

07.AWS EKS를 활용해 서버 배포하기

1. EC2에서 쿠버네티스를 활용했을 때 발생하는 AWS 예상 비용

✓ EC2

- EC2 인스턴스 (t4g.small) : 시간당 USD 0.0208 (24시간당 약 700원)

(비용이 걱정되시는 분들은 학습이 끝나자마자 인스턴스를 종료하시길 권장드립니다.)

- 데이터 전송 비용 : 1 GB당 0.1368 USD (1GB당 약 200원)

(실습 과정 동안 1GB 이하의 데이터만 전송합니다.)

- Public IPv4 비용 : 시간당 0.005 USD (24시간당 약 200원)

✓ RDS

- RDS 인스턴스 (t4g.micro) : 시간당 USD 0.025 (24시간당 약 800원)

(프리티어일 경우 월 750시간까지 무료)

- 스토리지 비용 : GB-월당 0.131 USD (20GB-24시간당 약 200원)

(프리티어일 경우 20GB까지 무료)

- Public IPv4 비용 : 시간당 0.005 USD (24시간당 약 200원)

✓ ECR

- 스토리지 비용 : GB-월당 USD 0.10 (1GB-24시간당 약 10원)

(실습 과정 동안 1GB 이하의 데이터만 저장합니다.)

- 데이터 전송 비용 : GB-월당 USD 0.126 (1GB-24시간당 약 6원)

(실습 과정 동안 1GB 이하의 데이터만 전송합니다.)

✓ ELB (Classic Load Balancer)

- 사용 비용 : 시간당 0.025 USD (24시간당 약 800원)
- 처리한 데이터 비용 : GB당 0.008 USD (1GB-약 10원) (실습 과정 동안 1GB 이하의 데이터만 처리합니다.)

✓ EKS

- EKS 클러스터 사용 비용 : 시간당 USD 0.10 (24시간당 약 3,500원) (비용이 걱정되시는 분들은 학습이 끝나자마자 EKS 클러스터를 종료하시길 권장드립니다.)

2. AWS EKS를 남들보다 빠르게 익히려면?!

✓ AWS EKS를 남들보다 빠르게 익히려면?!

지금까지의 쿠버네티스 핵심 개념은 다 배웠다. AWS EKS라고 크게 다를 건 없다. 겁먹을 필요 없다. AWS EKS는 단순히 셋팅법만 익히면 나머지는 다 똑같다. EKS를 셋팅하면서 모든 옵션을 다 알 필요가 없다. 딱 필요하고 중요한 부분에 대해서만 알고 있으면 된다. 파레토의 법칙을 잊지 말자.

완벽하게 모든 기능을 익히려는 순간 학습의 효율은 엄청 떨어진다. 우리 내신 시험을 보는 게 아니다. 현업에서 잘 안 쓰는 지엽적인 내용은 뛰어넘을 수도 있어야 한다. 쿠버네티스를 실제 다루는 것에 집중하자.

3. EKS란?

✓ EKS(Elastic Kubernetes Service)란?

EKS란 **AWS**에서 쿠버네티스를 편하게 관리하고 사용할 수 있게 만든 **AWS용 쿠버네티스**이다.

이와 비슷한 예로 MySQL과 같은 DB를 편하게 관리하고 사용할 수 있게 만든 서비스가 RDS이고,

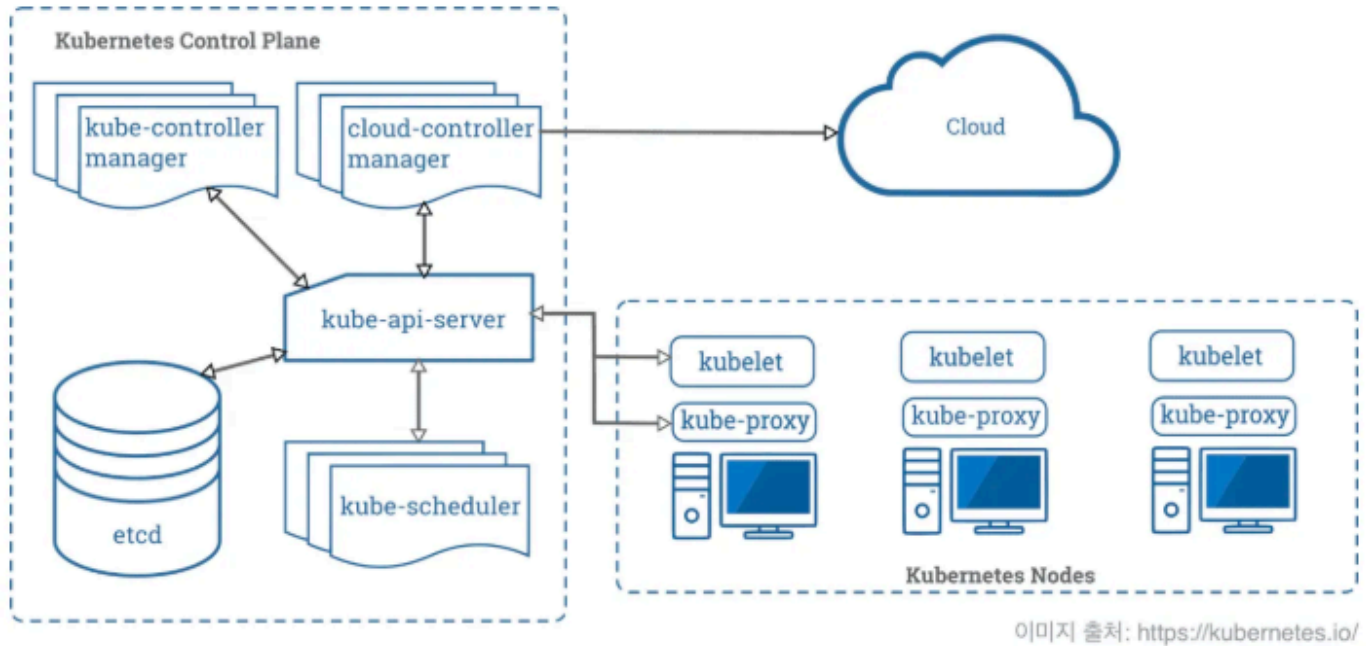
Redis와 같은 캐싱을 편하게 관리하고 사용할 수 있게 만든 서비스가 **ElastiCache**다.

✓ 현업에서도 EKS를 많이 사용할까?

쿠버네티스를 직접 설치해서 관리하는 게 생각보다 손이 많이 간다. 따라서 현업에서는 쿠버네티스를 EC2와 같은 서버에 직접 설치해서 쓰지 않고, AWS에서 제공하는 EKS를 활용하는 경우가 많다.

