1. 컨테이너

|  |
| --- |
| 컨테이너 학습 이유    소프트웨어 운영 플랫폼이 이렇게 바뀌고 있음  리눅스에서 컨테이너를 돌리는 이유  독립된 공간 형성  isolate 기능 지원  필요한만큼 HW 지원  즉, 리눅스 커널 기능을 사용하기 위해  그냥 프로그램과 컨테이너의 차이점  하는 일은 같지만 구조가 다르다.      도커를 쓰는 이유?  개발자가 만든 그대로 어디서든 돌릴 수 있다.  확장/축소가 쉽고 MSA, DevOps에 적합하다. |

2. 도커 컨테이너 설치

|  |
| --- |
| 컴퓨터(BareMetal 또는 Virtualmachine)  cpu: 2core, memory: 2GB이상  운영체제 (리눅스 또는 Windows)  도커 설치  도커 서비스 실행 |

3. virtualbox 설치 및 Network 구성

|  |
| --- |
|  |

3. ubuntu 20.04 설치

|  |
| --- |
|  |

4.centos7 설치

|  |
| --- |
|  |

5. docker설치

|  |
| --- |
| 계정 추가  루트계정 접속  $usermod -a -G docker guru |

6. 컨테이너

|  |
| --- |
| 컨테이너? 컨테이너 이미지?  컨테이너는 하나의 Application 프로세스    3개의 레이어를 가지고 만들어진 컨테이너 이다.  컨테이너 이미지는 여러 개의 레이어로 구성되어 하나의 Application이 잘 실행될 수 있도록 모아져 있는 이미지들의 조합이 하나의 컨테이너 이미지가 된다.    컨테이너 동작 방식 |
| 실습  1. Docker Hub에서 컨테이너 이미지 검색  2. 컨테이너 이미지 다운로드 후 image layer 보기  3. 컨테이너 실행하고 확인해 보기  $docker search nginx  $docker pull nginx:latest  $docker run --name web -d -p 80:80 nginx  $docker ps  $docker stop web  $docker rm web  $docker rmi nginx |
| 컨테이너 만들기  1. 무엇을 컨테이너로 만드는 건가요?      polyglot programming: 다양한 언어를 사용하여 프로그래밍 하는 것  컨테이너는 각각의 AplicationService에 맞춰 Container를 만들어 낸다.  2. 컨테이너 어떻게 만들어요? Dockerfile 이 뭐죠?  Dockerfile: Container를 만들 수 있도록 도와주는 명령어 집함        3. 내가 만든 컨테이너를 배포하려면?? |
| 실습  1. nodejs 애플리케이션 컨테이너 만들기 : hellojs  3. 만들어놓은 컨테이너 배포하기  $mkdir hellojs  $cd hellojs  $cat > hello.js  const http = require('http');  const os = require('os');  console.log("Test server starting...");  var handler = function(request, response) {  console.log("Received request from " + request.connection.remoteAddress);  response.writeHead(200);  response.end("Container Hostname: " + os.hostname() + "\n");  };  var www = http.createServer(handler);  www.listen(8080);  $vi dockerfile  FROM node:12  COPY hello.js /  CMD ["node", "/hello.js"]  $docker build -t hellojs:latest .  $docker images  2. 우분투 기반의 웹 서버 컨테이너 만들기  $cd  $mkdir webserver  $cd webserver  $vim dockerfile  FROM ubuntu:18.04  LABEL maintainer="Hyeonse Lee <dlgustp1487@naver.com>"  # install apache  RUN apt-get update \  && apt-get install -y apache2  RUN echo "TEST WEB" > /var/www/html/index.html  EXPOSE 80  CMD ["/usr/sbin/apache2ctl", "-DFOREGROUND"]  $docker build -t webserver:v1 .  $docker image ls  $docker run -d -p 80:80 --name web webserver:v1  $docker ps  $ curl localhost:80  $docker rm -f web  3. 만들어놓은 컨테이너 배포하기  $docker login  아이디 패스워드 입력  $docker images  $docker tag webserver:v1 lee1487/webserver:v1  $docker tag hellojs:latest lee1487/hellojs:latest  $docker images  $docker push lee1487/webserver:v1    [문제풀이]  1. webpage.sh 내용  #!/bin/bash  mkdir /htdocs  while:  do  /usr/games/fortune > /htdocs/index.html  sleep 10  done  2. dockerfile 내용  FROM debian  COPY webpage.sh /  RUN apt-get update \  && apt-get install -y fortune  CMD ["./webpage.sh"]  3. docker build 내용  docker build -t fortune:20.02 . |

