

재난안전 시스템의 고가용성 IoT 서비스를 위한 컨테이너 오케스트레이션 기반 개발

박성철, 이성호, 박정환, 박찬, 나호성

한전KDN

sungchul_0126@kdn.com, letitbe_953387@kdn.com, sypapa_14@kdn.com, chan-767710@kdn.com, sung_970529@kdn.com

DEVELOPMENT OF CONTAINER ORCHESTRATION-BASED HIGH AVAILABILITY IoT SERVICE FOR DISASTER SAFETY SYSTEM

Park Sung Chul, Lee Sung Ho, Park Jung Hwan, Park Chan, Na Ho Sung

KEPCO KDN

요약

공동구 내부의 재난확산 방지와 대응 전략 수립에 전력케이블의 부분방전 감시는 매우 중요한 요소이다. 부분방전은 화재 및 폭발에 의한 시발점이 될 수 있는 복합재난의 중요 요소로 전력케이블의 지속적인 감시가 필요하며 공동구 내부 전력케이블의 진단 기술개발을 통한 케이블 감시 기술 고도화 및 모니터링이 필요한 상황이다. 공동구는 사회 필수 기반을 수용하는 기반 시설로 도시발전에 따라 지속해서 증가하고 있으며 이에 따라 감시장치를 수용해야 하는 서버는 필연적으로 고가용성을 확보해야 한다.

1. 서론

공동구에 있는 전력 설비의 고장진단/원인분석을 하는 것은 전력 계통 운영에 필요한 최적의 설비투자 시기를 설정하기 위해 중요한 요소이다. 도심화가 지속되면서 생활 기반을 수용하는 공동구는 계속 확장되고 있으며 이에 따른 센서 등의 장치가 수없이 많은 상태이고 계속 증가하고 있다. 또한 유지보수 및 전문인력 투입을 통한 진단은 막대한 기간과 비용 투자가 필요하므로 설비 상태를 원격으로 감시/진단하기 위해 다양한 시스템이 도입되어 운영되고 있다.

배전 전력 설비의 상태감시 및 설비 고장 진단을 위한 시스템이 개발 중으로 부분방전 발생 여부 판단은 고장의 원인이 되는 고장 부분 식별을 위해 필요한 기능이다. 전력케이블에 설치된 센서 및 통신장치를 통해 센서의 계측 데이터를 서버로 송신 후 서버는 부분방전 데이터로 유형 또는 고장 여부를 분류하는 구성이다. 그러나 부분방전을 분류하기 위한 다양한 방법이 있고, 그에 따른 다양한 데이터들을 대량으로 수집, 분석해야 하는 서버는 부하가 발생한다. 사회의 기반 인프라를 안정적으로 감시하고 운용하기 위해서는 서버 또한 그에 준하는 고가용성 기능을 갖춰야 한다.

2. 본론

2.1 부분방전 데이터 수집/분석 특성

전력 설비 진단은 진단 부분의 특성을 고려하여 진단 방법을 선택해야 하며 케이블 및 접속재, 지상기 등 공동적으로 온도와 부분방전 여부가 고장 변수로서 감시 대상이다. 이 중 부분방전은 국부적으로 발생하는 전기적 방전으로 공기 갭, 내부 또는 절연체 표면 결함에 의해 발생하는 것으로 전력설비의 이상 여부를 판단하는 기준이 되는 중요한 요인이다. 하지만 부분방전이 발생했다고 해서 고장으로 판정하는 것은 아니고 고장의 징조이기 때문에 지속 감시해야 한다. 장시간 운영할 수 있는 안정적인 시스템의 구성을 고려해야 한다.

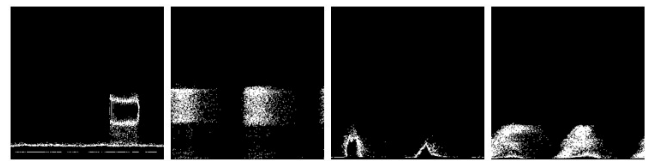


그림 1 부분방전 패턴 이미지

부분방전은 계측된 변수를 통한 판단, PRPD 이미지 패턴을 이용한 판단, 펄스 신호를 분석하는 판단 등 다양한 방법이 존재하며 다양한 데이터를 장기관점에서 수집, 분석, 운영이 가능해야 하며 많은 장치가 추가되더라도 서버의 물리적 확장을 통해 클러스터 구성이 용이하다면 이미지 데이터를 시계열 모델을 활용하여 추가적인 기능개발이 가능하게 된다.

2.2 공동구 환경 구성

공동구 부분방전 진단시스템은 공동구 내 접속점을 대상으로 데이터수집·전송 장치를 설치하고 일정 시간 계측 후 데이터를 서버로 전달한다.