4. 쿠버네티스와 EKS의 아키텍처 구조

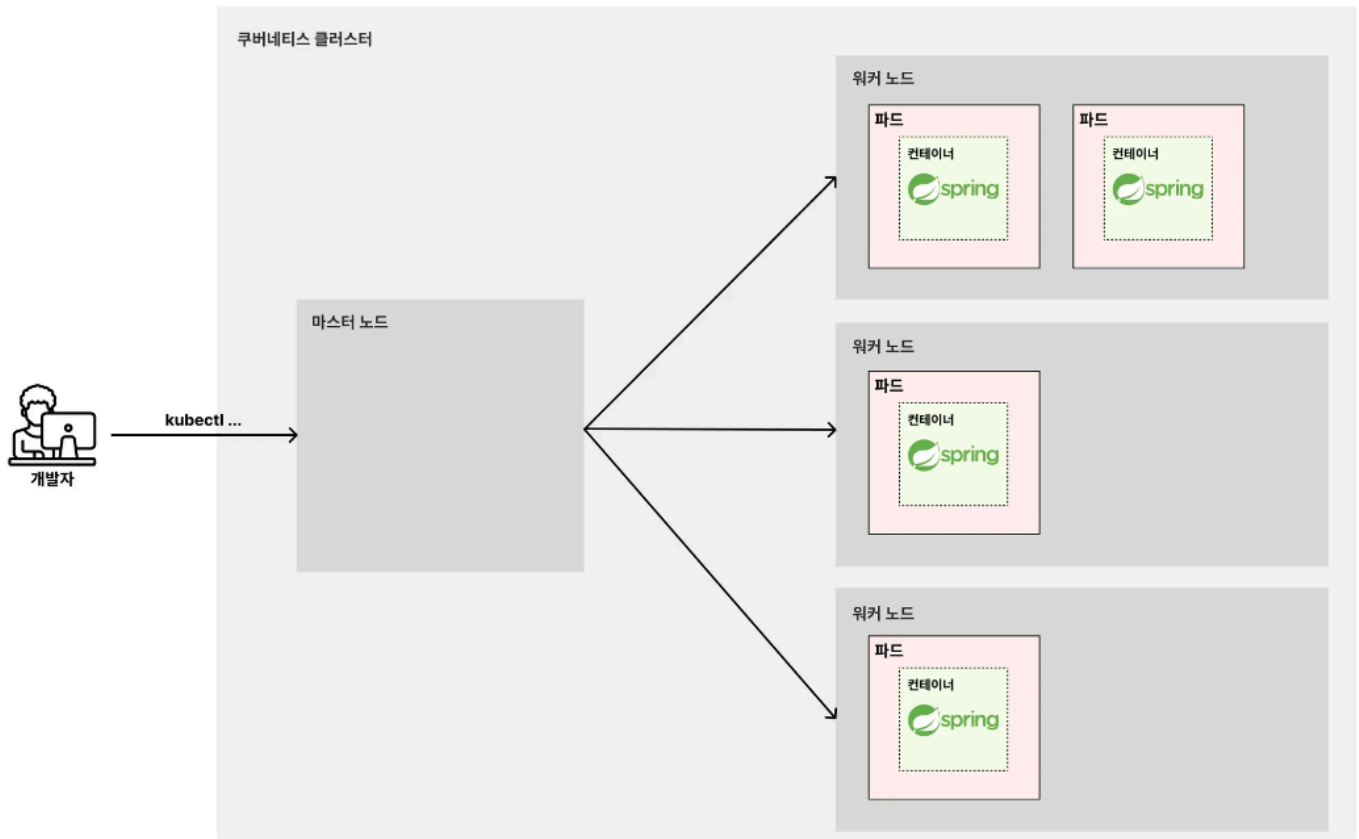
✓ 쿠버네티스의 복잡한 아키텍처 구조



쿠버네티스를 입문하는 입장에서 위와 같은 복잡한 아키텍처 구조를 전부 이해할 필요는 없다. 그런데 대부분의 책과 강의의 초반부를 보면 일일이 다 설명하고 있다. **etcd**가 뭔지, **Control Plane**은 뭔지, **kube-scheduler**가 뭔지 하나하나 다 설명한다. 이렇게 공부하니깐 쿠버네티스가 어렵게 느껴지고, 재미도 없고, 진도도 안 나가고, 이해도 안 되는 것이다. 이런 이유 때문에 이 강의의 초반에 쿠버네티스 아키텍처를 굳이 언급하지 않았다. 지금까지 쿠버네티스의 핵심 개념을 이해하는 데 크게 문제가 없었음을 느꼈을 것이다.

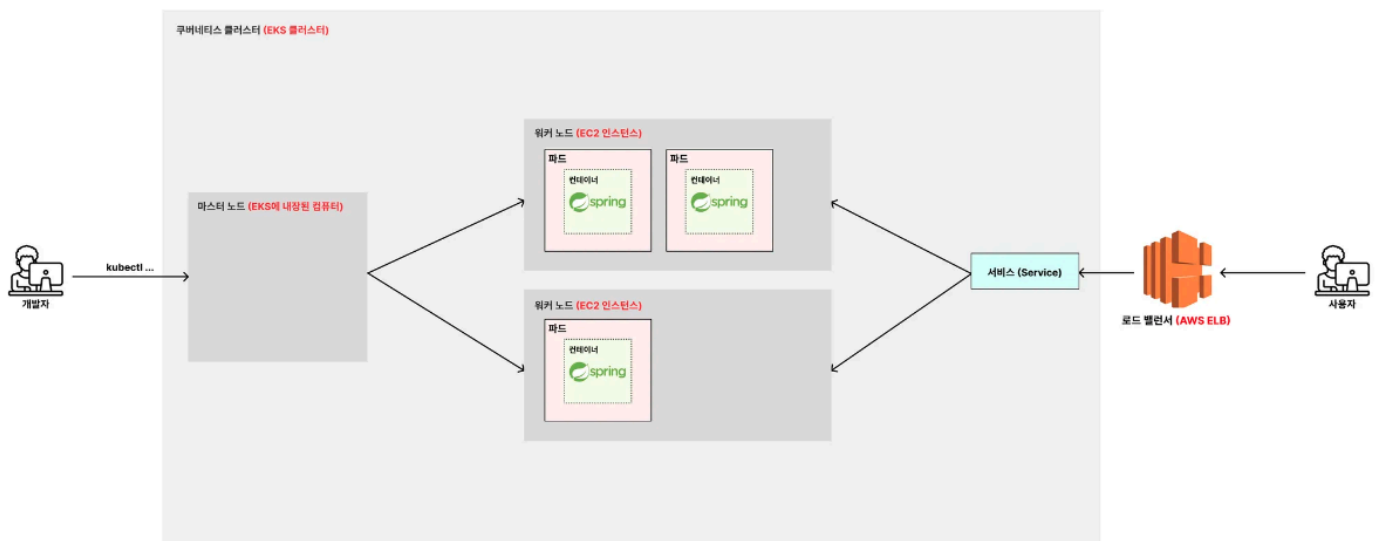
하지만 EKS를 다룰 때 아키텍처에서 기본적인 부분을 알아야 할 필요가 있어서 설명하고자 한다. 입문자 입장에서 알면 되는 부분만 간단화시켜서 살펴보자.

✓ 간단하게 표현한 쿠버네티스 아키텍처 구조



- **쿠버네티스 클러스터** : 하나의 마스터 노드와 여러 워커 노드들을 한 묶음으로 부르는 단위
- **마스터 노드** : 쿠버네티스 클러스터 전체를 관리하는 서버
- **워커 노드** : 쿠버네티스의 파드를 실행시키는 서버

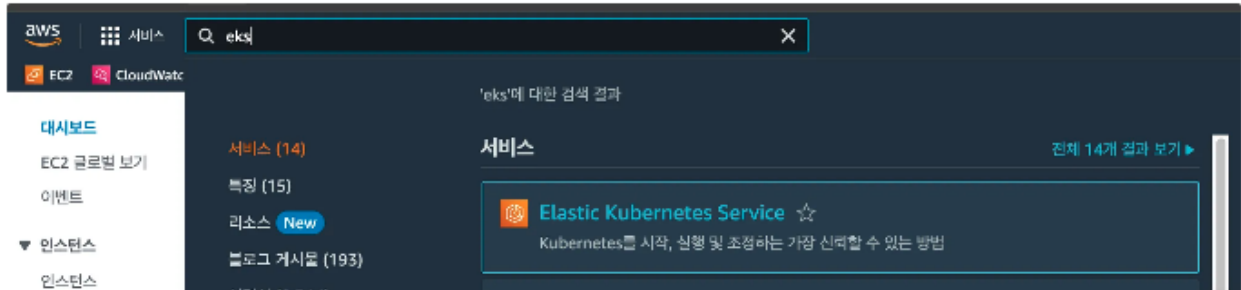
✅ EKS를 활용해 구성할 아키텍처 구조



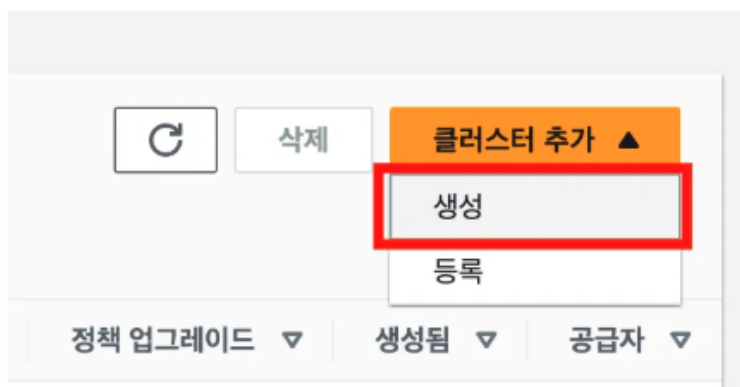
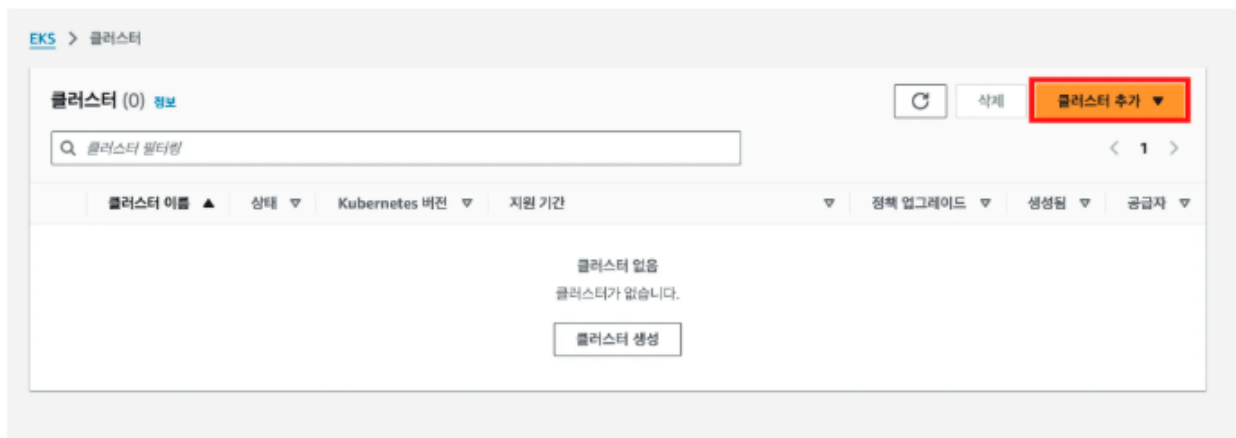
5. EKS 클러스터 생성하기

✓ EKS 클러스터 생성하기

1. EKS 서비스로 들어가기



2. 클러스터 추가하기



3. 클러스터 구성 셋팅하기

클러스터 구성 정보

이름

이 클러스터의 고유 이름을 입력합니다. 클러스터가 생성된 후에는 이 속성을 변경할 수 없습니다.

kube-practice

클러스터 이름은 문자 또는 숫자로 시작해야 하며 유니코드 문자 세트, 숫자, 하이픈 및 밑줄을 사용할 수 있습니다. 최대 길이는 100자입니다.

