

제한된 컴퓨터 환경을 위한 경량 컨테이너 런타임 엔진: Docker와의 비교 연구

김창인, 허유정, 장수영, 최창범

국립한밭대학교 컴퓨터공학과

E-mail: {[20217140, 20211939](mailto:20217140@edu.hanbat.ac.kr)}@edu.hanbat.ac.kr, {[syjang, cbchoi](mailto:syjang@hanbat.ac.kr)}@hanbat.ac.kr

요 약

컨테이너 기술은 현대 소프트웨어 개발과 배포에서 중요한 역할을 담당하고 있다. 그러나 Docker는 리소스 사용량이 많아 제한된 환경에서는 최적의 선택이 아닐 수 있다. 본 연구는 제한된 컴퓨터 환경에서 효율적으로 동작할 수 있는 경량화된 컨테이너 런타임의 개발과 성능 분석을 목표로 한다. 경량화된 컨테이너 런타임 엔진을 개발하여 Docker와 비교 실험을 수행하였다. 실험 결과, 경량화된 런타임은 Docker에 비해 컨테이너 시작 시간과 메모리 사용량에서 우수한 성능을 보였다. 이는 제한된 리소스 환경에서 경량화된 런타임이 더 효율적으로 동작할 수 있음을 시사한다.

I. 서 론

컨테이너 기술은 현대 소프트웨어 개발과 배포에서 중요한 역할을 담당하고 있다. Docker는 현재 가장 널리 사용되는 컨테이너화 도구로, 애플리케이션을 격리된 환경에서 실행할 수 있도록 지원한다. 그러나 Docker의 리소스 사용량과 성능 문제는 일부 환경, 특히 리소스가 제한된 소형 컴퓨터(예: 라즈베리 파이)에서 최적화되지 않은 경우가 많다[1] [2]. 소형 컴퓨터와 임베디드 시스템은 IoT(사물 인터넷) 및 엣지 컴퓨팅과 같은 다양한 분야에서 중요한 역할을 하고 있다. 이러한 환경에서는 제한된 리소스를 효율적으로 사용하기 위해 경량화된 컨테이너 런타임이 필요하다. 기존의 Docker는 기능이 풍부하지만, 리소스 사용량이 높아 제한된 환경에서는 최적의 선택이 아닐 수 있다. 따라서 경량화된 컨테이너 런타임을 개발하여 이러한 문제를 해결할 필요가 있다.

본 연구의 목적은 경량화된 컨테이너 런타임을 개발하고, 이를 통해 Docker와의 성능을 비교 분석하는 것이다. 본 연구에서는 실행 시간, 메모리 사용량, CPU 사용량 등의 성능 지표를 비교하여 경량화된 컨테이너 런타임의 효율성을 입증하고자 한다. 특히, 소형 컴퓨터 환경에서도 효율적으로 동작할 수 있는 경량화된 솔루션을 제안하는 것을 목표로 한다.

II. 본 론

1. 경량화된 컨테이너 런타임 엔진 설계

경량화된 컨테이너 런타임의 설계 목표는 메모리 및 CPU 사용량을 최소화하여 제한된 리소스를 효율적으로 사용하고, 컨테이너 시작 시간 및 실행 성능을 최적화하는 데 있다.

본 연구에서 제안하는 경량화된 컨테이너 런타임 엔진의 주요 구성 요소는 다음과 같다. 첫째, 프로세스 격리를 위한 Namespace 설정과 리소스 제한을 위한 cgroup 설정을 통해 격리된 실행 환경을 제공한다. 이는 각 컨테이너가 독립적으로 동작하도록 하여 자원 경쟁을 최소화한다. 둘째, 지정된 이미지를 사용하여 컨테이너를 실행하고 명령을 수행하는 기능을 포함한다. 이를 통해 컨테이너가 독립적으로 동작하며 필요한 리소스를 효율적으로 사용할 수 있도록 한다[3].

아래 그림1은 경량화된 컨테이너 런타임 엔진의 설계를 나타낸다.

