



# 아마존 EKS 업그레이드

유정열 / 당근마켓, AWS Hero

# 쿠버네티스



컨테이너화된 워크로드와 서비스를 자동적으로 배포, 스케일링 및 관리해주는 오픈소스 플랫폼

- 자동화된 롤아웃과 롤백
- 서비스 디스커버리와 로드 밸런싱
- 스토리지 오케스트레이션
- 자동화된 빈 패킹
- 자동화된 복구
- 시크릿과 구성 관리

# 아마존 EKS



- EKS는 k8s의 업스트림 및 인증된 k8s 버전
- EKS는 k8s 마이너 버전을 지원하여 고객이 업그레이드 할 수 있는 시간을 제공
- 성능, 신뢰성 및 보안 k8s를 위한 관리형 쿠버네티스 환경을 제공
- k8s의 실행, 운영 및 관리 작업을 쉽게 단순화

**아마존 EKS 를 통해 쉽고 안정적인 쿠버네티스 환경을 구축, 운영이 가능**

# 아마존 EKS 쿠버네테스 릴리즈 일정

Kubernetes 버전	업스트림 릴리즈	아마존 EKS 릴리즈	아마존 EKS 지원 종료
1.15	2019년 6월 19일	2020년 3월 10일	2021년 5월 3일
1.16	2019년 9월 8일	2020년 4월 30일	2021년 7월 25일
1.17	2019년 12월 9일	2020년 7월 10일	2021년 9월
1.18	2020년 3월 23일	2020년 10월 13일	2021년 11월
1.19 *	2020년 8월 26일	2021년 2월 16일	2022년 4월
1.20	2020년 12월 8일	2021년 5월 18일	2022년 6월

# 2021년 상반기 당근마켓 EKS 클러스터 상황

Stage	리전	Kubernetes 버전	목적
production	ap-northeast-2	1.15	운영, 배포, 모니터링
production	ap-northeast-2	1.15	서울 서비스
production	eu-west-2	1.15	영국 서비스
production	ca-central-1	1.15	북미 서비스
production	ap-northeast-1	1.15	일본 서비스
development	ap-northeast-2	1.15	운영, 배포
development	ap-northeast-2	1.15	서울 개발 테스트
development	eu-west-2	1.15	영국 개발 테스트
development	ca-central-1	1.15	북미 개발 테스트
development	ap-northeast-1	1.15	일본 개발 테스트

# 아마존 EKS 업그레이드

싱글 클러스터

멀티 클러스터



# 아마존 EKS 싱글 클러스터 업그레이드

- Master 노드

- EKS 마스터 업그레이드
- CoreDNS 업그레이드
- KubeProxy 업그레이드
- Amazon VPC CNI 플러그인 업그레이드

- Worker 노드

- 새 버전의 노드 그룹 생성
- Pod 이사

# 아마존 EKS 싱글 클러스터 업그레이드

- EKS 마스터 업그레이드

1.15 → 1.16 → 1.17 → 1.18



40분



40분



40분

# 아마존 EKS 싱글 클러스터 업그레이드



```
eksctl utils update-coredns --cluster=<cluster_name> --approve  
eksctl utils update-kube-proxy --cluster=<cluster_name> --approve  
eksctl utils update-aws-node --cluster=<cluster_name> --approve
```

<https://eksctl.io/usage/addon-upgrade/>

[https://docs.aws.amazon.com/ko\\_kr/eks/latest/userguide/update-cluster.html](https://docs.aws.amazon.com/ko_kr/eks/latest/userguide/update-cluster.html)

# 아마존 EKS 싱글 클러스터 업그레이드

- EKS 워커 노드 업그레이드

The screenshot shows the AWS EC2 Auto Scaling Groups page. The search bar contains "alpha-kr-b-workers". The table lists two auto scaling groups:

이름	시작 템플릿/구성	인스턴스	상태	원하는 용량
alpha-kr-b-workers-v1-20210613141255	alpha-kr-b-workers-v1-20210613141255	1	-	1
alpha-kr-b-workers-v2-20210613141255	alpha-kr-b-workers-v2-20210613141255	1	-	1