클러스터 IAM 역할 정보

클러스터 IAM 역할을 선택하여 Kubernetes 컨트롤 플레인인 사용자를 대신하여 AWS 리소스를 관리하도록 허용합니다. 클러스터 생성 후에는 이를 변경할 수 없습니다. 사용자 지정 역할을 새로 생성하려면 [Amazon EKS 사용 설명서](#)의 지침을 따르세요.

역할 선택

IAM 콘솔에서 역할 생성

4. EKS 클러스터의 IAM Role 생성하기

신뢰할 수 있는 엔터티 유형

☒ AWS 서비스

EC2, Lambda 등의 AWS 서비스가 이 계정에서 작업을 수행하도록 허용합니다.

☐ AWS 계정

사용자 또는 서드 파티에 속한 다른 AWS 계정의 엔터티가 이 계정에서 작업을 수행하도록 허용합니다.

☐ 웹 자격 증명

지정된 외부 웹 자격 증명 공급자의 인증된 사용자가 이 역할을 맡아 이 계정에서 작업을 수행하도록 허용합니다.

☐ SAML 2.0 연동

기업 디렉터리에서 SAML 2.0과 연동된 사용자가 이 계정에서 작업을 수행할 수 있도록 허용합니다.

☐ 사용자 지정 신뢰 정책

다른 사용자가 이 계정에서 작업을 수행할 수 있도록 사용자 지정 신뢰 정책을 생성합니다.

사용 사례

EC2, Lambda 등의 AWS 서비스가 이 계정에서 작업을 수행하도록 허용합니다.

서비스 또는 사용 사례

EKS

지정된 서비스에 대한 사용 사례를 선택합니다.

사용 사례

☐ EKS - Service

Allows EKS to manage clusters on your behalf.

☒ EKS - Cluster

Allows the cluster Kubernetes control plane to manage AWS resources on your behalf.

☐ EKS - Nodegroup

Allow EKS to manage nodegroups on your behalf.

☐ EKS - Fargate pod

Allows access to other AWS service resources that are required to run Amazon EKS pods on AWS Fargate.

☐ EKS - Fargate profile

Allows EKS to run Fargate tasks.

☐ EKS - Connector

Allows access to other AWS service resources that are required to connect to external clusters.

☐ EKS Local - Outpost

Allows Amazon EKS Local to call AWS services on your behalf.

☐ EKS - Pod Identity

Allows pods running in Amazon EKS cluster to access AWS resources.

취소

다음

권한 추가 정보

권한 정책 (1) 정보
 선택한 역할 유형에는 다음 정책이 포함됩니다.

정책 이름	유형
AmazonEKSClusterPolicy	AWS 관리형

▶ 권한 경계 설정 - 선택 사항

취소 이전 **다음**

5. 방금 생성한 **Role**을 선택해 지정하기

클러스터 구성 정보

이름
 이 클러스터의 고유 이름을 입력합니다. 클러스터가 생성된 후에는 이 속성을 변경할 수 없습니다.

 클러스터 이름은 문자 또는 숫자로 시작해야 하며 유니코드 문자 세트, 숫자, 하이픈 및 밑줄을 사용할 수 있습니다. 최대 길이는 100자입니다.

클러스터 IAM 역할 정보
 클러스터 IAM 역할을 선택하여 Kubernetes 컨트롤 플레인 사용자에 대신하여 AWS 리소스를 관리하도록 허용합니다. 클러스터 생성 후에는 이를 변경할 수 없습니다. 사용자 지정 역할을 새로 생성하려면 [Amazon EKS 사용 설명서](#)의 지침을 따르세요.

☒ kube-cluster-role
 arn:aws:iam::002177417362:role/kube-cluster-role

IAM 콘솔에서 역할 생성

6. 나머지 옵션은 그대로 두고 **다음** 버튼 누르기

Kubernetes 버전 설정

Kubernetes 버전 [정보](#)

이 클러스터의 Kubernetes 버전을 선택합니다.

1.31

정책 업그레이드 [정보](#)

다음 옵션 중 하나를 선택하세요. 표준 지원 기간이 적용되는 동안에는 나중에 설정을 전환할 수 있습니다.

☒ 확장됨

이 옵션은 출시일로부터 26개월 동안 Kubernetes 버전을 지원합니다. 추가 지원 기간에는 표준 지원 기간 종료 후 시작되는 시간당 추가 비용이 있습니다. 추가 지원이 종료되면 클러스터가 다음 버전으로 자동 업그레이드됩니다.

☐ 표준

이 옵션은 출시일로부터 14개월 동안 Kubernetes 버전을 지원합니다. 추가 비용은 없습니다. 표준 지원이 종료되면 클러스터가 다음 버전으로 자동 업그레이드됩니다.

클러스터 액세스 [정보](#)

IAM 보안 주체가 이 클러스터에 액세스하는 방법을 제어합니다.

무트스트랩 클러스터 관리자 액세스 [정보](#)

클러스터를 생성하는 IAM 보안 주체가 Kubernetes 클러스터 관리자 액세스 권한이 있는지 여부를 선택합니다.

☒ 클러스터 관리자 액세스 허용

IAM 보안 주체에 대한 클러스터 관리자 액세스를 허용합니다.

☐ 클러스터 관리자 액세스 허용 안 함

IAM 보안 주체에 대한 클러스터 관리자 액세스를 허용하지 않습니다.

클러스터 인증 모드 [정보](#)

클러스터가 인증된 IAM 보안 주체에 사용할 소스를 구성합니다.

☒ EKS API

클러스터가 인증된 IAM 보안 주체를 EKS 액세스 항목 API에서만 소싱합니다.

☐ EKS API 및 ConfigMap

클러스터가 인증된 IAM 보안 주체를 EKS 액세스 항목 API와 aws-auth ConfigMap 모두에서 소싱합니다.

☐ ConfigMap

클러스터가 인증된 IAM 보안 주체를 aws-auth ConfigMap에서만 소싱합니다.

암호 암호화 [정보](#)

일부 활성화되면 암호 암호화를 수정하거나 제거할 수 없습니다.

☐ KMS를 사용하여 Kubernetes 암호의 볼륨 암호화 활성화

볼륨 암호화는 Kubernetes 암호에 대한 추가 암호화 계층을 제공합니다.

ARC 영역 전환 [정보](#)

애플리케이션 트래픽을 전환하여 EKS 클러스터에서 장애가 발생한 가용 영역(AZ)으로부터 멀어지도록 합니다. 나중에 변경할 수 있습니다.

☐ 활성화됨

EKS는 클러스터를 ARC 영역 전환에 등록하여 영역 전환을 사용에 애플리케이션 트래픽이 AZ에서 멀어지도록 합니다.

☒ 비활성화된

EKS는 클러스터를 ARC 영역 전환에 등록하지 않습니다.

① 영역 전환을 시작하기 전에 AZ 장애에 대한 복원력을 갖추도록 클러스터 환경을 미리 설정해야 합니다.

[자세히 알아보기](#)

태그 (0) 정보

리소스와 연결된 태그가 없습니다.

새 태그 추가

최대 50개의 태그를 더 추가할 수 있습니다.

취소

다음

7. 다음 단계에서도 기본 옵션 그대로 두고 **다음** 버튼 누르기

네트워킹 정보

클러스터 생성 후에는 IP 주소 패밀리 및 서비스 IP 주소 범위를 변경할 수 없습니다.

