1. 内存资源限制
2. 重点技术

CGroup是Control Groups，它是Linux内核所提供的的一种可以限制，记录，隔离进程组所使用的物理资源（cpu，内存，i/o等）的机制，由于每个Docker容器相对于宿主机来说就是一个进程，因此Docker可以使用该机制来隔离使用的物理资源。

默认情况下，如果不对容器做任何限制，容器能够占用当前系统提供给容器的所有资源，因此我们必须对容器使用的资源做限制。Docker限制可以从内存，CPU，IO三方面入手。同时，还有一点，如果宿主机出现OOME异常，即内存溢出异常，任何进程都有可能被随机杀死，包括docker daemon，因此Docker调整了docker daemon的OOM优先级，以免直接被内核关闭。

1. 容器监控

应用在容器中启动后，要对容器做内存压力测试，了解正常业务下内存使用情况，然后才能进入生产环境中使用，因此一定要限制容器的内存上线，尽量保证主机的资源充足，一旦通过监控发现资源不足，就进行扩容或者对容器进行迁移。如果内存资源充足，尽量不要使用swap虚拟内存，它会导致内存计算复杂，对调度器不友好。

1. 容器内存设置方式

在Docker启动参数中，通过内存相关设置来限制容器的内存使用，参数的值是内存大小，单位是b,k,m,g，对应着bytes, KB, MB, GB

1. -m --memory

容器物理上能够使用的最大内存大小，容器绝对无法超过这个内存大小，最小为4MB。

1. –memory-swap

容器能够使用的swap大小，但是它还与-m的设置有关，设swap的设置为s，-m的设置为m，有如下设置：

1. s为正数，m为正数

容器总可用内存为s，其中物理内存为m，swap为s-m。

1. s为0，m为正数

相当于未设置swap，相当于物理内存为m，swap为2m。

1. s为unset，m为正数

同上。

1. s为-1，m为正数

若宿主机启用了swap，则容器可使用宿主机中所有swap资源。

注意，在容器中使用free命令看到的swap空间不是真正的大小。

1. –memory-wappiness

默认情况下，宿主机可以将容器使用的匿名页swap出来，这里设置一个0-100之间的数，代表允许swap出来的比例。

1. –memory-reservation

设置容器可以使用内存的软限制，容器在内存空间足够的情况下，可以超过该值，在内存空间不足的情况下，会回撤到该值以下。

1. –kernel-memory

容器可以使用的内核内存大小，最小值为4m。

1. –oom-kill-disable

是否运行Oomph时杀死容器，只有设置了-m，才可以设置为false，表示不需要杀死容器。如果没有设置-m，则必须设置为true，防止容器耗尽主机内存。

1. CPU限制
2. 限制方式

Docker容器对于CPU资源的限制可以在多核系统中限制容器使用哪些vCPU，限制方式有两种：

1. 有多个CPU密集型的容器竞争CPU时，设置各个容器能够使用的CPU时间相对比例。即动态地分配容器CPU时间，当容器不断地加入时，每个容器分配的时间会动态变化。
2. 以绝对的方式设置容器在每个调度周期内最多能够使用的CPU时间。
3. CPU限制设置
4. --cpuset-cpus=””

允许使用的CPU集，如果共有8个vCPU，则分为0-7，那么我们启动容器时设置的可以使用的CPU集可以设置为0-3，表示使用0-3这四个vCPU，或者0，表示只使用0这个vCPU。

1. -c, --cpu-shares=1024

CPU共享权值（相对权重），默认是1024。如何理解相对权重？每个容器都会设置自己的相对权重，它所能够使用的时间片比例就等于它的相对权重/所有容器的相对权重之和，每个容器的CPU使用时间是动态变化的。

1. --cpuset-mems=””

允许在上执行的内存节点。

1. --cpu-period=0

一个CPU中的调度周期，CFS周期的有效范围是1ms-1s，而调度周期的数值单位是微妙，因此数值范围是1000 – 1000000。

1. --cpu-quota=0

设置该容器在每个周期中可以使用的CPU时间，容器的CPU配额必须>=1ms，即>=1000，单位是微秒。

如下所示的例子：

docker run -it --cpu-period=50000 –cpu-quota=25000 ubuntu /bin/bash

解释： 一个CPU的可调度周期为50000微秒，该容器可以使用的CPU周期为25000，即使用一个CPU的一半调度时间。

docker run -it --cpu-period=10000 –cpu-quota=20000 ubuntu /bin/bash

解释： 一个CPU的可调度周期为10000，该容器的可调度周期为20000，即使用两个CPU的所有调度时间。

1. --cpus

限制容器可以使用的主机CPU个数，还可以指定小数。

1. 资源限制实验

使用一个压测镜像，能够不断地扩展使用的资源，如下所示：

docker run –name stress -it --rm -m 256m lorel/docker-stress-ng:latest stress -vm 2

解释：限制物理内存为256M，启动两个进程，使用docker stats stress查看资源占用情况，CPU利用率能够跑到100%左右，内存使用情况能够跑满256M。

docker run –name stress -it –rm –cpus 22 lorel/docker-stress-ng:latest stress –cpu 8

解释：限制使用2个CPU，容器中使用8个线程跑，结果是能够CPU能够跑满200%，内存占用不多。