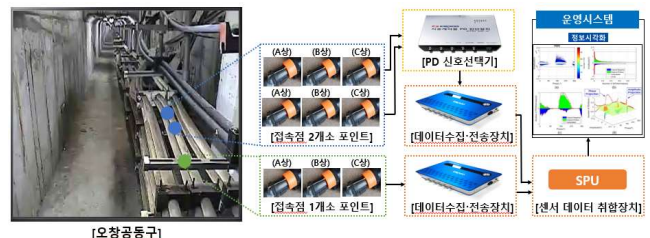


그림 2 설치 환경 및 시스템 구성도

데이터수집·전송 장치는 노이즈와 부분방전신호를 분류 후 운영시스템에 일정 시간마다 PRPD 패턴, 분석된 변수 등의 데이터를 전송하고 운영시스템은 데이터 유형에 따라 분리하여 적재시킨다.

2.3 부분방진 진단시스템 아키텍처

컨테이너 오케스트레이션 툴로 쿠버네티스를 도입하였다. 많은 장점 중 특히 중요했던 부분은 하드웨어 클러스터 구성 및 활용도 증가, 이에 따른 애플리케이션 배포 단순화, 상태 확인과 자가 치유, 오토스케일링 등이 있다.

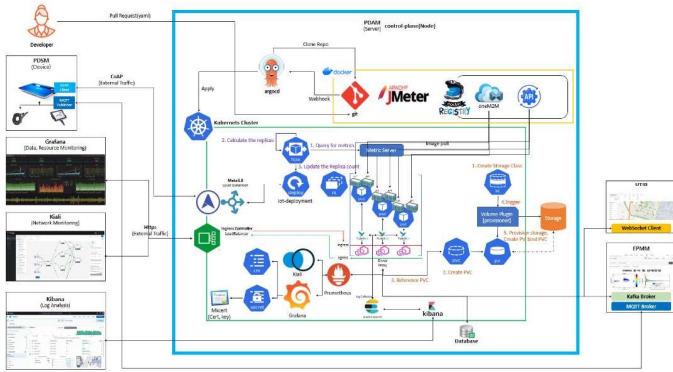


그림 3 쿠버네티스 기반 리소스 구성

2.3.1 기능별 환경 구성

폐쇄망 환경에서 동작시키기 위해 Local Registry를 구성하여 필요한 이미지들을 미리 배포하였고, 과제 내에서 개발한 IoT, API 서버 등을 Registry에 배포하였다. 외부 데이터 수신 및 로드밸런싱을 위해 Metallb LoadBalancer를 구성하였으며 파드 동작에 필요한 이미지들을 배포하였다.

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED NODE	READINESS GATES
pod/k8s-lot-dep-7855c955-4gc76	3/3	Running	12 (5d22h ago)	17d			<none>	<none>
pod/k8s-lot-dep-7855c955-7j2bq	3/3	Running	12 (5d22h ago)	17d			<none>	<none>

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
service/k8s-lot-loadbalancer-eno8403	LoadBalancer	ClusterIP	443/TCP	

NAME	READY	UP-TO-DATE	AVAILABLE	AGE	CONTAINERS	IMAGES
Deployment.apps/k8s-lot-dep	2/2	2	17d	17d	k8s-lot-container_k8s-api-container	localhost:5000/server-lot:1.4,localhost:5000/server-lot:1.4

NAME	REFERENCE	TARGETS	HMPPODS	MAXPODS	REPLICAS	AGE
horizontalpodautoscaler.autoscaling/k8s-lot-dep	Deployment.apps/k8s-lot-dep	0%/5%	2	5	2	21d

그림 4 쿠버네티스 기본 리소스

Istio를 도입하여 사이트카 프록시 배포하였다. 서비스 간 네트워크 통신을 가로채는 Mesh 네트워크 구성을 통해 유연한 서비스 구성을 가능하게 하였고 Prometheus, Kiali, Grafana등의 Addon 서비스를 추가로 구성하여 네트워크 및 장치 자원(CPU, Memory) 등의 모니터링을 가능하게 구성하였다.

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
ingress-nginx	pod/ingress-nginx-controller-778d4c6454-tmt9l	1/1	Running	4 (5d22h ago)	19d
istio-system	pod/grafana-69d7c557dd-lmcws	1/1	Running	4 (5d22h ago)	21d
istio-system	pod/istio-egressgateway-7653784c5-rdr2w	1/1	Running	4 (5d22h ago)	21d
istio-system	pod/istio-ingressgateway-869d377659-68lmj	1/1	Running	4 (5d22h ago)	21d
istio-system	pod/istiod-5c4f4498d-zg9ql	1/1	Running	4 (5d22h ago)	21d
istio-system	pod/kiali-97c9b4d6-lrrt9	1/1	Running	4 (5d22h ago)	15d
istio-system	pod/prometheus-5d5d6d6f-c-b8bl	2/2	Running	8 (5d22h ago)	17d

