】拾遗 – Rootless Container初探](#) (<https://developer.aliyun.com/article/700923>)
- [\[Run the Docker daemon as a non-root user \(Rootless mode\)\]](#) (<https://docs.docker.com/engine/security/rootless/>)
- [\[Experimenting with Rootless Docker\]](#) (<https://medium.com/@torontig/experimenting-with-rootless-docker-416c9ad8c0d6>)
- [\[浅谈Docker的安全性支持（下篇）\]](#) (<http://blog.itpub.net/31559359/viewspace-264596/>)
- [\[Docker v2.10 核心功能介绍和实践\]](#) (<https://mp.weixin.qq.com/s/MF21vWL722Wpxw0mR7A>)
- [\[shadow-utils-newuidmap\]](#) (<https://copr.fedorainfracloud.org/coprs/vbatts/shadow-utils-newuidmap/>)
- [\[Hardening Docker Daemon with Rootless Mode\]](#) (<https://www.youtube.com/watch?v=uWURUtlqLiqQ>)
- [\[Linux® Namespace : User\]](#) (<https://www.cnblogs.com/sparkdev/p/9442838.html>)
- [\[理解 docker 容器中的 uid 和 gid\]](#) (<https://www.cnblogs.com/sparkdev/p/9614164.html>)
- [\[隔离 docker 容器中的用户\]](#) (<https://www.cnblogs.com/sparkdev/p/9614326.html>)

本文参与 腾讯白媒体同步曝光计划，分享自微信公众号。

原创发表于 2021-12-26，如有侵权请联系 cloudcommunity@tencent.com 删除

<https://www.toutiao.com/i/700923/> 网络安全 容器镜像服务 容器 编程算法

评论



[登录](#) 后参与评论

推荐阅读

编辑精选文章

[MySQL是如何保证数据一致性的？](#)

换一批 ↗

Podman 保姆级使用教程，太顶了！

◇ 容器服务 编程算法 容器 容器镜像服务

cockpit-podman 软件包也作为 cockpit 插件可集成于 Web UI 中，实现 Web UI 管理容器。



[#开讲易场](#) 2022/04/09 ◇ 19.3K ◇ 1

Docker 大势已去，Podman 即将崛起

◇ 容器 容器镜像服务 ioping 编程算法

点上“上方”字源码，“选择”设为星标”被抛前浪，还是后浪？能浪的浪，才是好浪！每天 10:33 更新文章，每天掉亿点点头... 源码精品专栏 原创 | Java 2021 超神之路，很肝~ 中文详细注释的开源项目 RPC 框架 Dubbo 源码解析 网络应用框架 Netty 源码解析 消息中间件 RocketMQ 源码解析 数据库中间件...

[#道听途说](#) 2022/03/04 ◇ 889 ◇ 0

CentOS8 安装和使用podman

◇ ubuntu 镜像

使用rootless用户pull ubuntu镜像，竟然报这种错误。发现错误：

[#双鱼人](#) 2020/12/01 ◇ 1.9K ◇ 0

rootless Podman如何工作？【Programming】

◇ 容器 编程算法

在上一篇有关用户空间和Podman的文章中，我讨论了如何使用Podman命令来启动具有不同用户名空间的不同容器，从而更好地分离容器。Podman还利用用户名空间来以无根模式运行。基本上，当非特权用户运行Podman时，该工具将设置并加入用户名...



[#土豆](#) 2019/11/24 ◇ 2.3K ◇ 0

Docker安全入门与实战（一）

◇ 容器镜像服务 容器 安全 Linux

与其他介绍Docker的文章不同，由本文开启的系列文章将专注于Docker安全研究，一共分为6部分。



[#Oxuhao](#) 2022/06/21 ◇ 1K ◇ 0

隔离 Docker 容器中的用户

◇ 容器镜像服务 容器 网络 编程算法 Linux

笔者在前几期讲解 docker 容器中的 uid 和 gid 介绍了 docker 容器中的用户与宿主机上的用户的关系，得出的结论是：docker 默认没有隔离宿主机用户和容器中的用户。如果你已经了解了 Linux 的 user namespace 技术(参考《Linux Namespace : User》)，那...



[#星际拓云](#) 2022/07/19 ◇ 3.4K ◇ 1

Docker

◇ 容器 容器镜像服务 安全 测量扫描服务
与其他介绍Docker的文章不同，由本文开启的系列文章将专注于Docker安全研究。一共分为6部分。



千年的铁树开了花。聊聊account

◇ 容器 容器镜像服务 api 网站
account真是一個千年鐵樹，但神奇的是它又是在開新花。从我使用电脑第一起就需要记住账号，只有输入了账号密码才可以登录实验室那台Windows玩扫雷游戏。如今大红大紫的零信任重要的组成部分IAM也在该账号，零信任需要基于账号来回答一个灵魂拷...