7. docker registry

|  |
| --- |
| 1. 컨테이너 보관 창고(Registry)가 있어요?  Registry: 컨테이너 이미지를 저장하는 저장소  Docker Hub: hub.docker.com  Private Registry: 사내의 컨테이너 저장소    2. docker hub(registry)를 사용하고 싶어요    3. private registry를 구축하고 싶어요. |
| 1.hub.docker.com에 컨테이너 업로드 및 다운로드  $docker search httpd  docker search DockerHub에서 이미지 검색  $docker pull httpd  docker pull DockerHub에서 이미지 다운로드  $docker images  docker images 다운로드 받은 이미지 확인  $docker login  docker login DockerHub 로그인  $docker tag http:latest lee1487/httpd:latest  $docker images lee1487/httpd  $docker push lee1487/httpd:latest  2.private registry 운영하기  루트 접속  $docker ps  docker ps 동작 중인 컨테이너 확인  $docker run -d -p 5000:5000 --restart always --name registry registry:2  $hostname  $docker ps  $docker tag httpd:latest localhost:5000/httpd  $docker images localhost:5000/httpd  $docker push localhost:5000/httpd:latest  $cd /var/lib/docker/volumes/45fec1c6d6adb6c85c8ea6f8d0718c06f3545739a13e5e0e7527bcafdafa1cd3/\_data/docker/registry/v2/repositories |

8. 컨테이너 사용

|  |
| --- |
| 1. 컨테이너 이미지를 어떻게 사용해요?      2. 컨테이너를 실행하고 종료하는 명령을 알고 싶어요      pull -> create -> start  한번에 묶어놓은 것이 run 명령어  3. 동작중인 컨테이너를 관리하는 명령어가 궁금해요 |
| 실습  1. 컨테이너 이미지 관리 명령어 사용하기  2. 컨테이너 실행 및 운영하기  3. 컨테이너 종료하기  $docker search nginx  $docker pull nginx:1.14  $docker images  $docker mysql  $docker pull mysql:8  $docker images --no-trunc 이미지 명을 풀네임으로 출력  $docker create --name webserver nginx:1.14  $docker ps -a  $docker start webserver  docker start [컨테이너 명] 컨테이너 실행  $docker inspect webserver  docker inspect [컨테이너 명] 컨테이너 세부정보 확인  $docker inspect --format '{{.NetworkSettings.IPAddress}}' webserver  $alias cip="docker inspect --format '{{.NetworkSettings.IPAddress}}'"  alias [단축어 명]=”명령어” 자주 사용하는 명령어를 단축어로 등록  $cip webserver  $docker logs webserver  docker logs [컨테이너명] 컨테이너가 만들어 낸 로그 출력  새터미널  $docker logs -f webserver 실시간 모니터링 가능  $docker top webserver  docker top [컨테이너 명] 컨테이너에서 실행 중인 프로세스 출력  $docker exec -it webserver /bin/bash  docker exec -it [컨테이너 명] /bin/bash 컨테이너에 직접 접속하여 배시 쉘 사용  $cd /usr/share/nginx/html/  $echo "Hyeonse's HOMEPAGE" > index.html  $exit  $curl 172.17.0.3  $docker stop webserver  $docker rm webserver  docker rm [컨테이너 명] 컨테이너 삭제    [문제풀이]  1. docker search apache  docker pull httpd  2. docker run --name web -d httpd  3. docker ps  4. docker inspect web  5. curl 172.17.0.3  <html><body><h1>It works!</h1></body></html> 페이지가 표시됩니다.  6. docker logs web  7. docker rm -f web  8. docker rmi httpd |