# 아마존 EKS 싱글 클러스터 업그레이드

- EKS 워커 노드 업그레이드

```
nalbam@ip-192-168-0-236:~
```

```
(* |alpha-kr-b:default)~
```

```
(* |alpha-kr-b:default)~
```

```
(* |alpha-kr-b:default)~
```

```
(* |alpha-kr-b:default)~ k get no --show-labels | grep workers
```

```
ip-10-128-214-249.ap-northeast-2.compute.internal Ready <none> 15m v1.18.9-eks-d1db3c beta.kubernetes.io/arch=amd64,beta.kubernetes.io/instance-type=m5.4xlarge,beta.kubernetes.io/os=linux.failure-domain.beta.kubernetes.io/region=ap-northeast-2,failure-domain.beta.kubernetes.io/zone=ap-northeast-2a group=workers,instancegroup=workers-v2,krmt.io/instancegroup=workers-v2,kubernetes.io/arch=amd64,kubernetes.io/hostname=ip-10-128-214-249.ap-northeast-2.compute.internal,kubernetes.io/os=linux,node.kubernetes.io/instance-type=m5.4xlarge,subgroup=v2,topology.kubernetes.io/region=ap-northeast-2,topology.kubernetes.io/zone=ap-northeast-2a
```

```
ip-10-128-219-42.ap-northeast-2.compute.internal Ready,SchedulingDisabled <none> 14m v1.18.8-eks-7c9bda beta.kubernetes.io/arch=amd64,beta.kubernetes.io/instance-type=m5.4xlarge,beta.kubernetes.io/os=linux.failure-domain.beta.kubernetes.io/region=ap-northeast-2,failure-domain.beta.kubernetes.io/zone=ap-northeast-2c group=workers,instancegroup=workers-v1,krmt.io/instancegroup=workers-v1,kubernetes.io/arch=amd64,kubernetes.io/hostname=ip-10-128-219-42.ap-northeast-2.compute.internal,kubernetes.io/os=linux,node.kubernetes.io/instance-type=m5.4xlarge,subgroup=v1,topology.kubernetes.io/region=ap-northeast-2,topology.kubernetes.io/zone=ap-northeast-2c
```

```
(* |alpha-kr-b:default)~
```

```
(* |alpha-kr-b:default)~
```

```
(* |alpha-kr-b:default)~
```

# 아마존 EKS 싱글 클러스터 업그레이드

- EKS 워커 노드 업그레이드

```
nalbam@ip-192-168-0-236:~                                     ▾⌘1
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ k get no -l group=workers
NAME                      STATUS   ROLES     AGE      VERSION
ip-10-128-214-249.ap-northeast-2.compute.internal Ready    <none>   10m     v1.18.9-eks-d1db3c
ip-10-128-219-42.ap-northeast-2.compute.internal  Ready    <none>   9m14s   v1.18.8-eks-7c9bda
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ k get no -l instancegroup=workers-v1
NAME                      STATUS   ROLES     AGE      VERSION
ip-10-128-219-42.ap-northeast-2.compute.internal  Ready    <none>   9m23s   v1.18.8-eks-7c9bda
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ k get no -l instancegroup=workers-v1 | grep -v 'NAME' | awk '{print $1}'
ip-10-128-219-42.ap-northeast-2.compute.internal
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~
□ ~
```

# 아마존 EKS 싱글 클러스터 업그레이드

- EKS 워커 노드 업그레이드

```
nalbam@ip-192-168-0-236:~
```

```
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~ k get no -l instancegroup=workers-v1 | grep -v 'NAME' | awk '{print $1}' | xargs -I {} kubectl cordon {}
node/ip-10-128-219-42.ap-northeast-2.compute.internal cordoned
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~ k get no -l group=workers
NAME                      STATUS          ROLES   AGE    VERSION
ip-10-128-214-249.ap-northeast-2.compute.internal Ready      <none>  13m   v1.18.9-eks-d1db3c
ip-10-128-219-42.ap-northeast-2.compute.internal Ready  SchedulingDisabled  <none>  12m   v1.18.8-eks-7c9bda
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~
(* |alpha-kr-b:default)~
```

