

목차

docker image save/load export/import	3
<i>docker save (docker image -> tar)</i>	3
<i>docker load (tar -> docker image)</i>	3
<i>docker export (docker container -> tar)</i>	3
<i>docker import (tar -> docker image)</i>	3
<i>export / import 와 save / load의 차이</i>	4
참조링크	4

docker image save/load export/import

docker build나 commit으로 만들어진 이미지는 일반적으로 docker hub와 같은 registry에 push되고, 이를 다시 pull 받는 방식으로 사용된다.

하지만, 간혹 docker 이미지를 registry를 거치지 않고 이동해야 할 때가 있다.

이럴 때 사용자는 docker 이미지 혹은 컨테이너를 tar파일로 만들 수 있다.

docker save (docker image -> tar)

docker 이미지를 tar파일로 저장하기 위해서는 docker save 커맨드를 사용한다.

```
$ docker save [옵션] <파일명> [이미지명]
```

저장할 파일명을 지정하는 옵션은 -o 를 사용한다.

```
$ docker save -o nginx.tar nginx:latest
```

docker load (tar -> docker image)

tar파일로 만들어진 이미지를 다시 docker image로 되돌리기 위해서는 docker load 커맨드를 사용한다.

```
$ docker load -i [tar파일명]
```

docker export (docker container -> tar)

docker는 이미지 뿐 아니라 container를 tar파일로 저장하는 명령어를 제공한다.

```
$ docker export [컨테이너명 or 컨테이너ID] > xxx.tar
```

docker import (tar -> docker image)

export 커맨드를 통해 만들어진 tar 파일을 다시 docker image로 생성하는 명령어이다.

```
$ docker import [파일 or URL] - [image name[:tag name]]
```

root 권한으로 실행하지 않을 경우, 액세스 권한이 없는 파일들이 포함되지 않는 문제가 발생할 수 있다.

export / import 와 save / load의 차이

docker export의 경우 컨테이너를 동작하는데 필요한 모든 파일이 압축된다. 즉, tar파일에 컨테이너의 루트 파일시스템 전체가 들어있는 것이다. 반면에 docker save는 레이어 구조까지 포함한 형태로 압축이 된다.

즉, 기반이 되는 이미지가 같더라도 export와 save는 압축되는 파일 구조와 디렉터리가 다르다.

결론은 export를 통해 생성한 tar 파일은 import로, save로 생성한 파일은 load로 이미지화 해야 한다.

참조링크

- <https://www.leafcats.com/240>

From:
<https://atl.kr/dokuwiki/> - AllThatLinux!

Permanent link:
https://atl.kr/dokuwiki/doku.php/docker_image_save_load_export_import

Last update: **2021/06/30 01:00**

