

9 Kubernetes 트러블 슈팅 방법

06 DNS 로깅 및 이슈 사례 조치방법

실습 내용

06. DNS 로깅 및 이슈 사례 조치방법

순서

1. DNS 로깅 방법 소개
2. DNS 이슈 사례 소개
3. CoreDNS 튜닝 설정
4. DNS 로깅 방법 실습
5. DNS 이슈 사례 발생시 확인 방법 실습

1. DNS 로깅 방법 소개

구분	상세 설명
CoreDNS Log	<ul style="list-style-type: none"> • Kubernetes(EKS) 클러스터 내 모든 DNS 쿼리를 로깅 • DNS 쿼리를 통한 에러나 성능 저하요소 확인
nslookup	<ul style="list-style-type: none"> • 기본적인 DNS 쿼리 툴 • 간단한 명령과 옵션으로 각종 DNS 쿼리 가능
dig	<ul style="list-style-type: none"> • 전문적인 DNS 쿼리 툴 • 다양한 명령과 옵션으로 고급 DNS 쿼리 가능
Route53 Log	<ul style="list-style-type: none"> • AWS Route53에 등록된 DNS 쿼리 로깅 • DNS 쿼리를 수행한 Client의 정보 확인 가능

2. DNS 이슈 사례 소개

구분	상세 설명
DNS 쿼리 이슈	<ul style="list-style-type: none"> • POD의 DNS 쿼리가 안될 시 <ul style="list-style-type: none"> -> DNS 디버깅용 임시 POD 배포(nslookup, dig 명령어 수행 가능) -> POD 내 Shell 내부에서 /etc/resolv.conf 설정
DNS 성능 이슈	<ul style="list-style-type: none"> • DNS 쿼리 과부하로 성능 이슈가 발생할 때 <ul style="list-style-type: none"> -> Deployment에 dnsConfig 설정 -> ndot 적용 및 DNS 도메인 FQDN화
CoreDNS 튜닝	<ul style="list-style-type: none"> • CoreDNS 설정 변경으로 성능 및 가용성을 향상하고자 할 때 <ul style="list-style-type: none"> -> cache 시간 변경 (TTL 시간 증가) -> AWS Route53 직접 연동

3. CoreDNS 튜닝 설정

06.
DNS 로깅 및 이슈
사례 조치방법

\$ **kubectl edit cm coredns -n kube-system**

```
data:
  Corefile: |
    fastcampus.click { # DNS 도메인명 추가
      route53 fastcampus.click.:Z09958083P088Y5E7ZP8H # Route53 Hosted Zone ID 추가
    }
    .:53 {
      errors
      health
      kubernetes cluster.local in-addr.arpa ip6.arpa {
        pods insecure
        fallthrough in-addr.arpa ip6.arpa
      }
      prometheus :9153
      forward . /etc/resolv.conf
      cache 60 # 30 -> 60 변경
      loop
      reload
      loadbalance
      log # 추가
    }
```

4. DNS 로깅 방법 실습 #1

(1) CoreDNS 로깅 활성화

- 3번에 명시된 **ConfigMap** 내에서 "**log**" 추가

(2) nslookup 명령어

- 기본 쿼리

\$ **nslookup** <DNS 도메인명>

- CNAME 쿼리

\$ **nslookup -q=cname** <DNS 도메인명>

- 특정 DNS 서버 적용한 쿼리

\$ **nslookup** <DNS 도메인명> <특정 DNS 서버 IP>

4. DNS 로깅 방법 실습 #2

(3) dig 명령어

- 기본 쿼리

\$ **dig** <DNS 도메인명>

- 요약 쿼리

\$ **dig** +short <DNS 도메인명>

- CNAME 쿼리

\$ **dig** cname <DNS 도메인명>

- 특정 DNS 서버 적용한 쿼리

\$ **dig** @<특정 DNS 서버 IP> <DNS 도메인명>

- DNS 쿼리 추적

\$ **dig** +trace <DNS 도메인명>

4. DNS 로깅 방법 실습 #3

06.
DNS 로깅 및 이슈
사례 조치방법

(4) Route53 로깅 (글로벌 서비스는 us-east1 리전 선택)

- AWS CloudWatch > 로그 > 로그 그룹 > **/aws/route53/<지정한 로그 그룹명>**

5. DNS 이슈 사례 발생시 확인 방법 실습 #1 - DNS 쿼리 이슈

(1) DNS 디버깅용 임시 POD 배포(nslookup, dig 명령어 수행 가능)

```
$ kubectl apply -f https://k8s.io/examples/admin/dns/dnsutils.yaml
```

(2) POD 내 Shell 내부에서 /etc/resolv.conf 설정

```
$ kubectl exec -it <대상 POD명> - sh
```

```
$ vi /etc/resolv.conf 설정 및 $ nslookup kubernetes 10.100.0.10 쿼리 수행
```

```
nameserver 10.100.0.10
```

```
search default.svc.cluster.local svc.cluster.local cluster.local ec2.internal
```

```
options ndots:5
```

5. DNS 이슈 사례 발생시 확인 방법 실습 #2 - DNS 성능 이슈

(3) Deployment에 dnsConfig 설정

- Manifest 수정 혹은 kubectl edit (spec.template.spec.dnsConfig 부분)

```
dnsPolicy: ClusterFirst
dnsConfig:
  options:
    - name: ndots
      value: "1"
```

(4) ndot 적용 및 DNS 도메인 FQDN화

- ndot:5가 기본값이라 모든 DNS 도메인은 최대 5번까지 쿼리 수행
- 특정 도메인명 맨끝에 '.'(Dot)을 추가하면 해당 도메인은 FQDN으로 인식해 1번만 쿼리

5. DNS 이슈 사례 발생시 확인 방법 실습 #3 - CoreDNS 튜닝

06.
DNS 로깅 및 이슈
사례 조치방법

\$ **kubectl edit cm coredns -n kube-system**

```
data:
  Corefile: |
    fastcampus.click { # DNS 도메인명 추가
      route53 fastcampus.click.:Z09958083P088Y5E7ZP8H # Route53 Hosted Zone ID 추가
    }
    .:53 {
      errors
      health
      kubernetes cluster.local in-addr.arpa ip6.arpa {
        pods insecure
        fallthrough in-addr.arpa ip6.arpa
      }
      prometheus :9153
      forward . /etc/resolv.conf
      cache 60 # 30 -> 60 변경
      loop
      reload
      loadbalance
      log # 추가
    }
```