

# 하이브리드 클라우드 환경에서 원격 서버리스 함수 호출

김민진, 김영한\*  
 숭실대학교

kmj0212@dcn.ssu.ac.kr, \*younghak@ssu.ac.kr

## Remote serverless function call in hybrid cloud environments

Kim Min Jin, Kim Young Han\*  
 Soongil Univ.

### 요 약

사설 클라우드와 공공 클라우드를 동시에 이용하는 환경에서 비용절감을 위해서는 반드시 필요한 워크로드만을 공공 클라우드 환경에서 운용하도록 하는 것이 필요하다. 이를 위해서 공공 클라우드의 서비스는 FaaS(Function as a Service) 형태의 Serverless 컴퓨팅으로 운영하는 것이 적절하다. 본 논문에서는 원격지의 공공 클라우드 상의 FaaS를 사설 클라우드 환경에서 호출하는 방법에 대한 것으로 기존의 단순 FaaS의 API(Application Programming Interface)를 액세스 하는 대신 사설 클라우드 환경에서의 이벤트를 공공 클라우드의 FaaS와 연동하는 방법을 설계하였다. 이를 통하여 보다 다양한 형태로 공공 클라우드 내의 FaaS를 이용할 수 있을 것이다.

### I. 서론

하이브리드 클라우드는 다양한 환경의 조합으로 애플리케이션이 실행되는 클라우드 환경이다[1]. 공공 클라우드만 사용하기에는 비용이 많이 들게 되므로 최근에는 사설 클라우드와 같이 운용되고 있다. 이처럼 하이브리드 클라우드는 공공 클라우드와 사설 클라우드의 이점을 제공하며 데이터 센터의 기존 아키텍처를 활용한다. 그러나 기존 하이브리드 클라우드에서는 원격지의 공공 클라우드 상의 FaaS를 사설 클라우드 환경에서 API를 액세스하여 호출하는 방식이다. 예를 들어, 사설 클라우드 환경에서 AWS Lambda로부터 FaaS를 호출하는 경우가 있다. Lambda는 서버를 프로비저닝하거나 관리하지 않고도 코드를 실행할 수 있게 해주는 컴퓨팅 서비스이다[2]. 사설 클라우드 환경에서 AWS Lambda를 호출하기 위해서는 Lambda를 VPC(Virtual Private Cloud)에 포함시키고 사설 클라우드 환경 네트워크에서 AWS VPC로 연결되는 Site-to-Site VPN을 생성해야 한다. Site-to-Site VPN이란 IPSec 암호화 프로토콜을 사용해 AWS 클라우드 환경과 사설 클라우드 환경을 연결해주는 서비스이다. Site-to-site VPN 연결이 생성되면 AWS 가상 프라이빗 게이트웨이와 사설 클라우드의 게이트웨이 사이에 두개의 VPN 터널이 생성된다. 두 개의 터널은 각각 다른 가용 영역에 터널 엔드 포인트를 가지게 된다. 이 터널을 이용하여 사설 클라우드 환경에서 공공 클라우드 환경으로 Lambda를 이용할 수 있다. 이렇게 사설 클라우드 환경에서는 커스텀 게이트웨이 디바이스 또는 애플리케이션을 구성해야 하며, 공공 클라우드 VPC에 존재하는 Lambda를 호출하기 위해 [그림 1]과 같이 직접 API에 명령을 내려야 한다. 다른 공공 클라우드 환경의 FaaS를 호출하는 방법도 이와 비슷하다. 따라서 각각의 공공 클라우드에 맞게 인프라를 구성해야 하고 운영 비용이 많이 든다는 단점이 있다.

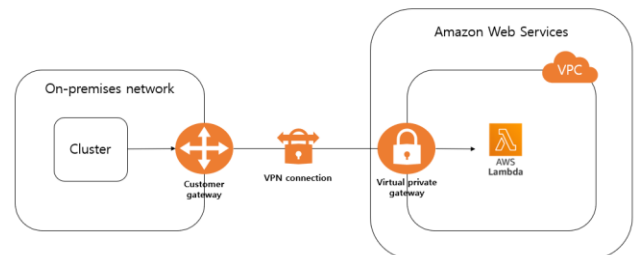


그림 1. AWS Site-to-Site VPN

이에 본 논문에서는 사설 클라우드 환경에서의 CloudEvents를 통해 다양한 공공 클라우드의 FaaS와 연동하는 방법을 설계하고자 한다. 이벤트는 이벤트 생성자마다 형식이 다양하므로 서비스, 플랫폼 및 시스템 간에 상호 운용성을 제공하기 위해 이벤트 구조 및 메타데이터를 표준화하는 Cloudevents를 사용하였다.

### II. 제안 구조

본 논문에서 제안하는 구조는 아래 [그림 2]과 같다.

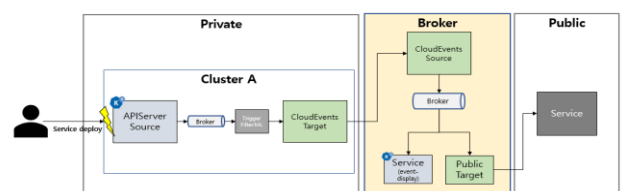


그림 2. 하이브리드 클라우드에서 원격 이벤트 기반 서버리스 함수 제안 구조

인프라 구조는 사설 클라우드 환경과 공공 클라우드 환경을 조합한 하이브리드 클라우드 환경으로 이루어져 있다. 그리고 두 환경의 연결을 도와주는 브로커가 있으며 클러스터로 구성 되어있다.