# 아마존 EKS 싱글 클러스터 업그레이드

- EKS 워커 노드 업그레이드

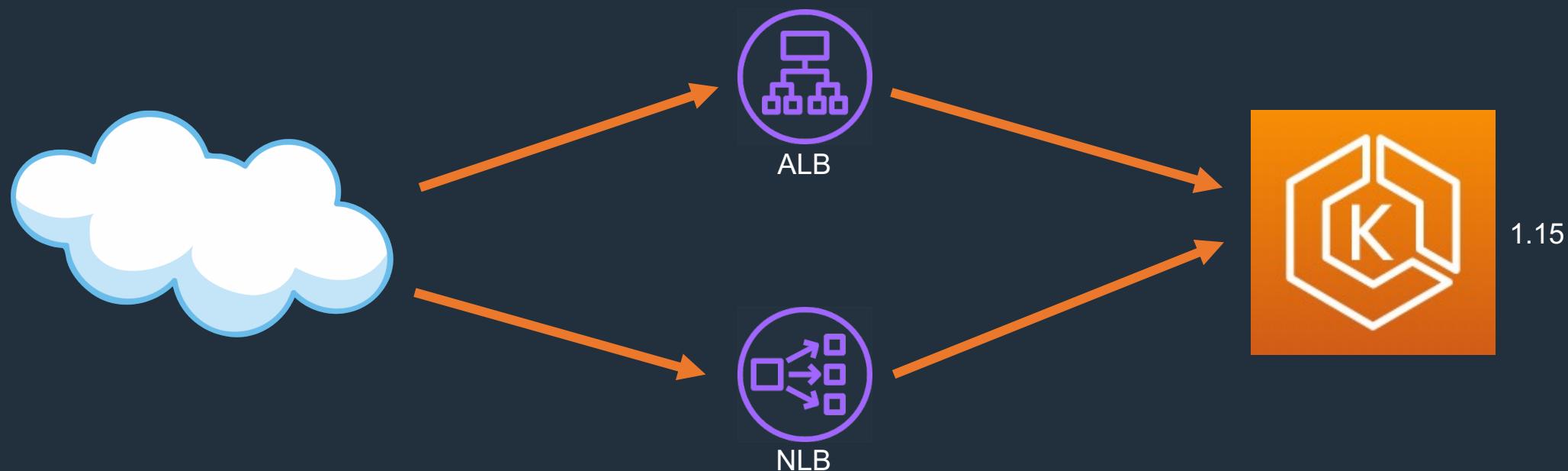
```
nalbam@ip-192-168-0-236:~
```

```
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~ k get no -l instancegroup=workers-v1 | grep -v 'NAME' | awk '{print $1}' | \
xargs -I {} kubectl drain --delete-emptydir-data --ignore-daemonsets --skip-wait-for-delete-timeout=0 {}
node/ip-10-128-219-42.ap-northeast-2.compute.internal already cordoned
WARNING: ignoring DaemonSet-managed Pods: addon-aws-efs-csi-driver/efs-csi-node-trn6m, addon-logging/promtail-zxsfn, addon-monitoring/node
-exporter-mmjml, fluentd/forwarder-z5pzl, kube-system/aws-node-nqnjp, kube-system/kube-proxy-k62d8
node/ip-10-128-219-42.ap-northeast-2.compute.internal drained
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~
(* |alpha-kr-b:default)~ ~
```

