

# 쿠버네티스 기반 클라우드 시스템에서의 백업 및 복원

강문준, 김영한\*  
승실대학교

mjkang10@dcn.ssu.ac.kr, \*younghak@ssu.ac.kr

## Backup and Restore in Kubernetes based Cloud System

Kang Moon Jun, Kim Young Han\*  
Soongsil Univ.

### 요 약

쿠버네티스를 오케스트레이션 플랫폼으로 사용하는 클라우드 시스템에서의 백업 및 복원은 쿠버네티스 클러스터 내의 모든 노드, 파드와 다른 모든 오브젝트 리소스들의 백업과정을 포함한다. 본 논문에서는 쿠버네티스 환경에서의 백업 및 복원에 대한 현 접근방식과 관련 이슈들에 대해 분석하고 이에 기반하여 추후 연구 방향을 제시한다.

### I. 서 론

쿠버네티스 기반 클라우드 시스템에서의 백업은 쿠버네티스 클러스터에서 동작하는 파드, 컨트롤 플레인, 볼륨 등 모든 구성요소들뿐 아니라 컨테이너화된 서비스들을 모두 백업하는 작업을 뜻한다[1]. 최근에는 쿠버네티스 인프라에서 마이크로 서비스 구조로 동작하는 애플리케이션이 증가함에 따라, 각각의 서비스 구성요소를 백업하는 작업이 필수적이다. 이러한 마이크로 서비스 구조로 구성되는 컨테이너화된 애플리케이션은 여러 노드에 걸쳐 여러 파드 내에서 동작하기 때문에 가상 머신 혹은 물리적 머신 등 단일 머신의 데이터에 초점을 맞춘 기존의 백업 방식과는 차별되는 방안이 필요하다[2]. 이에, 쿠버네티스 백업은 컨테이너가 실행되며 발생하는 모든 데이터들과 볼륨, 그리고 etcd 에 저장되는 리소스들의 정보 및 yaml 매니페스트 파일 등 모든 데이터 및 애플리케이션 구성 정보를 수집하여 다른 객체 스토리지 위치로 백업하는데 초점을 둔다. 그리고 백업 데이터에 대한 보안성을 높이고 재해 복구, 서비스 마이그레이션 등의 경우와 같이 백업한 쿠버네티스 리소스들을 새로운 쿠버네티스 클러스터에 복구하기 위한 작업의 필요성 또한 증가하는 추세이다. 이와 같이 쿠버네티스 백업 및 복구 시스템의 필요성과 함께 기존 방식과는 차별화되는 쿠버네티스 환경에서의 새로운 백업 방식의 필요성이 제기됨에 따라 이와 관련된 연구와 개발이 활발히 진행되고 있다. 이에, 본 논문에서는 쿠버네티스 백업 및 복원에 대한 현재의 접근방식들을 세분화하여 분석한다. 또한, 현재 남겨진 관련 이슈들을 분석하고, 이에 기반하여 가능한 추후 연구 방향을 제시한다.

### II. 본 론

본론에서는 쿠버네티스 백업 및 복원에 대한 현재의 접근방식에 대해 세분화하여 분석한 내용을 서술한다.

#### 1. 백업 데이터 유형

백업 데이터의 유형은 쿠버네티스 오브젝트의 정보를 나타내는 configuration data 와 컨테이너가 실행되며 발생하는 persistent data 로 나누어지며, 이러한 백업 데이터들은 외부 스토리지에 저장된다.

- 1) Configuration data : 모든 쿠버네티스 리소스의 정보는 etcd 데이터베이스에 저장되기 때문에, etcd 데이터베이스 내부의 데이터 백업 과정이 필수적이다. 또한, 개인 레포지토리를 사용하는 경우 컨테이너 이미지와 컨테이너 이미지를 만들기 위한 도커 파일 및 yaml 매니페스트 파일도 함께 백업될 필요가 있다.
- 2) Persistent data : 쿠버네티스 상에서 컨테이너가 실행되며 발생하는 대부분의 데이터들은 별도의 데이터베이스와 PersistentVolume 을 이용하여 저장된다. 따라서, 클러스터에서 사용되는 데이터베이스나 PersistentVolume 을 volumesnapshot 기능을 이용해 백업하여 내부에 저장된 persistent data 에 대한 백업 데이터를 생성한다.

#### 2. 백업 방식

대부분의 경우에서 persistent data 백업에 대한 접근방식은 동일하지만 configuration data 의 백업 데이터를 생성하는 방식은 크게 두 가지로 구분된다. 첫 번째는 etcd 데이터베이스에 직접 접근하여 백업 데이터를 생성하는 방식으로 etcd 데이터베이스 전체를 백업하거나 etcd cli 를 통해 접근하여 내부 데이터를 추출하는 방식이다. 하지만, 이러한 방식은 etcd 데이터베이스에 대한 접근권한이 주어지지 않는 AWS 의 EKS 와 같은 managed Kubernetes Cluster 환경에서는 사용이 어렵다는 점과 여러 etcd 서버를 보유한 클러스터의 경우에도 사용에 문제가 있다. 이에 이러한 문제점들을 극복하고자 etcd

데이터베이스에 직접 접근하지 않고 Kubernetes API 서버를 통해 etcd 내의 특정 리소스들에 대한 state 정보를 전송받아 백업 데이터를 생성하는 방식이 제안되었다. 아래 그림은 해당 방식을 나타낸 예시로, 오픈소스 프로젝트 velero의 동작을 나타낸다[3],[4].

