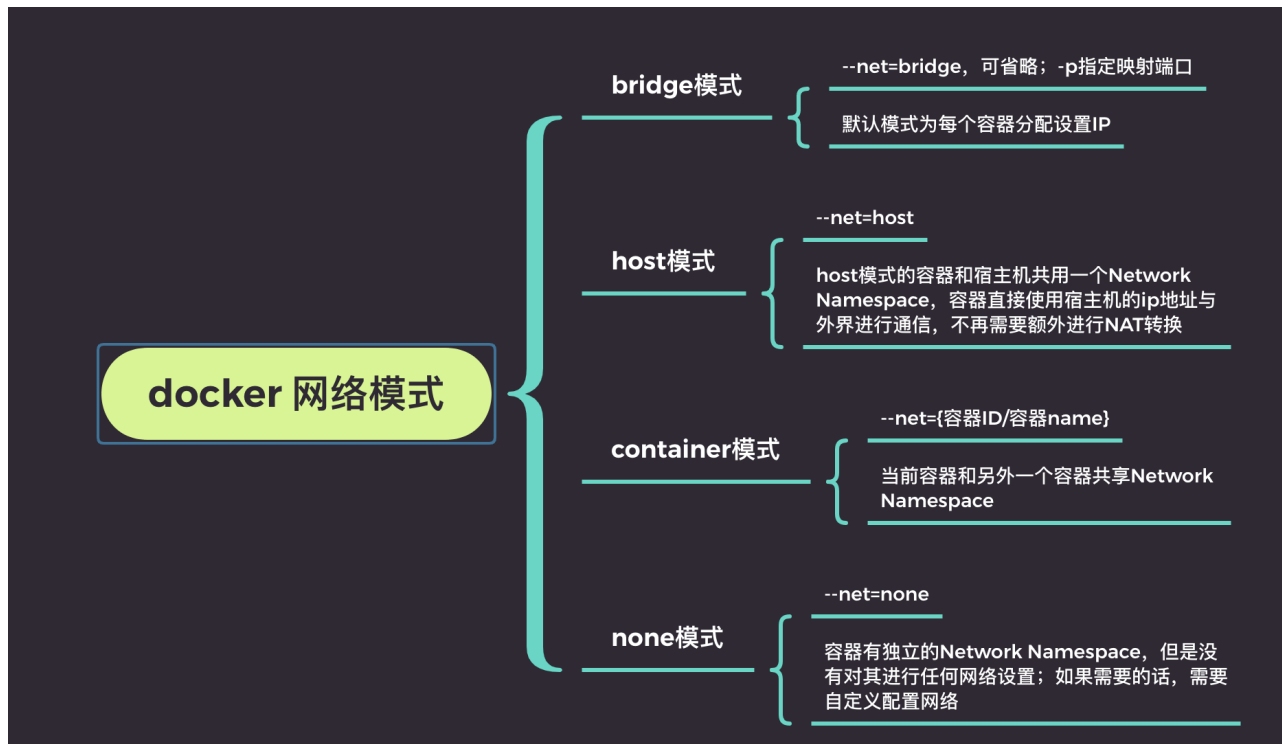


## 1. 了解 Docker 的容器网络，简述 Docker 4 种容器网络的差别，并提交对应文档。

1. host模式的容器和宿主机共用一个“Network Namespace”，容器直接使用宿主机的ip地址与外界进行通信，不再需要额外进行NAT转换
2. Container模式创建的容器会与指定的容器共享IP和端口范围
3. None模式关闭了容器的网络功能
4. Bridge默认模式为每个容器分配设置IP



## 2. 简述 Docker 容器与虚拟机的区别，它们在哪些场景使用更合理，并提交对应文档。

### Docker 容器与虚拟机的区别

- 虚拟机：虚拟机是通过Hypervisor(虚拟机管理系统，常见的有VMWare workstation、VirtualBox)，虚拟出网卡、cpu、内存等虚拟硬件，再在其上建立虚拟机，每个虚拟机是个独立的操作系统，拥有自己的系统内核。
- Docker容器：容器是利用namespace将文件系统、进程、网络、设备等资源进行隔离，利用cgroup对权限、cpu资源进行限制，最终让容器之间互不影响，容器无法影响宿主机。