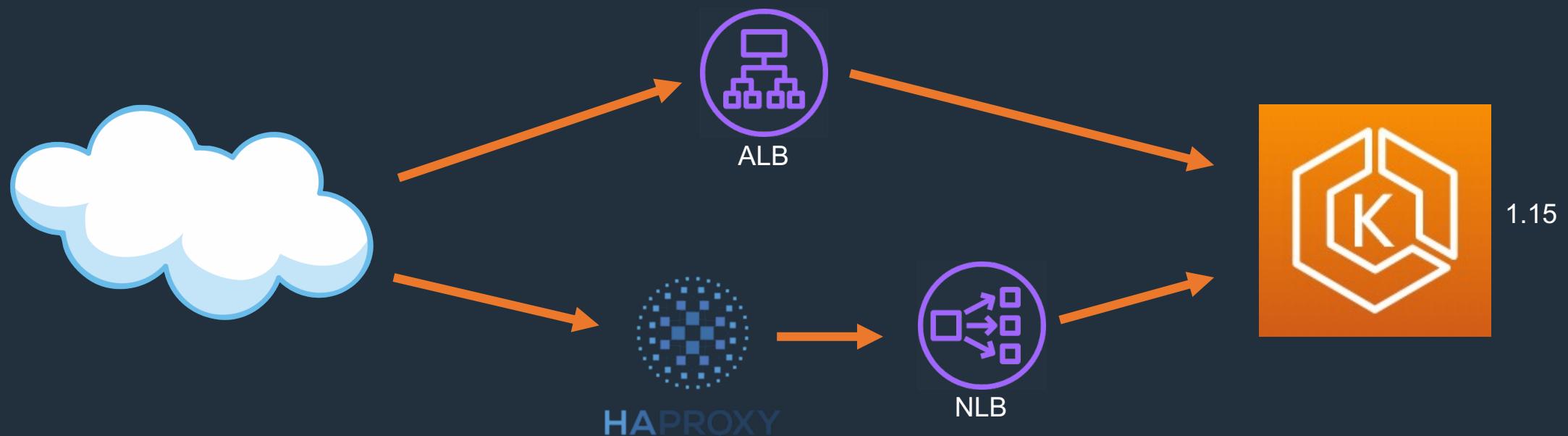
# 아마존 EKS 싱글 클러스터 업그레이드

- 버전을 한번에 하나씩 업그레이드 가능
- 마스터 노드 업그레이드에서 시간을 상당히 소요
- 잘 동작하고 있는 기존 시스템을 크게 변경 하지 않음
- 노드그룹 과 cordon, drain 으로 무중단 업그레이드 가능
- 한번 해보면 생각보다 어렵지는 않음
- 롤백은 불가능

