8 Kubernetes 보안 강화 방법

09 cert-manager를 활용한 TLS 인증서 관리

실습 내용

순서

1. 사전 준비사항
2. Ingress-Nginx 설치 (Helm Chart)
3. cert-manager를 통한 TLS 인증서 발급
4. 테스트용 K8s Object 배포 및 실행
5. Route53 도메인 등록 및 HTTPS 접속 검증

실습 예제코드 경로

Chapter08 > Ch08_09-cert-manager-test

1. 사전 준비사항

(1) EKS Cluster

(2) Domain 주소

- Route53 > 등록된 도메인 > 도메인 등록에서 도메인 준비 (신청후 이메일 인증 1회)
- Domain이 발급 되었다면 다음의 경로에서 도메인을 확인해 사용가능
  - Route53 > 호스팅 영역 > 본인이 등록한 도메인 이름 클릭

2. Ingress-Nginx 설치 (Helm Chart)

(1) Helm Chart를 이용한 Ingress-Nginx 설치

- Chapter08 > Ch08_09-cert-manager-test > ingress-nginx

```
$ helm install ingress-nginx ./
```

(2) Service - LoadBalancer Type을 통한 자동 ELB 생성 확인

```
$ kubectl get service
```

2. cert-manager를 통한 TLS 인증서 발급

(1) TLS 인증서 자동 발급을 위한 ClusterIssuer 배포 명령어

- Chapter08 > Ch08_09-cert-manager-test > cert-manager

```
$ kubectl apply -f cluster-issuer.yaml
```

(2) ClusterIssuer 배포 및 실행 결과 확인 명령어

```
$ kubectl get clusterissuers
```

3. 테스트용 K8s Object 배포 및 실행

(1) 테스트용 Deployment, Service, Ingress 설치 명령어

- Chapter08 > Ch08_09-cert-manager-test > k8s-manifests

```
$ kubectl apply -f ./
```

(2) 테스트용 K8s Object 배포 및 실행 결과 확인 명령어

```
$ kubectl get all
```

```
$ kubectl get ingress
```

4. Route53 도메인 등록 및 HTTPS 접속 검증 #1

(1) AWS Management Console에서 Route53 서비스 접속

- 상단 검색창 > Route53 > **호스팅 영역** > 본인이 생성한 도메인 이름 클릭

(2) Route53 도메인 등록

- 레코드 생성 > 레코드 이름 입력 > 레코드 유형 - **CNAME** 선택

  > 값 - Service **External-IP값(ELB DNS주소)** 복사, 붙여넣기

  > **레코드 생성** 버튼 클릭 > 도메인 등록 확인

4. Route53 도메인 등록 및 HTTPS 접속 검증 #2

(3) 등록된 Domain 주소를 웹브라우저에 입력

- 페이지가 안 나올 경우, Domain 주소 앞에 **https://**를 붙임

(4) HTTPS 접속 검증

1. 웹브라우저 주소창 맨앞에 **자물쇠** 모양 클릭
2. "**이 사이트는 보안 연결(HTTPS)이 사용되었습니다.**" 항목 클릭
3. "**인증서가 유효함**" 항목 클릭