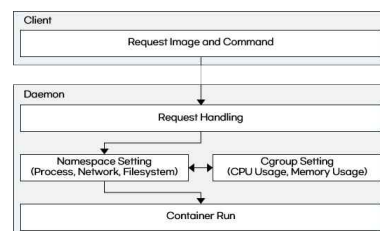


그림 1. 경량화된 런타임 엔진 설계도

2. 실험 방법

경량화된 컨테이너 런타임의 성능을 평가하기 위해 Docker와 비교 실험을 수행하였다. 실험 환경은 아래 표 1과 같다.

표 1. 실험 환경

Item	Details
CPU	12th Gen Intel(R) Core(TM) i9-12900K 3.19 GHz
Memory	64.0GB
OS	Ubuntu 20.04 LTS
Kernel	5.15.0-113-generic
Docker	26.0.0,
Test Image	Nginx

실험은 세 단계로 진행되었다. 첫째, 컨테이너 시작 시간을 측정하였다. Docker와 경량화된 런타임에서 Nginx 컨테이너를 실행하고 시작 시간을 측정하여 비교하였다. 둘째, 컨테이너 실행 중 메모리 사용량을 docker stats와 top 명령어를 통해 측정하였다. 이 측정은 컨테이너가 정상적으로 실행된 후 일정 시간 동안의 평균 값을 기반으로 하였다. 셋째, ApacheBench 도구를 사용하여 Nginx 서버에 부하를 가한 후 CPU 사용량을 측정하였다.

3. 실험 결과

실험 결과는 다음과 같다. 먼저, 컨테이너 시작 시간 측정에서 경량화된 런타임이 Docker보다 약 8배 빠른 시작 시간을 보였다. 메모리 사용량 측정에서는 경량화된 런타임이 Docker보다 약 60% 적은 메모리를 사용하였다. CPU 사용량 측정에서는 경량화된 런타임이 Docker에 비해 상대적으로 높은 CPU 사용량을 보였으나, 이는 경량화된 런타임의 최적화가 아직 덜 되었기 때문일 수 있다.

표 2. 성능 비교 결과

Item		Docker	Lightweight Runtime
Container Start Time			
- real	(s)	0.273	0.031
- user	(ms)	17	9
- sys	(ms)	13	4
Memory Usage	(MiB)	18.2	7.28
CPU Usage	(%)	14	20

실험 결과, 경량화된 런타임 엔진이 Docker보다 빠른 컨테이너 시작 시간과 적은 메모리 사용량을 보였다. 이는 경량화된 런타임이 소형 컴퓨터 환경에서 더 효율적으로 동작할 수 있음을 시사한다.

III. 결 론

본 연구에서는 제한된 컴퓨터 환경에서 효율적으로 동작할 수 있는 경량화된 컨테이너 런타임 엔진을 개발하고, 이를 Docker와 비교하여 성능을 분석하였다. 실험 결과, 경량화된 런타임은 컨테이너 시작 시간과 메모리 사용량 측면에서 Docker보다 우수한 성능을 보였다. 이는 제한된 리소스 환경에서의 활용 가능성을 높여준다. 다만, CPU 사용량 측면에서는 경량화된 런타임이 상대적으로 높은 값을 보여, 추가적인 최적화가 필요함을 시사한다. 향후 연구에서는 CPU 사용량을 최적화하고, 다양한 애플리케이션을 대상으로 성능을 검증하는 방향으로 진행할 예정이다. 이를 통해 실제 환경에서의 적용 가능성을 높일 수 있을 것이다.

감사의 글

“본 연구는 2024년 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 연구결과로 수행되었음” (2022-0-01068)

References

[1] S. Sultan, I. Ahmad and T. Dimitriou, "Container Security: Issues, Challenges, and the Road Ahead," in *IEEE Access*, vol. 7, pp. 52976-52996, 2019

[2] Kozhirbayev, Z., Sinnott, R.O., "A performance comparison of container-based technologies for the cloud," *Future Generation. Computer. Systems*, vol. 68, pp. 175 - 182, 2017.

[3] Espe, Lennart, et al. "Performance Evaluation of Container Runtimes." *CLOSER*, pp. 273-281, 2020.

[4] Docker Documentation: Docker, <https://docs.docker.com>, Accessed: Jul. 9, 2024.

[5] Linux Namespaces: Linux, <https://man7.org/linux/man-pages/man7/namespaces.7.html>, Accessed: Jul. 9, 2024.

[6] cgroups : <https://github.com/containerd/cgroups.git>