9. 컨테이너 관리

|  |
| --- |
| 1. 컨테이너 하드웨어 리소스 제한 어떻게 해요?  기본으로 컨테이너는 호스트 하드웨어 리소스의 사용 제한을 받지 않는다.            2. 컨테이너 사용 리소스를 확인하는 모니터링 툴이 있나요? |
| 메모리 리소스 제한 실습  $mkdir build  $cd build  $vim Dockerfile  FROM debian  MAINTAINER Hyeonse LEE <dlgustp1487@naver.com>  RUN apt-get update; apt-get install stress -y  CMD ["/bin/sh", "-c", "stress -c 2"]  $docker build -t stress .  $docker run -m 100m --memory-swap 100m stress:latest stress --vm 1 --vm-bytes 90m -t 5s  $docker run -m 100m --memory-swap 100m stress:latest stress --vm 1 --vm-bytes 150m -t 5s  $docker run -m 100m stress:latest stress --vm 1 --vm-bytes 150m -t 5s  docker rm containerID로 3개 지우기  $docker run -d -m 100M --name m4 --oom-kill-disable=true nginx  $docker inspect m4  $cat /sys/fs/cgroup/memory/docker/18c62f2df79795d8367857f3773010a62f40a671ab197dd1d92faa6dccba0c6e/memory.oom\_control  CPU 리소스 제한  $lscpu  $docker run --cpuset-cpus 1 --name c1 -d stress:latest stress --cpu 1  실행중인 컨테이너 삭제  $docker run --cpuset-cpus 0 --name c1 -d stress:latest stress --cpu 1  $docker rm -f c1  $docker run --cpuset-cpus 0-1 --name c2 -d stress stress --cpu 1  $htop  $docker rm -f c2  $docker run -c 2048 --name cload1 -d stress:latest  $docker run --name cload2 -d stress:latest  $docker run -c 512 --name cload3 -d stress:latest  $docker run -c 512 --name cload4 -d stress:latest  $docker ps -a  $docker stats cload1  $docker stats  Block I/O 제한  $lsblk  $docker run -it --rm --device-write-iops /dev/sda:10 ubuntu:latest /bin/bash  $dd if=/dev/zero of=file1 bs=1M count=10 oflag=direct  $dd if=/dev/zero of=file1 bs=1M count=10 oflag=direct  $exit  $docker run -it --rm --device-write-iops /dev/sda:100 ubuntu:latest /bin/bash  $dd if=/dev/zero of=file1 bs=1M count=10 oflag=direct  $dd if=/dev/zero of=file1 bs=1M count=10 oflag=direct  $exit  cAdvisor 실행  $VERSION=v0.36.0 # use the latest release version from https://github.com/google/cadvisor/releases  sudo docker run \  --volume=/:/rootfs:ro \  --volume=/var/run:/var/run:ro \  --volume=/sys:/sys:ro \  --volume=/var/lib/docker/:/var/lib/docker:ro \  --volume=/dev/disk/:/dev/disk:ro \  --publish=8080:8080 \  --detach=true \  --name=cadvisor \  --privileged \  --device=/dev/kmsg \  gcr.io/cadvisor/cadvisor:$VERSION  gui환경에서 파이어폭스 http://localhost:8080    1.  $docker run -d -m 200M --memory-swap 300M --cpuset-cpus 1 --name db -e MYSQL\_ROOT\_PASSWORD=pass -p 3306:3306 mysql:latest  2.  $docker stats db |

10. 컨테이너가 사용하는 스토리지

|  |
| --- |
| 1. 컨테이너 볼륨이 뭐에요?  컨테이너 이미지는 readonly  컨테이너에 추가되는 데이터들은 별도의 RW 레이어에 저장    2. 데이터를 보존하고 싶어요      3. 컨테이너끼리 데이터 공유 가능한가요? |
| 실습  1. mysql db data 영구 보존하기    $docker run -d --name db -v /dbdata:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=pass mysql:latest  $docker exec -it db /bin/bash  $mysql -u root -ppass  $show databases;  $create database ttabae;  $exit  $exit  $cd /dbdata/  $docker rm -f 유  $ls /dbdata/  $ docker run -d --name db -v /var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=pass mysql:latest  $docker inspect 유  $cd /var/lib/docker/volumes/6299c4769c97ee497b35fb9ea0a6bd0ea9ff659cfb44698338a34186cb637a17/\_data  $ls  $docker rm -f db  $docker volume ls  $docker volume rm 89ff88cba98da5df4ac357ec169720d71ad346351f5194497028623a92fb538b  2. 웹데이터 readonly 서비스로 컨테이너에 지원하기    $mkdir /webdata  $cd /webdata/  $echo "<h1>YouTUBE TTABAE-LEARN</h1>" > index.html  $cat index.html  $docker run -d --name web -p 80:80 -v /webdata:/usr/share/nginx/html:ro nginx:1.14  $docker ps  3. 컨테이너간 데이터 공유하기    $docker rm -f web  $df -h /  df 명령어 디스크 사용량을 모니터링 할 때 사용  $mkdir lab8  $cd lab8  $cat df.sh  #! /bin/bash  mkdir -p /webdata  while true  do  df -h / > /webdata/index.html  sleep 10  done  $cat > dockerfile  FROM ubuntu:18.04  ADD df.sh /bin/df.sh  RUN chmod +x /bin/df.sh  ENTRYPOINT ["/bin/df.sh"]  $docker build -t lee1487/df:latest .  $docker images  $docker run -d -v /webdata:/webdata --name df lee1487/df:latest  $docker run -d --name web -v /webdata:/usr/share/nginx/html:ro -p 80:80 nginx:1.14 |

11. 컨테이너간 통신(네트워크)

|  |
| --- |
| 1. 컨테이너는 어떻게 통신하나요?    2. 컨테이너 포트를 외부로 노출할 수 있어요?    3. 컨테이너 네트워크를 추가할 수 있나요?    4. 컨테이너끼리 통신은 어떻게 하나요? |