

**Linux实验报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学 期 | 2024-2025学年第一学期 | | | | |
| 课 程 | Linux操作系统 | | | | |
| 专　 业 | 计算机科学与技术 | | 班 　 级 | | 2班 |
| 学生姓名 | 陈伟 | | 学　 号 | | 22219111239 |
| 组 号 | 无 | 同组学生姓名 | | 无 | |
| 实验（项目）名称 | | Linux操作系统（11次）实验 | | | |
| 任课教师 | 周旭立 | | 成 绩 | |  |

# Docker

## 一、实验目的

docker

## 二、实验环境（以后略）

装有Ubuntu 的工作站或桌面的系统，若不涉及图形界面的话，也可以是服务器系统。如果，不考虑图形界面的的话，也可以是其它版本的Linux系统。

出于学习和研究的目的，在虚拟机控制下的Linux系统可能更好。在虚拟机软件的控制下，可在同一Windows或Linux系统下安装（本书使用或其它）多个版本的Linux、Unix或Windows系统，这对于虚拟化、云计算、大数据和集群等部署与研究具有特殊意义。

## 三、实验方法与注意事项（以后略）

为了保证各种实验的成功和顺利进行，建议实验者以root用户登录系统（可以使用sudo -s 命令后输入密码进行root）。当然，以一个具有管理员属性的普通用户身份工作，也是可以的。在此种情况下，管理命令可能需要使用sudo命令来执行。但不论怎么样，实验室内的实验系统与环境是共用的，请不要在系统内做对系统或对其他用户不安全的事情。

## 四、实验过程

#什么是docker

Docker 是一个开源的容器化平台，用于开发、打包和运行应用程序。它通过提供一个轻量级的虚拟化方式，确保应用程序能够以独立的容器形式运行，从而实现跨平台的兼容性

#为什么要用docker

Docker 通过提供一致的开发和运行环境、简化部署、节省资源以及支持微服务等特性，极大地提高了应用开发、部署和维护的效率，特别是在云原生应用和现代 DevOps 流程中

#docker有哪些应用场景

开发环境的统一

CI/CD 流程

微服务架构

跨平台和跨环境部署

容器化旧应用

高效的资源利用和隔离

DevOps 和基础设施自动化

安全性增强

多版本支持

数据处理与科学计算

#什么是docker 镜像

是一个只读的模板，包含运行应用程序所需的所有依赖和配置。例如，一个 Python 应用程序的镜像可能包括 Python 运行时、代码和相关的库。

类似于虚拟机的快照，镜像是容器运行的基础。

#什么是docker 容器

是镜像的运行实例。它是一个轻量级、可移植的运行环境，封装了应用程序及其依赖。

容器与宿主机共享操作系统内核，因此比传统的虚拟机效率更高。

#解释docker 镜像 和 docker 容器之间有什么区别。

Docker 镜像是一个只读的模板，包含了运行应用程序所需的一切内容：操作系统、运行时环境、库、依赖项以及应用程序本身的代码。镜像本质上是一个静态的文件，可以看作是应用的快照。

镜像是不可变的，一旦创建，镜像中的内容不会改变，所有修改都会在容器内进行，而不是在镜像本身。

Docker 容器是镜像的运行实例，是一个可执行的环境，用于运行应用程序。它是一个轻量级的、隔离的运行时环境，包含了从镜像启动后生成的应用状态、进程、文件系统等。

容器是动态的，可读写的，它基于镜像运行，并可以对应用程序的状态、数据等进行修改。

#打个比喻，如果说docker images像 面向对象编程中的一个类（class），那docker 容器就像是面向对象编程中的什么？

对象

#什么是docker registry

**Docker Registry** 是一个用于存储和管理 Docker 镜像的服务。它是一个集中式的存储库，可以让用户推送（上传）和拉取（下载） Docker 镜像。Docker Registry 的核心功能是为 Docker 镜像提供版本控制、存储、检索等功能。

#描述 Docker 镜像、容器和仓库的关系

镜像是容器的基础，容器通过镜像启动并运行，而镜像又是存储在仓库中的。仓库则充当了镜像的存储和分发平台，帮助开发者共享和使用镜像。

#docker container里面是否有自己的操作系统

Docker 容器本身**不包含完整的操作系统**