NAMESPACE	NAME	TYPE	CLUSTER-IP	EXTERNAL-IP
service/grafana-external	ExternalName	grafana.istio-system.svc.cluster.local	<none>	<none>
ingress-nginx	service/ingress-nginx-controller	LoadBalancer	<none>	<none>
ingress-nginx	service/ingress-nginx-controller-admission	ClusterIP	<none>	<none>
ingress-nginx	service/kiali-external	ExternalName	kiali.istio-system.svc.cluster.local	<none>
istio-system	service/grafana	ClusterIP	<none>	<none>
istio-system	service/istio-egressgateway	ClusterIP	<none>	<none>
istio-system	service/istio-ingressgateway	LoadBalancer	<none>	<none>
istio-system	service/istiod	ClusterIP	<none>	<none>
istio-system	service/kiali	ClusterIP	<none>	<none>
istio-system	service/prometheus	ClusterIP	<none>	<none>

그림 5 Istio Addon 서비스 구성

컨테이너의 동작상태를 초기 실행 시 ReadinessProbe를 이용해 체크하였으며 주기적으로 동작 가능 여부 확인을 위해 LivenessProbe를 이용해 파드내부의 컨테이너 동작을 체크하여 오류발생 시 재실행 할 수 있도록 구성하였다.

IoT 장치 수량 증가 등의 이유로 분석요청 데이터가 증가할 경우를 고려해 HPA(Horizontal Pod AutoScaler)리소스를 통해 CPU, Memory 등의 임계치를 설정하여 동적으로 스케일업/다운이 가능하도록 구성하였다.

컨테이너 특성으로 오류 등의 종류 내부 데이터를 유실하게 되는데 이를 해결하기 위해 Log 데이터를 보관/분석할 수 있는 EFK

stack(ElasticSearch, FluentBit, Kibana)을 구성하여 Log 분석을 수행할 수 있는 환경으로 구성하였으며 이를 통해 컨테이너에서 발생하는 로그의 적재 및 특정 키워드를 통한 시각화가 쉽게 가능해진다.

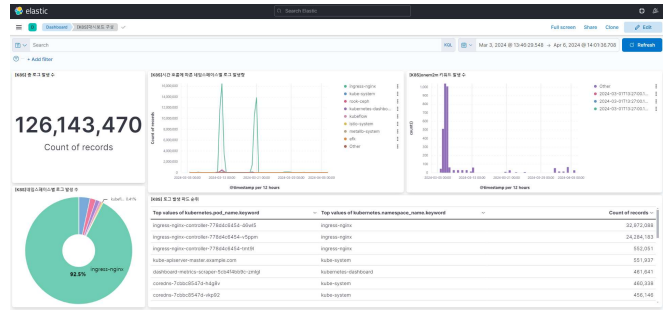


그림 6 EFK 기반 Kibana 대시보드

2.3.2 기능 검증 툴

IoT 서버로 oneM2M을 채택하여 실행 중으로 프로토콜은 release2에서 사용하는 모든 프로토콜이 바인딩한 상태로 테스트는 http 프로토콜을 이용하여 진행하였다.

테스트 클라이언트로는 Apache Jmeter를 이용하여 진행하였고, 실제 장치에서 서버로 전달하는 데이터를 담아서 진행하였으며 2,000대까지 부하 테스트를 진행하였다.

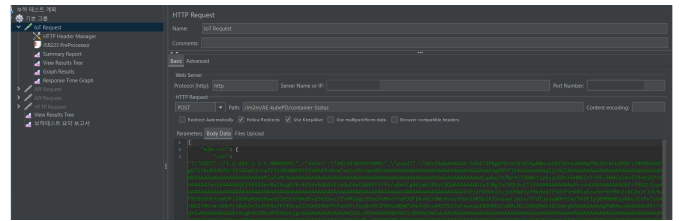


그림 7 http 기반 Jmeter 테스트 환경 구성

2.3.3 테스트 결과

부분방진 관련 데이터들을 수신 및 분석하여 처리된 것을 확인하였으며, 각 모니터링 툴에서도 리소스의 상태 변화를 관측할 수 있었다.

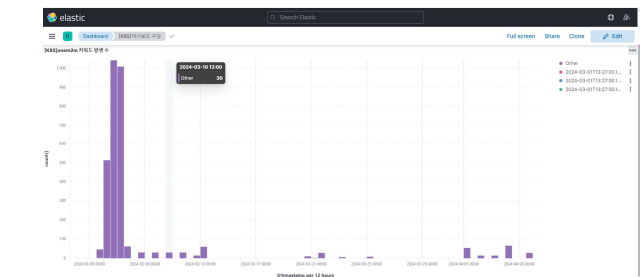


그림 8 Log 데이터 기반 oneM2M 키워드 변화량

3. 결론

공용 인프라의 복잡성과 규모가 날이 갈수록 증대되는 가운데 시스템의 스케일업 및 다양한 솔루션을 복합 운영 및 통합 감시를 해야 하는 시스템의 경우 안정적인 운영을 위해서는 오케스트레이션 툴과 애드온 기능들의 도입을 고려하는 것이 적합하다고 생각한다.

이 논문은 2024년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No. 2020-0-00061, 디지털트윈 기반의 지하공동구 화재, 재난 지원 통합플랫폼 기술개발)