VPC | 정보

EKS 클러스터 리소스에 사용할 VPC를 선택합니다. 새 VPC를 생성하려면 [VPC 콘솔](#) (으)로 이동합니다.

vpc-d0897abb | 기본값

서브넷 정보

클러스터와의 원활한 통신을 위해 제어 플레인 이 탄력적 네트워크 인터페이스(ENI)를 배치할 수 있는 서브넷을 VPC 내에서 선택합니다. 새 서브넷을 생성하려면 [VPC 콘솔](#) (으)의 해당 페이지로 이동합니다.

서브넷 선택

subnet-6f48a514

×

subnet-3a9dcc76

×

ap-northeast-2b 172.31.32.0/20

ap-northeast-2c 172.31.16.0/20

subnet-1f3e6043

×

subnet-35db365e

×

ap-northeast-2d 172.31.48.0/20

ap-northeast-2a 172.31.0.0/20

선택한 서브넷 지우기

보안 그룹 | 정보

컨트롤 플레인 서브넷에서 생성된 EKS 관리형 탄력적 네트워크 인터페이스에 적용할 보안 그룹을 선택합니다. 새 보안 그룹을 생성하려면 [VPC 콘솔](#) (으)의 해당 페이지로 이동합니다.

보안 그룹 선택

클러스터 IP 주소 패밀리 선택 | 정보

클러스터의 Pod 및 서비스에 대한 IP 주소 유형을 지정합니다.

☒ IPv4

☐ IPv6

Kubernetes 서비스 IP 주소 블록 구성 | 정보

☐ 클러스터 서비스가 IP 주소를 수신할 범위를 지정합니다.

클러스터 엔드포인트 액세스 정보

Kubernetes API 서버 엔드포인트에 대한 액세스 권한을 구성합니다.

☐ 퍼블릭

VPC 외부에서 클러스터 엔드포인트에 액세스할 수 있습니다. 작업자 노드 트래픽은 엔드포인트에 연결하기 위해 VPC를 벗어납니다.

☒ 퍼블릭 및 프라이빗

VPC 외부에서 클러스터 엔드포인트에 액세스할 수 있습니다. 엔드포인트에 대한 작업자 노드 트래픽은 VPC 내에 유지됩니다.

☐ 프라이빗

클러스터 엔드포인트는 VPC를 통해서만 액세스할 수 있습니다. 엔드포인트에 대한 작업자 노드 트래픽은 VPC 내에 유지됩니다.

▶ 고급 설정

취소

이전

다음

관찰성 구성

▶ 관찰성 정보

지표

Prometheus [정보](#)

☒ Prometheus 지표를 Amazon Managed Service for Prometheus로 전송

Amazon Managed Service for Prometheus를 사용하여 애플리케이션 및 인프라 지표를 모니터링하세요. 이러한 지표에는 시스템 상태 및 성능 데이터가 포함됩니다.

CloudWatch [정보](#)

- ① CloudWatch Observability 추가 기능을 통해 클러스터에서 CloudWatch Observability를 활성화할 수 있습니다. 클러스터가 생성된 후 추가 기능 탭으로 이동하여 CloudWatch Observability 추가 기능을 설치하세요. 그런 다음 CloudWatch Application Signals 및 Container Insights를 활성화하고 CloudWatch로 텔레메트리 모으기를 시작하세요.

제어 플레인 로깅 [정보](#)

Amazon EKS 제어 플레인에서 CloudWatch Logs로 감사 및 진단 로그를 전송합니다.

☒ API 서버

클러스터에 대한 API 요청과 관련된 로그입니다.

☒ 감사

Kubernetes API를 통한 클러스터 액세스와 관련된 로그입니다.

☒ Authenticator

클러스터에 대한 인증 요청과 관련된 로그입니다.

☒ 컨트롤러 관리자

클러스터 컨트롤러 상태와 관련된 로그입니다.

☒ 스케줄러

예약 결정과 관련된 로그입니다.

취소

이전

다음

추가 기능 선택

여러 범주의 추가 기능을 검토한 다음 추가 기능을 선택하여 클러스터를 개선합니다.

Amazon EKS 추가 기능 (11) 정보

kube-proxy 정보 ☒

클러스터 내에서 포트 네트워킹을 활성화합니다.

카테고리
networking

Amazon VPC CNI 정보 ☒

클러스터 내에서 포트 네트워킹을 활성화합니다.

카테고리
networking

CoreDNS 정보 ☒

클러스터 내에서 서비스 검색을 활성화합니다.

카테고리
networking

Amazon EKS Pod Identity 에이전트 정보 ☒

EKS Pod Identity 에이전트를 설치하고 EKS Pod Identity를 사용하여 Kubernetes 서비스 계정을 통해 포드에 AWS IAM 권한을 부여합니다.

카테고리
security

Amazon GuardDuty EKS 런타임 모니터링 정보 ☐

클러스터 내에 EKS 런타임 모니터링 추가 기능을 설치합니다. Amazon GuardDuty 내에서 EKS 런타임 모니터링을 활성화해야 합니다.

카테고리
security

취소

이전

다음

선택한 추가 기능 설정 구성

설정을 선택하여 클러스터에 대한 추가 기능을 구성합니다.

Amazon VPC CNI 정보

[추가 기능 제거](#)

카테고리
networking

상태
✔ 설치 준비 완료

버전
이 추가 기능의 버전을 선택합니다.

v1.18.3-eksbuild.2 ▼

CoreDNS 정보

[추가 기능 제거](#)

카테고리
networking

상태
✔ 설치 준비 완료

버전
이 추가 기능의 버전을 선택합니다.

v1.11.3-eksbuild.1 ▼

Amazon EKS Pod Identity 에이전트 정보

[추가 기능 제거](#)

카테고리
security

상태
✔ 설치 준비 완료

버전
이 추가 기능의 버전을 선택합니다.

v1.3.2-eksbuild.2 ▼

kube-proxy 정보

[추가 기능 제거](#)

카테고리
networking

상태
✔ 설치 준비 완료

버전
이 추가 기능의 버전을 선택합니다.

v1.31.0-eksbuild.2 ▼

[취소](#)[이전](#)[다음](#)

8. '검토 및 생성' 페이지에서 **생성** 버튼 누르기

추가 기능 이름	유형	상태
coredns	networking	✅ 설치 준비 완료
eks-pod-identity-agent	security	✅ 설치 준비 완료
kube-proxy	networking	✅ 설치 준비 완료
vpc-cni	networking	✅ 설치 준비 완료

5단계: 버전 편집

선택한 추가 기능 버전 (4)

추가 기능 이름	버전
coredns	v1.11.3-eksbuild.1
eks-pod-identity-agent	v1.3.2-eksbuild.2
kube-proxy	v1.31.0-eksbuild.2
vpc-cni	v1.18.3-eksbuild.2

취소 이전 생성

9. 생성이 완료될 때까지 기다리기

약 **10분~15분** 정도 걸린다.

Amazon Elastic Kubernetes Service

클러스터

Amazon EKS Anywhere

Enterprise 구독

관련 서비스

Amazon ECR

AWS Batch

문서 설정

성명서

이동식 계층

kube-practice 클러스터가 생성되는 동안 관리형 노드 그룹 및 Fargate 프로파일들을 추가할 수 있습니다. 이제 기다려 주십시오.

EKS

클러스터

kube-practice

클러스터 정보

상태

Kubernetes 버전

지운 기간

공급자

🔄 생성 중

1.21

① 2025년 11월 26일까지 보존 지

EKS

개요

리소스

컴퓨팅

비즈니스

추가 기능

액세스

관할권

임그라이드 인사이트

임타이브 기록

대그

세부 정보

API 서버 엔드포인트

OpenID Connect 공급자 URL

생성일

연동 기간

클러스터 IAM 역할 ARN

클러스터 ARN

-

-

aws-eksctl-northeast-2-002177417362-1

kube-practice



클러스터 삭제

▼ 클러스터 정보 정보

상태  완료	Kubernetes 버전 <small>정보</small> 1.31	지원 기간  2025년 11월 26일까지 표준 지원	공급자 EKS
---	---	--	------------

개요

리소스

컴퓨팅

네트워킹

추가 기능 **1**

액세스

관할성

업그레이드 인사이트

업데이트 기록

태그

6. EKS 워커 노드 추가하기

✓ EKS 워커 노드 추가하기

1. 노드 그룹 추가하기

The screenshot shows the AWS Management Console interface for an EKS cluster named 'kube-practice'. The breadcrumb navigation at the top indicates the path: EKS > 클러스터 > kube-practice. On the right side, there are buttons for 'Refresh' (a circular arrow icon) and 'Delete Cluster' (클러스터 삭제).

The main section is titled 'kube-practice' and contains a 'Cluster Information' (클러스터 정보) tab. This tab displays several key metrics: the cluster status is 'Active' (활성) with a green checkmark; the Kubernetes version is 1.31; the support period (지원 기간) is 'Standard Support until November 26, 2025' (표준 지원 2025년 11월 26일까지); and the provider (공급자) is EKS.