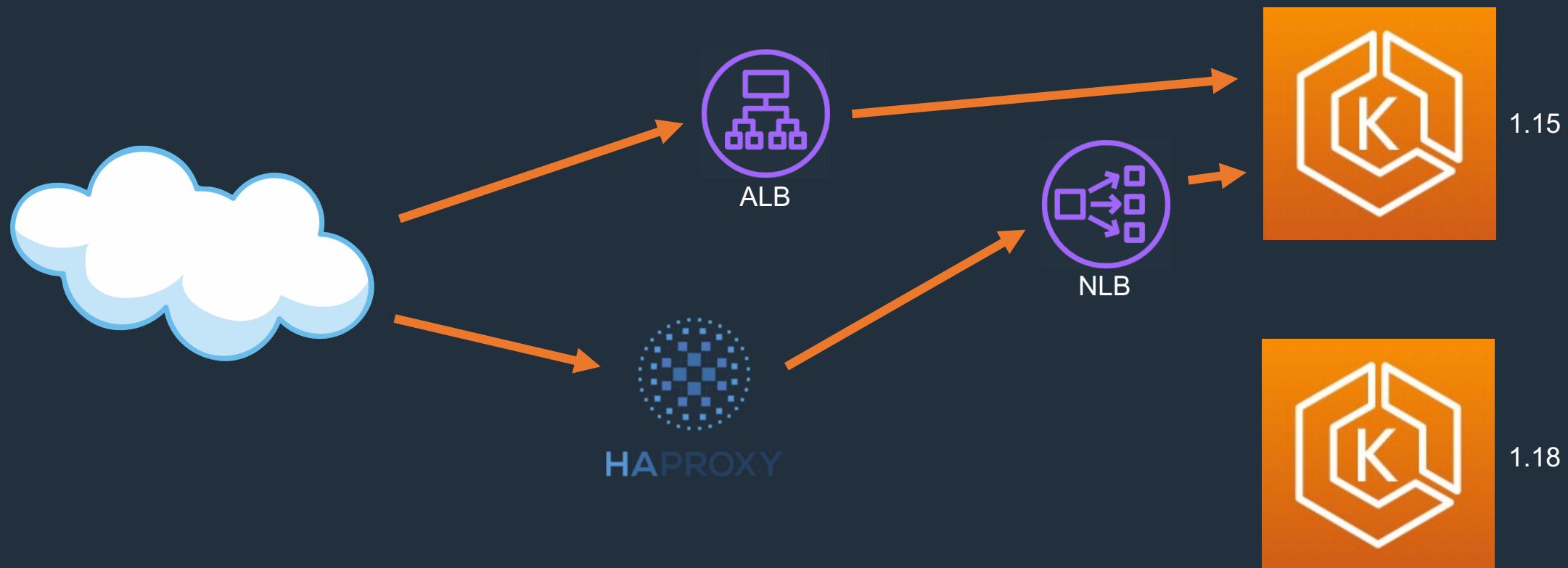
# 아마존 EKS 멀티 클러스터 업그레이드



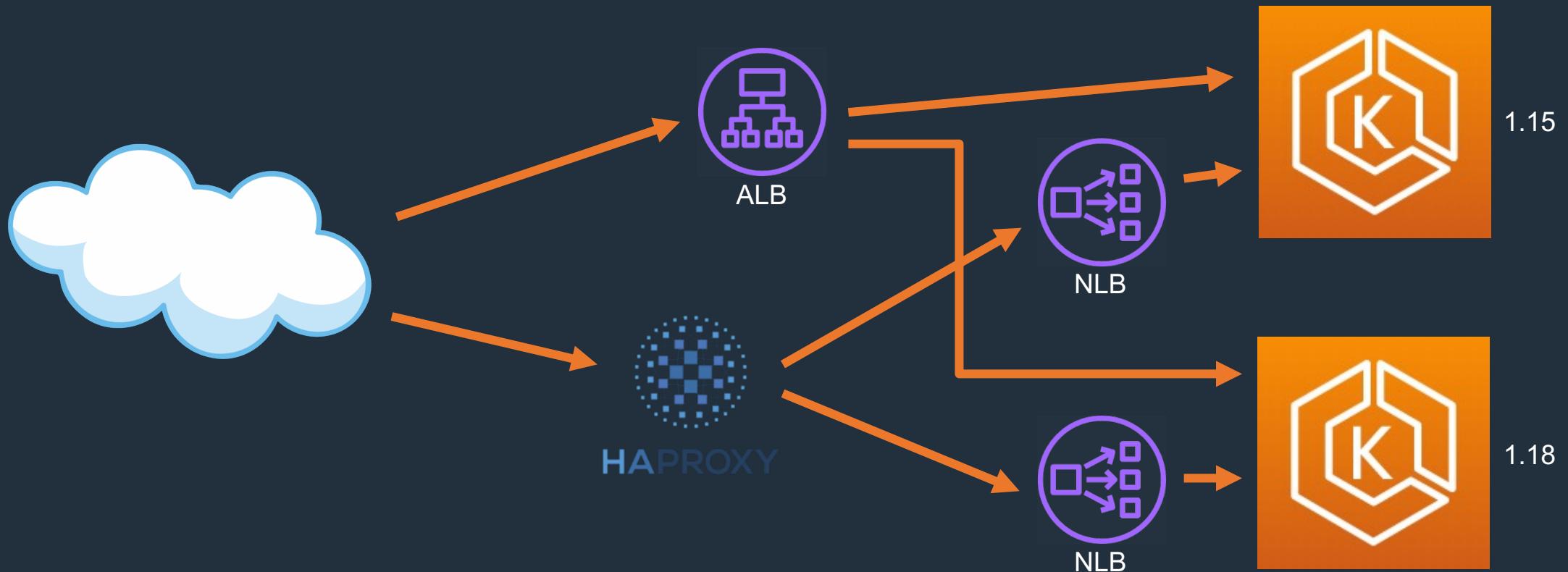
# 아마존 EKS 멀티 클러스터 업그레이드



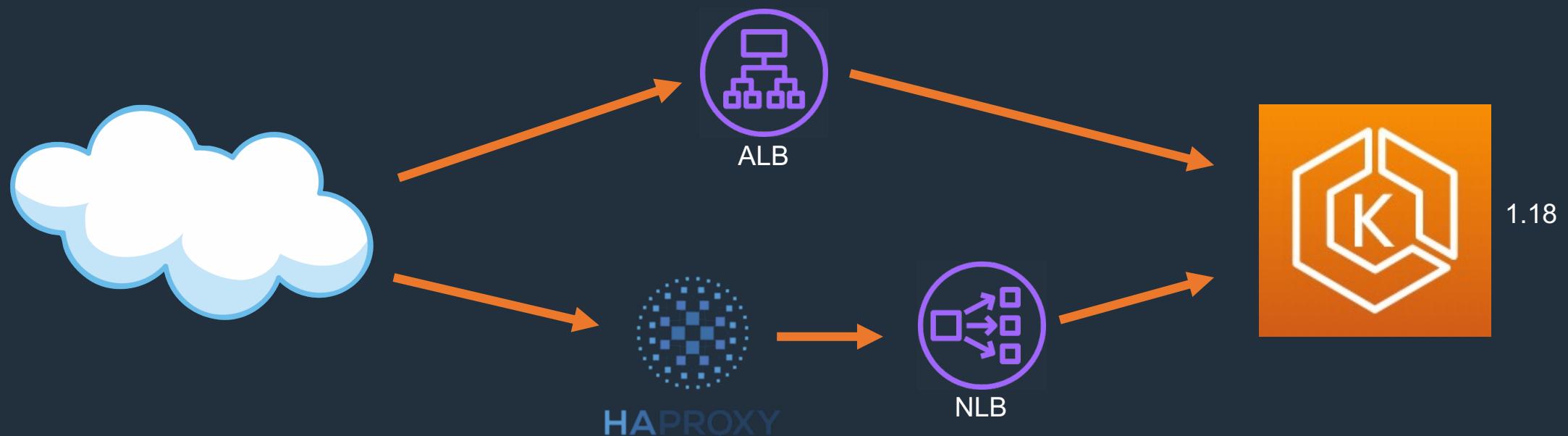
# 아마존 EKS 멀티 클러스터 업그레이드



# 아마존 EKS 멀티 클러스터 업그레이드



# 아마존 EKS 멀티 클러스터 업그레이드



# 아마존 EKS 멀티 클러스터 업그레이드

- HAProxy 트래픽 전환



# 아마존 EKS 멀티 클러스터 업그레이드

- ALB 트래픽 전환

Load Balancer 생성 작업

이름	DNS 이름	상태	VPC ID	가용 영역	유형
alpha-webapp	alpha-webapp-1043183907.a...	활성	vpc-395e2151	ap-northeast-2a, ap-nor...	application

로드 밸런서: alpha-webapp

설명 리스너 모니터링 통합 서비스 태그

리스너는 프로토콜 및 포트를 사용하여 연결 요청을 수신합니다. 리스너 및 리스너 규칙을 추가, 제거 또는 업데이트할 수 있습니다.

리스너 속성을 보고 편집하려면 리스너를 선택하고 [편집]을 선택합니다.

리스너 추가 편집 삭제

리스너 ID	보안 정책	SSL 인증서	규칙
HTTP : 80	해당 사항 없음	해당 사항 없음	기본값: 다음으로 전달 중: 가중치가 적용된 여러 대상 그룹 <a href="#">규칙 보기/편집</a>
arn...e6ffc96f1e87f8f8			
HTTPS : 443	ELBSecurityPolicy-2016-08	기본값: 10652655-c378-4d5b-b1a3-e0ad55cdd027 (ACM) <a href="#">인증서 보기/편집</a>	기본값: 다음으로 전달 중: 가중치가 적용된 여러 대상 그룹 <a href="#">규칙 보기/편집</a>
arn...bd34b562e968454e			

# 아마존 EKS 멀티 클러스터 업그레이드

- ALB 트래픽 전환

The screenshot shows the AWS Lambda function configuration interface with three stages listed:

- Stage 15 (ARN: ARN...c0ca7):**
  - IF:** 경로는 /api\* 또는 /ad\* 또는 /admin\* 또는 /api\_internal\* 또는 /vv/\*
  - THEN:** 다음으로 전달:
    - alpha-webapp-b: 100 (100%)
    - alpha-webapp-a: 0
    - alpha-webapp-default: 0그룹 수준 고정성: 끔
- Stage 16 (ARN: ARN...c00f1):**
  - IF:** 경로는 /health\_check
  - THEN:** 다음으로 전달:
    - alpha-webapp-b: 100 (100%)
    - alpha-webapp-a: 0
    - alpha-webapp-default: 0그룹 수준 고정성: 끈
- 마지막 (Stage HTTP 80: 기본 작업):**
  - IF:** 요청 시에만 라우팅
  - THEN:** 다음으로 전달:
    - alpha-webapp-b: 100 (100%)
    - alpha-webapp-a: 0
    - alpha-webapp-default: 0그룹 수준 고정성: 끈

# 아마존 EKS 멀티 클러스터 업그레이드

- ALB 트래픽 전환

The screenshot shows the AWS EC2 Target groups console. At the top, there is a search bar with the ARN of the target group: `arn:aws:elasticloadbalancing:ap-northeast-2:...:targetgroup/alpha-webapp-b/b4f5bd046b85564d`. Below the search bar, a table lists the target group details:

Name	ARN	Port	Protocol	Target type	Load balancer	VPC ID
alpha-webapp-b	arn:aws:elasticloadbalancin...	80	HTTP	IP	alpha-webapp	vpc-395e2151

Below the table, the target group configuration for `alpha-webapp-b` is shown. The `Targets` tab is selected. It displays one registered target:

IP address	Port	Zone	Health status	Health status details
10.128.214.131	80	ap-northeast-2a	healthy	(green circle icon)