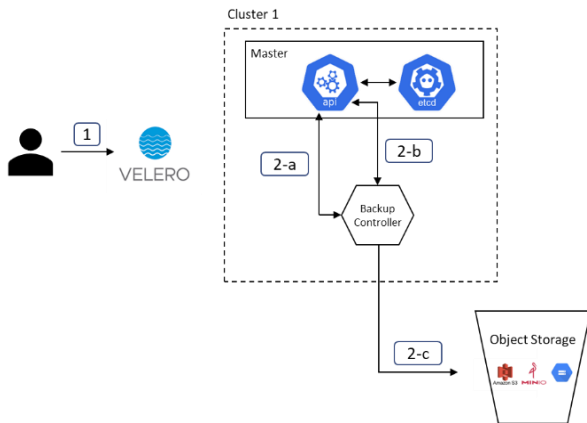


그림 1. Backup method of Velero

### 3. 백업 데이터 생성 시기

백업 데이터 생성 시기는 수동 혹은 스케줄링 방식으로 결정된다. 스케줄링 방식은 cron 표현식으로 정의되며 특정 주기마다 백업 작업이 수행되는 방식이다. 이를 통해 경우에 따라 자동으로 백업 데이터를 생성할 수 있으며, 의도치 않은 cluster failure가 발생하는 경우를 대비하기 위해서 스케줄링 방식을 이용한 주기적인 백업 데이터 생성이 요구된다. 하지만, 주기적으로 생성되는 백업 데이터의 용량에 대한 관리나 복원 시 사용되는 백업 데이터 등에 대한 문제는 여전히 백업 시스템에서의 주요 고려사항이다[5].

## III. 관련 이슈

### 1. 예측 불가능성

백업 데이터의 생성 시기는 앞서 언급한 바와 같이 수동 혹은 스케줄링 방식으로 정해진다. 하지만 이러한 방식은 컴퓨팅 리소스 및 스토리지 용량에 제한이 있으므로 백업 데이터의 관리 측면에서 비효율적이다.[6] 이에 따라 동작하는 컨테이너의 수가 예측하기 어려운 수준으로 자유롭게 스케일링되는 쿠버네티스 환경에 대응하고 백업 데이터를 효율적으로 관리할 수 있는 추가적인 방안이 요구된다.

### 2. 클러스터 복원 과정의 자동화

생성한 백업 데이터가 복원될 대상 클러스터를 지정하고 저장되어 있는 백업 데이터 중 어떤 리소스들을 복원할 것인지 그리고 복원 시기의 결정 등 쿠버네티스 클러스터의 전반적인 복원 과정에 대한 자동화가 필요하다. 자동화 스크립트를 이용하는 방법도 있지만, 지속적으로 변화하는 쿠버네티스 클러스터와 다양한 경우의 복원 시나리오를 고려한 포괄적인 대처 방안이 필요하다.

## IV. 추후 연구 방향

### 1. 예측 기반 백업 과정 자동화

쿠버네티스 클러스터에 대한 failure detection 등의 기능은 현재에도 상용화되어 있다. 이와 같이 cluster failure에 대한 원인을 분석하고 분석 결과에 따른 인공지능 기반 예측 시스템을 이용하여 백업 데이터 생성 및 관리를 현재의 수동 혹은 스케줄링 방식보다 효율적으로 수행하는 쿠버네티스 클러스터 백업 시스템을 구축할 수 있다.

### 2. Dead capture 방식의 백업

현재의 쿠버네티스 백업은 쿠버네티스 클러스터 내의 애플리케이션과 애플리케이션의 동작에 따라 생성되는 state 정보가 저장되는 볼륨도 함께 백업을 수행하는 방식으로 stateful 애플리케이션에 대한 백업 및 복원을 구현한다. 하지만, 비디오 스트리밍 애플리케이션과 같이 애플리케이션의 state 정보 뿐만 아니라 애플리케이션의 실행 상태에 대한 정보까지도 백업하여 의도치 않은 cluster failure 발생 시 혹은 서비스 대상의 위치가 지속적으로 변경되는 경우에 더 높은 Quality of Service(QoS)를 제공할 수 있는 방안에 대한 연구도 진행될 필요가 있다.

## V. 결론

최근 재해 복구 및 서비스 마이그레이션 등의 경우를 위한 쿠버네티스 백업 및 복원에 대한 관심이 증가하면서 이에 초점을 맞춘 관련 연구와 개발이 함께 이루어지고 있다. 본 논문에서는 쿠버네티스 백업 및 복원에 대한 현재의 접근방식을 백업 데이터 및 방식, 그리고 백업 시기로 세분화하여 여러 관점에서 분석하고 남아있는 관련 이슈들에 대해 정리한다. 또한, 이러한 분석 결과를 기반으로 추후 진행 가능한 연구 방향을 제안한다.

## ACKNOWLEDGMENT

이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2020-0-00946, 하이브리드 클라우드 환경에서의 고속, 자동 서비스 복구 및 이전 소프트웨어 개발)

## 참고 문헌

- [1] What is Kubernetes Backup?  
<https://portworx.com/kubernetes-backup/>
- [2] <https://metallic.io/knowledge-center/glossary/kubernetes-backup>
- [3] <https://velero.io/docs/v1.10/how-velero-works/>
- [4] Implementing cluster backup solution to build resilient cloud architecture, Dang Minh Bui
- [5] Disaster Recovery Analysis of different Cloud Managed Kubernetes Clusters, Sergio Fernández Rubi
- [6] Five Challenges of Kubernetes Backup,  
<https://dcig.com/2021/07/five-challenges-kubernetes-backup/>