## 3. 安装配置 Docker-compose 并截图。

```
[chenjing@chenjings-mbp ceshiren_homework % docker-compose --version
Docker Compose version v2.15.1
chenjing@chenjings-mbp ceshiren_homework % ]
```

## 4. 独立完成 Docker-compose 搭建 mariadb 和 testlink，让服务正常启动起来，过程上传截图。

```
1 version: '2'
2
3 services:
4   mariadb:
5     image: 'bitnami/mariadb:latest'
6     environment:
7       - ALLOW_EMPTY_PASSWORD=yes
8       - MARIADB_USER=bn_testlink
9       - MARIADB_DATABASE=bitnami_testlink
10  testlink:
11    image: 'bitnami/testlink:latest'
12    environment:
13      - ALLOW_EMPTY_PASSWORD=yes
14      - TESTLINK_DATABASE_USER=bn_testlink
15      - TESTLINK_DATABASE_NAME=bitnami_testlink
16    ports:
17      - '5003:8080'
18      - '444:443'
19    depends_on:
20      - mariadb
21
```

```
chenjing@centos ~$ docker-compose up -d
[+] Running 2/0
  Container chenjing-mariadb-1 Running
  Container chenjing-testlink-1 Running
chenjing@centos ~$ docker-compose top
chenjing-mariadb-1
  UID      PID      PPID     C    STIME   TTY      TIME      CMD
  1001     7297    7271    0    03:23   ?        00:00:01  /opt/bitnami/mariadb/sbin/mysqld --defaults-file=/opt/bitnami/mariadb/conf/my.cnf --basedir=/opt/bitnami/mariadb --datadir=/opt/bitnami/mariadb/data --socket=/opt/bitnami/mariadb/tmp/mysql.sock --pid-file=/opt/bitnami/mariadb/tmp/mysql.pid
chenjing-testlink-1
  UID      PID      PPID     C    STIME   TTY      TIME      CMD
  1001     7491    7448    0    03:23   ?        00:00:00  /opt/bitnami/apache/bin/httpd -f /opt/bitnami/apache/conf/httpd.conf -D FOREGROUND
  1001     7771    7491    0    03:23   ?        00:00:00  /opt/bitnami/apache/bin/httpd -f /opt/bitnami/apache/conf/httpd.conf -D FOREGROUND
  1001     7772    7491    0    03:23   ?        00:00:00  /opt/bitnami/apache/bin/httpd -f /opt/bitnami/apache/conf/httpd.conf -D FOREGROUND
  1001     7773    7491    0    03:23   ?        00:00:00  /opt/bitnami/apache/bin/httpd -f /opt/bitnami/apache/conf/httpd.conf -D FOREGROUND
  1001     7774    7491    0    03:23   ?        00:00:00  /opt/bitnami/apache/bin/httpd -f /opt/bitnami/apache/conf/httpd.conf -D FOREGROUND
  1001     7775    7491    0    03:23   ?        00:00:00  /opt/bitnami/apache/bin/httpd -f /opt/bitnami/apache/conf/httpd.conf -D FOREGROUND
chenjing@centos ~$ docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
3706542646e   bitnami/testlink:latest             "/opt/bitnami/script-  38 minutes ago Up 37 minutes 8443/tcp, 0.0.0.0:444->443/tcp, 0.0.0.0:5003->8080/tcp   chenjing-testlink-1
afbf7566c200   bitnami/mariadb:latest              "/opt/bitnami/script-  38 minutes ago Up 37 minutes 3306/tcp                                chenjing-mariadb-1
f7774d2d7ed    selenium/node-chrome:4.0.0-rc-2-prerelease-20210923 "/opt/bin/entry_poin-  About an hour ago Up About an hour 0.0.0.0:5902->5900/tcp                  selenium-node
e58636061f     selenium/hub:4.0.0-rc-2-prerelease-20210923      "/opt/bin/entry_poin-  About an hour ago Up About an hour 0.0.0.0:5442->4442/tcp, 0.0.0.0:5443->4443/tcp, 0.0.0.0:5444->4444/tcp   selenium-hub
0bb63d2f2a88   jenkins/jenkins:lts                "/usr/bin/tini -- /u-  5 days ago      Up 2 hours    0.0.0.0:50000->50000/tcp, 0.0.0.0:5004->8080/tcp   myjenkins
chenjing@centos ~$
```