Below the cluster information, there is a horizontal navigation bar with tabs for 'Overview' (개요), 'Resources' (리소스), 'Add-ons' (추가 기능), 'Access' (액세스), 'Monitoring' (관찰성), 'Upgrade Path' (업그레이드 인사이트), 'Update History' (업데이트 기록), and 'Tags' (태그). The 'Add-ons' tab is currently selected and highlighted with a red box.

The 'Add-ons' tab shows a list of installed add-ons, which is currently empty. Below the list, there is a message: 'No add-ons found. This cluster may not have the necessary permissions to view add-ons.' (이 클러스터에 노드(가) 있거나 해당 클러스터를 볼 수 있는 권한이 없습니다.)

At the bottom of the console, there is a section for 'Node Groups' (노드 그룹). It shows a list of node groups, which is currently empty. Below the list, there is a message: 'No node groups found. This cluster does not have any node groups. Amazon EKS managed node groups are not shown in the AWS console.' (이 클러스터에는 노드 그룹이 없습니다. Amazon EKS 관리형 노드 그룹이 아닌 노드는 AWS 콘솔에 표시되지 않습니다.)

At the bottom right of the 'Node Groups' section, there is a button labeled 'Add Node Group' (노드 그룹 추가), which is highlighted with a red box.

2. 노드 그룹 구성 셋팅하기

노드 그룹 구성 정보

노드 그룹은 Amazon EKS 클러스터에 컴퓨팅 용량을 제공하는 EC2 인스턴스의 그룹입니다. 클러스터에는 여러 노드 그룹을 추가할 수 있습니다.

노드 그룹 구성

노드 그룹을 생성한 후에는 이러한 속성을 변경할 수 없습니다.

이름
이 노드 그룹에 대한 고유한 이름을 할당합니다.


노드 그룹 이름은 문자 또는 숫자로 시작해야 하며 유니코드 문자 세트, 숫자, 하이픈 및 밑줄을 포함할 수 있습니다. 최대 길이는 63자입니다.


노드 IAM 역할 정보
노드에서 사용할 IAM 역할을 선택합니다. 새 역할을 생성하려면 [IAM 콘솔](#)(으)로 이동합니다.

역할 선택 ▼

↻

❗ 관리형 노드 그룹 삭제 시 서비스가 중단될 수 있기 때문에 선택한 역할은 자체 관리형 노드 그룹에서 사용하지 않아야 합니다.

[자세히 알아보기](#) 

IAM 콘솔에서 역할 생성 

3. EKS 노드 그룹의 IAM Role 생성하기

신뢰할 수 있는 엔티티 선택 정보

신뢰할 수 있는 엔티티 유형

☒ AWS 서비스

EC2, Lambda 등의 AWS 서비스가 이 계정에서 작업을 수행하도록 허용합니다.

☐ AWS 계정

사용자 또는 서드 파티어 혹은 다른 AWS 계정의 엔티티가 이 계정에서 작업을 수행하도록 허용합니다.

☐ 웹 자격 증명

지정된 외부 웹 자격 증명 공급자와 연동된 사용자가 이 역할을 빌려 이 계정에서 작업을 수행하도록 허용합니다.

☐ SAML 2.0 연동

기업 디렉터리에서 SAML 2.0과 연동된 사용자가 이 계정에서 작업을 수행할 수 있도록 허용합니다.

☐ 사용자 지정 신뢰 정책

다른 사용자가 이 계정에서 작업을 수행할 수 있도록 사용자 지정 신뢰 정책을 생성합니다.

사용 사례

EC2, Lambda 등의 AWS 서비스가 이 계정에서 작업을 수행하도록 허용합니다.

서비스 또는 사용 사례

EC2

지정된 서비스에 대한 사용 사례를 선택합니다.

사용 사례

☒ EC2

Allows EC2 instances to call AWS services on your behalf.

☐ EC2 Role for AWS Systems Manager

Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.

☐ EC2 Spot Fleet Role

Allows EC2 Spot Fleet to request and terminate Spot instances on your behalf.

☐ EC2 - Spot Fleet Auto Scaling

Allows Auto Scaling to access and update EC2 spot fleets on your behalf.

☐ EC2 - Spot Fleet Tagging

Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.

☐ EC2 - Spot Instances

Allows EC2 Spot instances to launch and manage spot instances on your behalf.

☐ EC2 - Spot Fleet

Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.

☐ EC2 - Scheduled Instances

Allows EC2 Scheduled instances to manage instances on your behalf.

취소

다음

권한 정책 3,968 정보

새 역할에 연결할 정책을 하나 이상 선택합니다.

Q 검색

필터링 기준 유형

모든 유형

< 1 2 3 4 5 6 7 ... 49 >

	정책 이름	유형	설명
<input type="checkbox"/>	AdministratorAccess	AWS 관리형 - 직무	Provides full access to AWS services an...
<input type="checkbox"/>	AdministratorAccess-Amplify	AWS 관리형	Grants account administrative permis...
<input type="checkbox"/>	AdministratorAccess-AWSElasticBeans...	AWS 관리형	Grants account administrative permis...
<input type="checkbox"/>	AlexaForBusinessDeviceSetup	AWS 관리형	Provide device setup access to AlexaFo...
<input type="checkbox"/>	AlexaForBusinessFullAccess	AWS 관리형	Grants full access to AlexaForBusiness ...
<input type="checkbox"/>	AlexaForBusinessGatewayExecution	AWS 관리형	Provide gateway execution access to A...
<input type="checkbox"/>	AlexaForBusinessLifesizeDelegatedAc...	AWS 관리형	Provide access to Lifesize AVS devices
<input type="checkbox"/>	AlexaForBusinessPolyDelegatedAcces...	AWS 관리형	Provide access to Poly AVS devices
<input type="checkbox"/>	AlexaForBusinessReadOnlyAccess	AWS 관리형	Provide read only access to AlexaForB...
<input type="checkbox"/>	AmazonAPIGatewayAdministrator	AWS 관리형	Provides full access to create/edit/dele...
<input type="checkbox"/>	AmazonAPIGatewayInvokeFullAccess	AWS 관리형	Provides full access to invoke APIs in A...
<input type="checkbox"/>	AmazonAPIGatewayPushToCloudWat...	AWS 관리형	Allows API Gateway to push logs to us...
<input type="checkbox"/>	AmazonAppFlowFullAccess	AWS 관리형	Provides full access to Amazon AppFlo...
<input type="checkbox"/>	AmazonAppFlowReadOnlyAccess	AWS 관리형	Provides read only access to Amazon A...
<input type="checkbox"/>	AmazonAppStreamFullAccess	AWS 관리형	Provides full access to Amazon AppStr...
<input type="checkbox"/>	AmazonAppStreamPCAAccess	AWS 관리형	Amazon AppStream 2.0 access to AWS...
<input type="checkbox"/>	AmazonAppStreamReadOnlyAccess	AWS 관리형	Provides read only access to Amazon A...
<input type="checkbox"/>	AmazonAppStreamServiceAccess	AWS 관리형	Default policy for Amazon AppStream ...
<input type="checkbox"/>	AmazonAthenaFullAccess	AWS 관리형	Provide full access to Amazon Athena ...
<input type="checkbox"/>	AmazonAugmentedAIFullAccess	AWS 관리형	Provides access to perform all operati...

▶ 권한 경계 설정 - 선택 사항

취소

이전

다음

→ 이미 권한이 체크된 상태

이름 지정, 검토 및 생성

역할 세부 정보

역할 이름

이 역할을 식별하는 데 사용되는 이름을 입력합니다.

kube-node-group-role

최대 64자입니다. 영소문 'a-z', 밑줄 '_' 문자를 사용하세요.

설명

이 역할에 대해 간단한 설명을 추가합니다.

Allows EC2 instances to call AWS services on your behalf.