6.Docker镜像与容器安全最佳实践

◇ 容器 容器 容器的安全 linux
描述: 在企业中信息系统安全与业务是同样重要, 随着传统运维方式向着容器化运维方式的转变, 当下企业里通常都会采用Docker来运行容器化部署和承载业务, 由于运维人员或者开发人员对容器安全的关注较少, 只是简单认为容器是有隔离和限制的, 就算是容器被黑...


Dockert使用

◇ 容器 容器镜像服务 socket编程
执行这个命令后, 镜本就会自动的将一切准备工作做好, 并且把Docker CE 的Edge版本安装在系统中。


浅谈日常使用的 Docker 底层原理~三大底座

◇ 容器镜像服务 容器 进程 镜像 原理
适合的读者, 对Docker有过简单了解的朋友, 想要进一步了解Docker容器的朋友。


docker namespaces

◇ 容器镜像服务
<https://docs.docker.com/engine/security/users-remap/#prerequisites>


浅析Docker运行安全

◇ 编程算法 容器 容器镜像服务 linux ipv6
AppArmor 主要的作用是设置某个可执行程序的访问控制权限, 可以限制程序 读/写某个目录/文件, 打开/读/写网络端口等等。


彻夜熬夜！17 个 Docker 常见疑难杂症解决方案汇总！

◇ 容器镜像服务 容器 unix
[问题截图] 今天通过云监控系统, 看到公司其中一台服务器的磁盘快慢, 跳脚上去了看了一下, 发现 /var/lib/docker 这个目录特别大。由上述原因, 我们都知道, 在 /var/lib/docker 中存储的都是关于容器的存储, 所以也不能随便的将其删除。


在Docker守护进程停机期间保持容器运行（即重启Docker时，正在运行的容器不会停止）

◇ 容器镜像服务 容器
在默认情况下, 当 Docker 守护进程终止时, 它将关闭正在运行的容器。不过, 我们可以配置守护进程, 以便在该守护进程不可用时 容器仍在运行。这种功能称为实时恢复, 实时还原选项有助于减少由于守护进程崩溃、计划中断或升级而导致的容器停机时间。


24 个 Docker 疑难杂症处理技巧

◇ 容器 容器镜像服务 ipv6 tcp/udp unix
默认情况下系统会将 Docker 容器放在 /var/lib/docker 目录下。


docker 系列：底层知识

◇ linux 容器镜像服务 容器 nginx
Docker 采用的是 C/S 架构, 使用 REST API, UNIX 套接字或网络接口进行通信。一般客户端会和 Docker 服务运行在同一台机器上, 像我们平常使用的 docker build, pull, run 等命令就是发送到本地客户端上的, 本地客户端再发送给 Docker 服务端。另外, ...


Docker超详细版（基础+进阶）

◇ 容器镜像服务
Docker 1. 简介 1.1 什么是虚拟化 在计算机中, 虚拟化 (Virtualization) 是将计算机的各种实体资源, 如服务器、网络、内存及存储等, 予以抽象, 转换后呈现出来, 打破实体结构间的不可切割的障碍, 使用户可以比原本的组态更好的方式来应用这些资源。这些资源的新鲜部分是不受现有资源的...


云安全 | 容器基础设施所面临的风险学习

◇ socket编程 安全 unix https
下图是 Docker 官方给出的架构图, 里面包括了 Docker 客户端、Docker 容器所在的宿主机和 Docker 镜像仓库三个部分。


Docker常用命令备忘录

◇ 容器镜像服务 容器 登录 镜像 网络
yum install docker 逐步安装 docker -v 查看版本 systemctl start/stop/restart 查看 docker 启动、停止、重启、状态、开启自动启动 docker docker info 查看需要信息 docker --help 查看帮助 docker images 查看所有的镜像 (在 /var/lib/docker 目录) docker search imagename 搜索镜像 docker pull centos...


社区

技术文章
技术问答
技术沙龙
技术视频
学习中心
技术百科
技术专区

活动

白媒同步曝光计划
邀请作者入驻
白荐上首页
技术竞赛

资源

技术周刊
社区问答
开发者手册
开发者实验室

关于

社区规范
免责声明
联系我们
友情链接

腾讯云开发者

扫码关注腾讯云开发者
微信号: 云代码

热门产品

域名注册
云服务器
区块链服务
消息队列
网络加速
云数据库
域名解析
云存储
视频直播

热门推荐

人脸识别
腾讯会议
企业云
CDN加速
视频通话
图像分析
MySQL 数据库
SSL 证书
语音识别

更多推荐

数据安全
负载均衡
短信
文字识别
云点播
商标注册
小程序开发
网站监控
数据迁移