사설 클라우드 환경에 존재하는 클러스터 A는 리소스가 큰 서비스, 예를 들면 ML 과 같은 서비스를 배포하지 못하도록 트리거 컴포넌트에 미리 정의한다. 따라서 해당 서비스가 사설 클러스터 내부로 들어오게 되면 pending 상태로 존재하게 된다. 공공 클라우드에는 사설 클라우드에서 돌아가지 못하는 서비스가 FaaS 형태의 서버리스 컴퓨팅 서비스로 존재한다. 따라서 사용자가 사설 클라우드 환경에 있는 클러스터 A 로 ML 서비스를 배포하게 되면 클러스터 A 에서는 API 서버에 명령이 전달되는 순간을 이벤트 소스로 받게 되어 API 서버 소스 컴포넌트가 트리거 되게 된다. 이 컴포넌트는 쿠버네티스에 존재하는 API 서버로 파드 생성 명령이 들어오면 동작하도록 쿠버네티스 Role 오브젝트를 통해 구성한다. 따라서 소스 컴포넌트가 트리거 되어 서비스 정보 이벤트는 트리거 컴포넌트의 필터를 거쳐 클라우드 이벤트 타겟으로 보내진다. 여기서 클라우드 이벤트 타겟은 클러스터 간의 연결을 위해 존재하는 컴포넌트이며 이벤트 오픈소스 플랫폼에서 제공한다. 이 컴포넌트는 [그림 3]와 같이 클라우드 이벤트 소스 컴포넌트의 HTTP 엔드포인트를 통해 연결된다. 이벤트 소스 컴포넌트는 클라우드 이벤트 타겟 컴포넌트로부터 이벤트 정보를 받아오는 컴포넌트이며 이벤트 오픈소스 플랫폼에서 제공하고 있는 컴포넌트이다.

```
apiVersion: targets.triggermesh.io/v1alpha1
kind: CloudEventsTarget
metadata:
  name: gateway-out
  namespace: triggermesh-cluster
spec:
  endpoint: http://cloudeventssource-gateway-in.triggermesh-cluster.192.168.40.115.sslip.id
```

그림 3. 클라우드 이벤트 타겟 구성

이로 인해 사설 클라우드에서 ML 서비스를 FaaS 형태로 실행하게 된다. 여기서 두 환경은 공공 클라우드 환경과 사설 클라우드 환경으로 서로 다른 클라우드이기 때문에 중간에 서비스를 전달해주는 인프라가 있어야 한다. 따라서 클러스터로 구성된 브로커라는 인프라를 구성하여 사설 클라우드 환경에 존재하는 클라우드 이벤트 타겟 컴포넌트에서 전송된 서비스의 정보를 클라우드 이벤트 소스 컴포넌트를 통해 가져오고, 퍼블릭 타겟을 통해 공공 클라우드로 서비스 정보 이벤트를 보낸다. 브로커 클러스터 내에 존재하는 트리거 컴포넌트는 컴포넌트를 배포하기 전에 미리 필터를 정의하여 각각의 역할에 맞게 필터링한다. 따라서 이벤트가 제대로 전송이 되었는지 확인하는 이벤트 모니터링 서비스와 공공 클라우드로 서비스 정보를 전송하는 퍼블릭 타겟으로 구성하였다. 이러한 서비스는 이벤트가 왔을 때만 동작하는 서버리스 형태로 생성된다. 이와 같은 구조의 장점은 다음과 같다. 기존 인프라 구성에서는 공공 클라우드 환경으로 서비스를 호출해야 할 때 직접 API 액세스 명령을 내려야 했다. 하지만 본 제안구조는 CloudEvents를 통해 사설 클라우드 환경에서 특정 서비스가 왔다는 것을 인지하여 서비스 정보만 전달하면 된다. 또한, 기존에 각각의 공공 클라우드의 맞게 인프라를 꾸며야 하였다. 그러나 본 제안구조에서는 브로커

클러스터를 두어 다양한 공공 클라우드로부터 FaaS 를 호출할 수 있다.

### III. 결론

본 논문에서는 하이브리드 클라우드에서 서비스 운영에 드는 비용을 효율적으로 사용하기 위해 리소스가 많이 드는 특정 서비스에 대해 이벤트를 활용하여 원격으로 공공 클라우드 환경에서 FaaS 를 호출하는 방안을 제시하였다. 또한, 직접 API 액세스를 사용하여 서비스를 배포하는 것이 아닌 사설 클라우드 환경에서 CloudEvents 를 생성하여 공공 클라우드 서버리스 서비스를 이용하였다. 이를 통하여 보다 다양한 형태로 공공 클라우드 내의 FaaS 를 호출할 수 있으며 비용 절감에도 효과적이다. 추후 연구로써, 실제 구현 환경에서 이를 제공하기 위한 구체적인 기능을 설계하고 실험을 수행할 것이다.

### ACKNOWLEDGMENT

이 논문은 2023 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2020- 0-00946, 하이브리드 클라우드 환경에서의 고속, 자동 서비스 복구 및 이전 소프트웨어 개발)

### 참 고 문 헌

- [1] What is a hybrid cloud, <https://cloud.google.com/learn/what-is-hybrid-cloud>, 2022
- [2] What is aws lambda, [https://docs.aws.amazon.com/ko\\_kr/lambda/latest/dg/welcome.html](https://docs.aws.amazon.com/ko_kr/lambda/latest/dg/welcome.html), 2022
- [3] Functions, events, triggers, <https://cloud.google.com/blog/ko/products/serverless/learn-about-cloud-functions-events-and-triggers>, 2021
- [4] What is aws site-to-site VPN, [https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC\\_VPN.html](https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC_VPN.html), 2022
- [5] CloudEvents, <https://cloudevents.io/>, 2022
- [6] Kubernetes, <https://kubernetes.io/>, 2022
- [7] Knative, <https://knative.dev/docs/>, 2022
- [8] TriggerMesh, <https://docs.triggermesh.io/>, 2022