최대 문자 수: 1000. 문자[A-Z 및 a-z], 숫자[0-9], 점, 세 줄 또는 다음 문자 중 하나를 사용합니다: *,=, @,~/[()!%\$%^&*]:"'

1단계: 신뢰할 수 있는 엔터티 선택

편집

신뢰 정책

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
8       ],
9       "Principal": {
10        "Service": [
11          "ec2.amazonaws.com"
12        ]
13      }
14    ]
15  }
```

2단계: 권한 추가

편집

권한 정책 요약

정책 이름	유형	다음으로 연결됨
AmazonEC2ContainerRegistryReadOnly	AWS 관리형	권한 정책
AmazonEKS_CNI_Policy	AWS 관리형	권한 정책
AmazonEKSEKSPolicy	AWS 관리형	권한 정책

3단계: 태그 추가

태그 추가 - 선택 사항 정보

태그는 리소스를 식별, 정리 또는 검색하는 데 도움이 되도록 AWS 리소스에 추가할 수 있는 가-값 쌍입니다.

리소스와 연결된 태그가 없습니다.

새 태그 추가

최대 50개의 태그를 더 추가할 수 있습니다.

취소

이전

역할 생성

4. 방금 생성한 **Role** 선택해 지정하기

노드 그룹 구성

노드 그룹을 생성한 후에는 이러한 속성을 변경할 수 없습니다.

이름
이 노드 그룹에 대한 고유한 이름을 할당합니다.

kube-practice-node-group

노드 그룹 이름은 문자 또는 숫자로 시작해야 하며 유니코드 문자 세트, 숫자, 하이픈 및 밑줄을 포함할 수 있습니다. 최대 길이는 63자입니다.

노드 IAM 역할 [정보](#)
노드에서 사용할 IAM 역할을 선택합니다. 새 역할을 생성하려면 [IAM 콘솔](#)(으)로 이동합니다.

kube-node-group-role ▲

🔍 |역할 필터링

kube-node-group-role

arn:aws:iam::002177417362:role/kube-node-group-role

자세히 알아보기 ↗

🔄

IAM 콘솔에서 역할 생성 ↗

시작 템플릿 정보

노드 그룹을 생성한 후에는 이러한 속성을 변경할 수 없습니다.

☒ 시작 템플릿 사용

EC2 시작 템플릿을 사용하여 이 노드 그룹을 구성합니다.

Kubernetes 레이블 정보

이 노드 그룹에는 레이블이 없습니다.

레이블 추가

추가할 수 있는 나머지 레이블: 50

Kubernetes 테인트 정보

이 노드 그룹에는 테인트가 없습니다.

테인트 추가

추가할 수 있는 나머지 테인트: 50

태그 정보

리소스와 연결된 태그가 없습니다.

새 태그 추가

최대 50개의 태그를 더 추가할 수 있습니다.

취소

다음

5. 컴퓨팅 및 조정 구성 설정하기

컴퓨팅 및 조정 구성 설정

노드 그룹 컴퓨팅 구성

노드 그룹을 생성한 후에는 이러한 속성을 변경할 수 없습니다.

AMI 유형 [정보](#)

노드에 대한 EKS 최적화 Amazon Machine Image를 선택합니다.

Amazon Linux 2 Arm (AL2_ARM_64) ▼

용량 유형

이 노드 그룹에 대한 용량 구매 옵션을 선택합니다.

On-Demand ▼

인스턴스 유형 [정보](#)

이 노드 그룹에 대해 선호하는 인스턴스 유형을 선택합니다.

🔍 인스턴스 유형 입력

t4g.small ✕

vCPU: 2 vCPUs Memory: 2 GiB Network: Up to 5 Gigabit Max ENI: 3 Max IPs: 12

디스크 크기

각 노드에 연결되는 EBS 볼륨의 크기를 선택합니다.

20

GiB

노드 그룹 조정 구성

원하는 크기

그룹에서 처음에 시작할 노드 수를 설정합니다.

노드

원하는 노드 크기는 0보다 크거나 같아야 함

최소 크기

그룹에서 축소할 수 있는 최소 노드 수를 설정합니다.

노드

최소 노드 크기는 0보다 크거나 같아야 함

최대 크기

그룹에서 확장할 수 있는 최대 노드 수를 설정합니다.

노드

최대 노드 크기는 1보다 크거나 같아야 하며 최소 크기보다 작을 수 없음

노드 그룹 업데이트 구성 정보

최대 사용 불가

노드 그룹 버전 업데이트 중에 사용할 수 있는 노드의 최대 허용 수 또는 백분율을 설정합니다.

☒ 수

숫자 입력

☐ 백분율

백분율 지정

값

노드

노드 수는 0보다 커야 합니다.

취소

이전

다음

6. 나머지 옵션은 그대로 두기

네트워킹 지정

노드 그룹 네트워크 구성

노드 그룹을 생성한 후에는 이러한 속성을 변경할 수 없습니다.

서브넷 정보

노드가 실행될 VPC의 서브넷을 지정합니다. 새 서브넷을 생성하려면 [VPC 콘솔](#)의 해당 페이지로 이동합니다.

서브넷 선택

subnet-6f48a514
ap-northeast-2b 172.31.32.0/20

subnet-3a9dcc76
ap-northeast-2c 172.31.16.0/20

subnet-1f3e6043
ap-northeast-2d 172.31.48.0/20

subnet-35db365e
ap-northeast-2a 172.31.0.0/20

선택한 서브넷 지우기

☐ 노드에 대한 원격 액세스 허용 [정보](#)

취소

이전

다음

Amazon Linux 2 (AL2_x86_64)

노드 그룹 조정 구성

원하는 크기	최소 크기	최대 크기
2 노드	2 노드	2 노드

노드 그룹 업데이트 구성

최대 사용 불가
1 노드

3단계: 네트워킹

편집

노드 그룹 네트워크 구성

서브넷 subnet-6f48a514 subnet-3a9dcc76 subnet-1f3e6043 subnet-35db365e	노드에 대한 원격 액세스 허용 off
---	-------------------------

취소

이전

생성

7. 노드 그룹이 생성될 때까지 기다리기

EKS > 클러스터 > kube-practice > 노드 그룹 > kube-practice-node-group

kube-practice-node-group

노드 그룹 구성 정보

Kubernetes 버전 1.31	AMI 유형 정보 AL2_x86_64	상태 ⏸ 생성 중
AMI 릴리스 버전 정보 1.31.0-20241106	인스턴스 유형 t3a.small	디스크 크기 20 GiB

세부 정보

노드 그룹 ARN arn:aws:eks:ap-northeast-2:002177417362:nodegroup/kube-practice/kube-practice-node-group/2ec98331-e75d-29c9-30ff-9db3211845a3	Auto Scaling 그룹 이름 eks-kube-practice-node-group-2ec98331-e75d-29c9-30ff-9db3211845a3	용량 유형 On-Demand	서브넷 subnet-6f48a514 subnet-3a9dcc76 subnet-1f3e6043 subnet-35db365e
생성됨 몇 초 전	노드 IAM 역할 ARN arn:aws:iam::002177417362:role/kube-node-group-role	원하는 크기 2 노드	노드에 대한 원격 액세스 허용 off
		최소 크기 2 노드	
		최대 크기 2 노드	

EKS > 클러스터 > kube-practice > 노드 그룹 > kube-practice-node-group

kube-practice-node-group

노드 그룹 구성 정보

Kubernetes 버전 1.31	AMI 유형 정보 AL2_x86_64	상태 ✅ 활성
AMI 릴리스 버전 정보 1.31.0-20241106	인스턴스 유형 t3a.small	디스크 크기 20 GiB

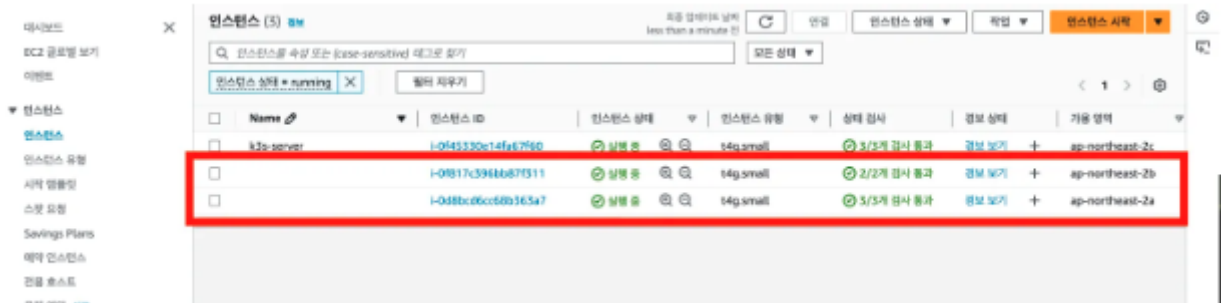
세부 정보

노드 그룹 ARN arn:aws:eks:ap-northeast-2:002177417362:nodegroup/kube-practice/kube-practice-node-group/2ec98331-e75d-29c9-30ff-9db3211845a3	Auto Scaling 그룹 이름 eks-kube-practice-node-group-2ec98331-e75d-29c9-30ff-9db3211845a3	용량 유형 On-Demand	서브넷 subnet-6f48a514 subnet-3a9dcc76 subnet-1f3e6043 subnet-35db365e
생성됨 5분 전	노드 IAM 역할 ARN arn:aws:iam::002177417362:role/kube-node-group-role	원하는 크기 2 노드	노드에 대한 원격 액세스 허용 off
		최소 크기 2 노드	
		최대 크기 2 노드	

5분 정도 기다리면 활성화된다.

8. EC2 인스턴스 확인하기

EC2 인스턴스 페이지에 들어가면 새로운 EC2 인스턴스 2개가 생성되어 있는 걸 확인할 수 있다. EKS 클러스터에서 하나의 워커 노드(Worker Node)가 하나의 EC2 인스턴스에서 실행되는 구조이기 때문이다.



7. 로컬에서 **EKS** 클러스터 조정할 수 있게 셋팅하기

✅ 로컬에서 **EKS** 클러스터 조정할 수 있게 셋팅하기

1. 현재 **kubectl**이 어떤 클러스터 환경에서 작동되고 있는 지 확인하기