# 아마존 EKS 멀티 클러스터 업그레이드

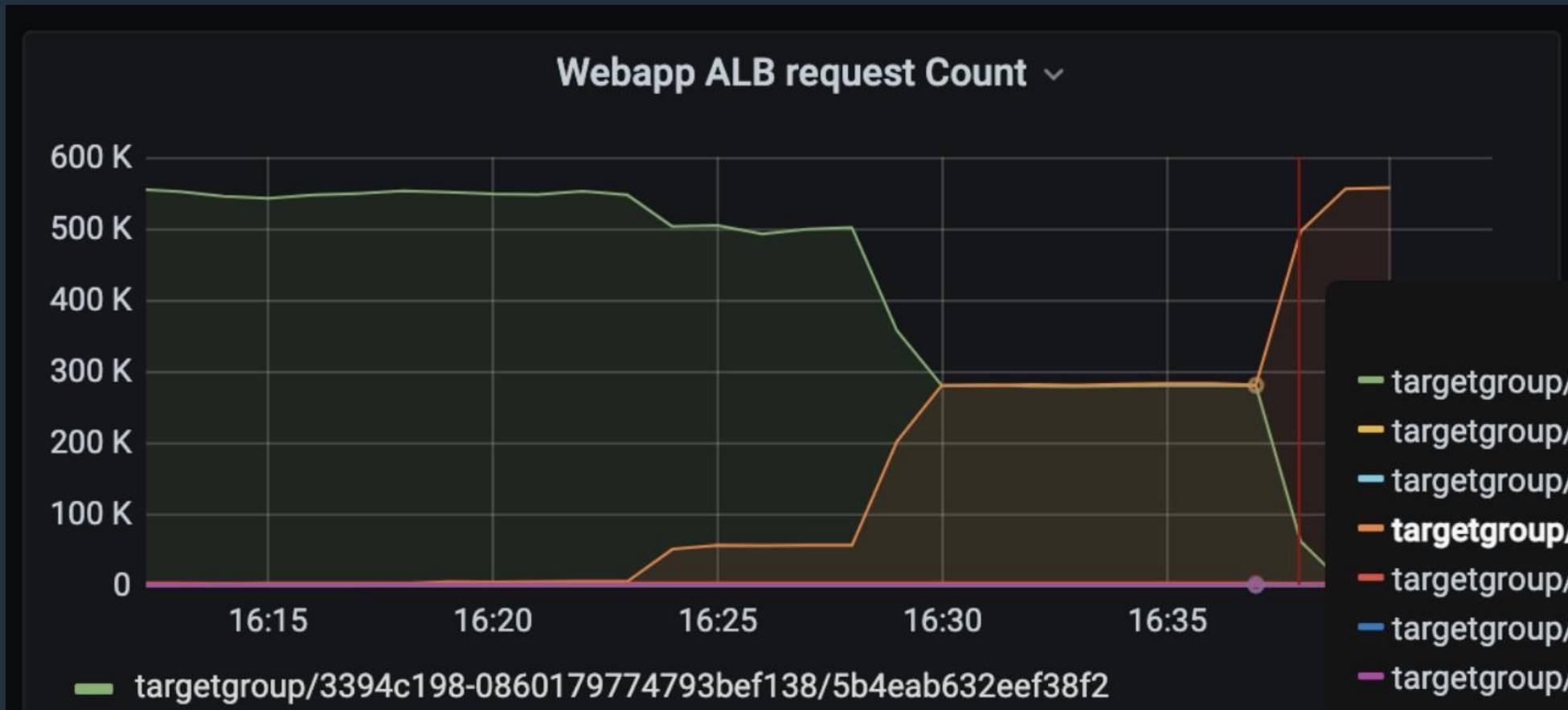
- ALB 트래픽 전환

LIVE MANIFEST DIFF DESIRED MANIFEST

```
1 apiVersion: elbv2.k8s.aws/v1beta1
2 kind: TargetGroupBinding
3 metadata:
4   labels:
5     app: raw
6     app.kubernetes.io/instance: aws-load-balancer-tgb-webapp-alpha-kr-b
7     chart: raw-0.2.5
8     heritage: Helm
9     release: aws-load-balancer-tgb-webapp-alpha-kr-b
10    name: webapp
11    namespace: webapp
12  spec:
13    serviceRef:
14      name: webapp
15      port: http
16    targetGroupARN: >-
17      arn:aws:elasticloadbalancing:ap-northeast-2: :targetgroup/alpha-webapp-b/b4f5bd046b85564d
18    targetType: ip
19
```

# 아마존 EKS 멀티 클러스터 업그레이드

- ALB 트래픽 전환



# 아마존 EKS 멀티 클러스터 업그레이드

- 버전을 여러 단계 건너뛰어 업그레이드 가능
- 기존의 내부 시스템 버전도 같이 업그레이드 가능
  - Istio 1.5 -> 1.8
- 앞단에 트래픽 조절을 위한 시스템 필수 (ALB, HAProxy 등)
- 트래픽 조절만 잘하면 무중단 업그레이드 가능
- 트래픽 조절로 상시 멀티 클러스터 운영도 가능
  - A-zone 클러스터, C-zone 클러스터
- Pod 는 동시 실행 가능하도록 변경 필요 (Job 등)
- 언제든 이전 상태로 롤백이 가능
- 새 술은 새 그릇에

# 아마존 EKS 쿠버네테스 릴리즈 일정

Kubernetes 버전	업스트림 릴리즈	아마존 EKS 릴리즈	아마존 EKS 지원 종료
1.15	2019년 6월 19일	2020년 3월 10일	2021년 5월 3일
1.16	2019년 9월 8일	2020년 4월 30일	2021년 7월 25일
1.17	2019년 12월 9일	2020년 7월 10일	2021년 9월
<b>1.18 ←</b>	<b>2020년 3월 23일</b>	<b>2020년 10월 13일</b>	<b>2021년 11월</b>
1.19 *	2020년 8월 26일	2021년 2월 16일	2022년 4월
1.20	2020년 12월 8일	2021년 5월 18일	2022년 6월

# 사람을 찾아요.



AWS

IaC

Terraform

DevOps

Kubernetes

GitOps

Prometheus

SRE

AWSKRUG #container @nalbam



감사합니다.