```
$ kubectl config get-contexts
```

```
jaeseong ~ kubectl config get-contexts
CURRENT  NAME                CLUSTER                AUTHINFO                NAMESPACE
*         docker-desktop      docker-desktop         docker-desktop
```

현재는 **kubectl**이 Docker Desktop의 쿠버네티스 클러스터를 작동시키고 있는 걸 알 수 있다.

2. **kubectl**에 **EKS** 클러스터 추가하기

```
# aws eks --region ap-northeast-2 update-kubeconfig --name
<EKS 클러스터 이름>
$ aws eks --region ap-northeast-2 update-kubeconfig --name
kube-practice
```

3. 잘 적용 됐는지 확인하기

```
$ kubectl config get-contexts
```

```
* jaeseong ~ kubectl config get-contexts
CURRENT  NAME                CLUSTER                AUTHINFO                NAMESPACE
*         arn:aws:eks:ap-northeast-2:002177417362:cluster/kube-practice
docker-desktop      arn:aws:eks:ap-northeast-2:002177417362:cluster/kube-practice
                                arn:aws:eks:ap-northeast-2:002177417362:cluster/kube-practice
                                docker-desktop
```

[참고]

```
# 다른 클러스터로 전환
$ kubectl config use-context <컨텍스트 이름>

# 특정 컨텍스트 삭제
$ kubectl config unset contexts.<컨텍스트 이름>
```

8. EKS에 백엔드(Spring Boot) 서버 배포하기 (+ RDS, ECR)

✓ EKS에 백엔드(Spring Boot) 서버 배포하기 (+ RDS, ECR)

1. 매니페스트 파일 수정하기 spring-deployment.yaml

```
apiVersion: apps/v1
kind: Deployment

# Deployment 기본 정보
metadata:
  name: spring-deployment # Deployment 이름

# Deployment 세부 정보
spec:
  replicas: 3 # 생성할 파드의 복제본 개수
  selector:
    matchLabels:
      app: backend-app # 아래에서 정의한 Pod 중 'app: backend-app'이라는 값을
                        # 가진 파드를 선택

# 배포할 Pod 정의
template:
  metadata:
    labels: # 레이블 (= 카테고리)
      app: backend-app
  spec:
    containers:
      - name: spring-container # 컨테이너 이름
        image:
002177417362.dkr.ecr.ap-northeast-2.amazonaws.com/kube-ecr:2.0 #
        컨테이너를 생성할 때 사용할 이미지
        ports:
          - containerPort: 8080 # 컨테이너에서 사용하는 포트를 명시적으로 표현
        env:
          - name: DB_HOST
            valueFrom:
              configMapKeyRef:
                name: spring-config
                key: db-host
          - name: DB_PORT
            valueFrom:
              configMapKeyRef:
                name: spring-config
                key: db-port
          - name: DB_NAME
```

```

valueFrom:
  configMapKeyRef:
    name: spring-config
    key: db-name
- name: DB_USERNAME
valueFrom:
  secretKeyRef:
    name: spring-secret
    key: db-username
- name: DB_PASSWORD
valueFrom:
  secretKeyRef:
    name: spring-secret
    key: db-password

```

spring-secret.yaml

```

apiVersion: v1
kind: Secret
type: Opaque # 임의의 사용자 정의 데이터를 저장할 때 사용하는 타입

# Secret 기본 정보
metadata:
  name: spring-secret # Secret 이름

# Key, Value 형식으로 값 저장
stringData:
  db-username: admin
  db-password: password

```

spring-config.yaml

```

apiVersion: v1
kind: ConfigMap

# ConfigMap 기본 정보
metadata:
  name: spring-config # ConfigMap 이름

# Key, Value 형식으로 설정값 저장
data:
  db-host: kube-database.coseefawhrzc.ap-northeast-2.rds.amazonaws.com
  db-port: "3306"
  db-name: mydb

```

spring-service.yaml

```
apiVersion: v1
kind: Service

# Service 기본 정보
metadata:
  name: spring-service

# Service 세부 정보
spec:
  type: LoadBalancer # Service의 종류
  selector:
    app: backend-app # 실행되고 있는 파드 중 'app: backend-app'이라는 값을 가진
    파드와 서비스를 연결
  ports:
    - protocol: TCP # 서비스에 접속하기 위한 프로토콜
      port: 80 # 외부에서 사용자가 요청을 보낼 때 사용하는 포트 번호
      targetPort: 8080 # 매핑하기 위한 파드의 포트 번호
      nodePort: 30000 # 외부에서 사용자들이 접근하게 될 포트 번호
```

- **NodePort** : 쿠버네티스 내부에서 해당 서비스에 접속하기 위한 포트를 열고 외부에서 접속 가능하도록 한다. ⇒ 들어오는 요청을 여러 **Worker Node**로 트래픽을 분산시키지 않는다.
- **ClusterIP** : 쿠버네티스 내부에서만 통신할 수 있는 IP 주소를 부여. 외부에서는 요청할 수 없다.
- **LoadBalancer** : 외부의 로드밸런서(AWS의 로드밸런서 등)를 활용해 외부에서 접속할 수 있도록 연결한다. ⇒ 들어오는 요청을 여러 **Worker Node**로 트래픽을 분산시켜준다.

2. 매니페스트 파일을 통해 오브젝트 생성하기

```
$ kubectl apply -f spring-secret.yaml
$ kubectl apply -f spring-config.yaml
$ kubectl apply -f spring-deployment.yaml
$ kubectl apply -f spring-service.yaml
```

3. 잘 생성 됐는지 확인하기

```
$ kubectl get secret
$ kubectl get configmap
$ kubectl get deployment
```



```
$ kubectl get pods
```

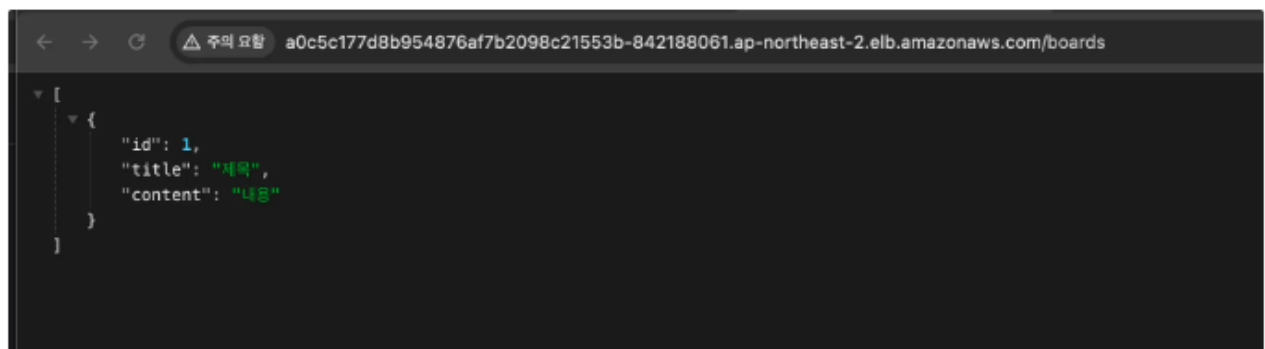
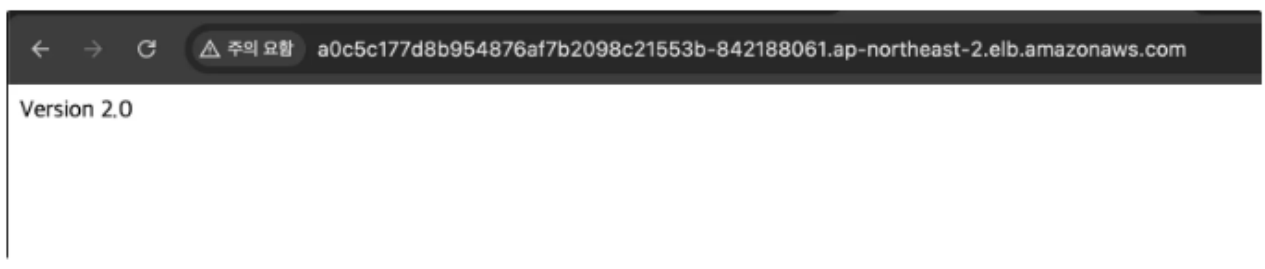
```
$ kubectl get service
```

```
jaeseong ~/Downloads/kubernetes-manifests ▶ main ▶ kubectl get service
```

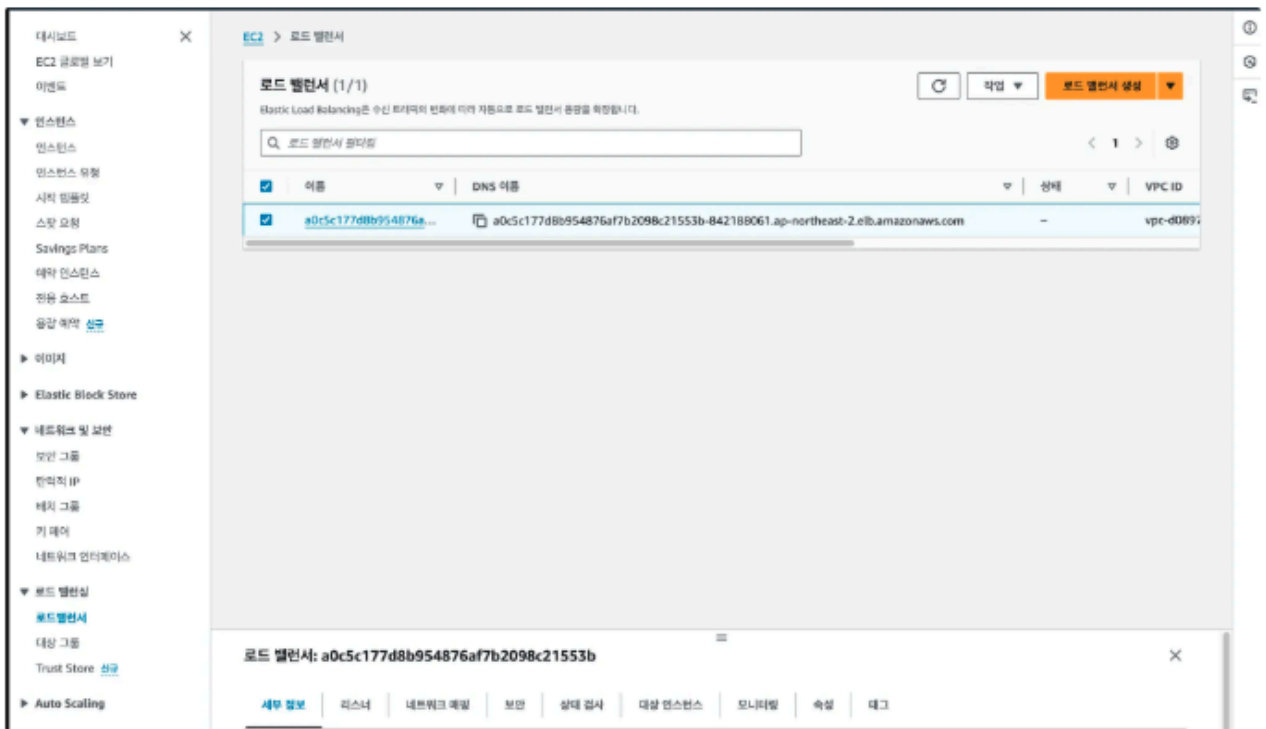
NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.100.0.1	<none>	443/TCP	9m3s
spring-service	LoadBalancer	10.100.251.204	a0c5c177d8b954876af7b2098c21553b-842188061.ap-northeast-2.elb.amazonaws.com	80:32371/TCP	33s

Service의 **Type**을 **LoadBalancer**로 했더니 외부에서 접속할 수 있는 주소가 주어졌다.

4. Service의 주소로 접속해보기\



5. 정말 로드밸런서가 생성 됐는지 확인하기



✅ 아키텍처 다시 한 번 짚어보기



9. 비용 나가지 않게 **EC2, RDS, ECR, EKS** 종료하기

✓ 비용 나가지 않게 **EKS** 종료하기

1. 실행 중인 오브젝트 종료하기

실행 중인 파드가 있으면 **EKS**의 노드 그룹이 삭제되지 않는다.

```
$ kubectl delete all --all
```

2. **EKS** 노드 그룹 삭제하기

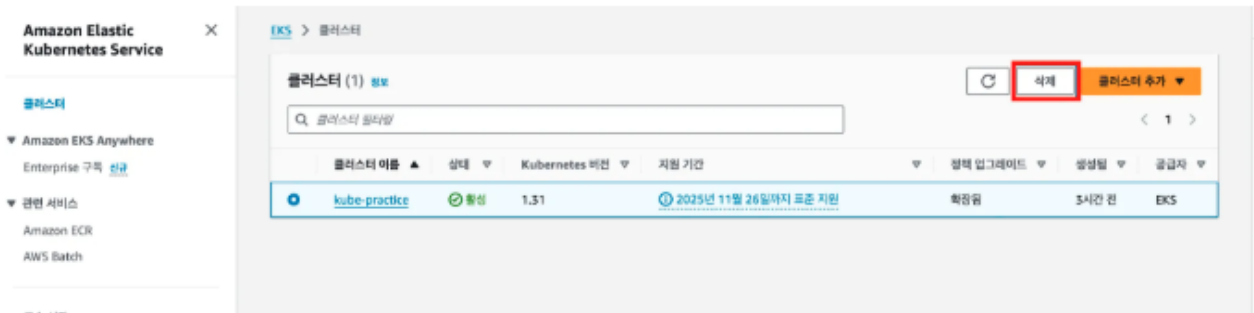
The screenshot shows the Amazon Elastic Kubernetes Service (EKS) console for a cluster named 'kube-practice'. The 'Nodes' tab is selected, showing a table of nodes. The 'kube-practice-node-group' is highlighted, and the 'Delete' button is visible. The console also shows the 'Fargate Profiles' section, which is currently empty.

노드 이름	인스턴스 유형	노드 그룹	생성됨	상태
ip-172-51-11-113.ap-northeast-2.compute.internal	t4g.small	kube-practice-node-group	생성됨 37분 전	준비 완료
ip-172-51-33-253.ap-northeast-2.compute.internal	t4g.small	kube-practice-node-group	생성됨 37분 전	준비 완료

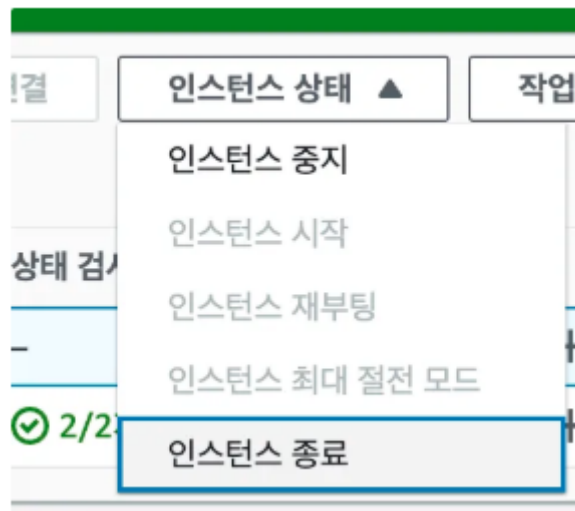
그룹 이름	원하는 크기	AMI 릴리스 버전	사막 앰블럼	상태
kube-practice-node-group	2	1.31.0-20241106	-	활성

3. **EKS** 클러스터 삭제하기

EKS의 노드 그룹이 삭제가 완료돼야만 **EKS** 클러스터를 삭제할 수 있다. **EKS** 노드 그룹이 삭제될 때까지 조금만 기다리자.



✓ 비용 나가지 않게 **EC2** 종료하기





✓ 비용 나가지 않게 **RDS** 종료하기



kube-database 인스턴스를 삭제



DB 인스턴스 **kube-database**을(를) 영구적으로 삭제합니다. 이 작업은 실행 취소할 수 없습니다.

 이 작업을 계속하면 인스턴스와 모든 해당 콘텐츠가 삭제되고 관련 리소스에 영향을 줄 수 있습니다. [자세히 알아보기](#) 

☐ 최종 스냅샷 생성

DB 인스턴스를 삭제하기 전에 최종 DB 스냅샷을 생성할지 여부를 결정합니다.

☐ 자동 백업 보존


삭제 후 1일 동안 자동 백업을 보존할지 결정합니다.

☒ 인스턴스 삭제 시 시스템 스냅샷 및 특정 시점으로 복구를 포함한 자동화된 백업을 더 이상 사용할 수 없다는 점을 인정합니다.

실수로 삭제되는 것을 방지하기 위해 추가 서면 동의를 제공하세요.

삭제를 확인하려면 필드에 *delete me*을(를) 입력하세요.

delete me

 인스턴스를 삭제한 후에는 자동화된 백업을 더 이상 사용할 수 없기 때문에 인스턴스를 삭제하기 전에 최종 스냅샷을 만드는 것을 권장합니다.

취소

삭제

! 최종 스냅샷 생성과 자동 백업 보존을 체크하면 비용이 나간다. 따라서 실제 운영용 데이터베이스가 아니라면 체크를 해제하고 삭제를 하자.

 비용 나가지 않게 **ECR** 종료하기

