



Bitbucket

NAME	Project URL	Comment
kai-backend	https://bitbucket.org/skillsolutiondev/backend/src	including all kai studio app services in « launcher » directory: 1. aoι-loadbalancer-app: loadbalancer for Azure openAI service 2. file-manager-app 3. instance-deployer-app: listener for table <i>instances</i> change event 4. instance-manager-app 5. organization-deployer-app: listener for table <i>organizations</i> change event 6. kai-studio-resource-check 7. organization-manager-app
sdk-js	https://bitbucket.org/skillsolutiondev/sdk-js/src	open api for client to test
ui	https://bitbucket.org/skillsolutiondev/ui/src	kai-studio-app



supabase

<https://supabase.com/dashboard/project/btsrniyewpsmyhogauao>

Table name	Description	Comment
organizations	subscription_type: type of subscription, <i>STARTER / PREMIUM / ENTERPRISE</i> status: <i>DRAFT / DEPLOYED / NEED_DELETED</i>	When a new organization is inserted into database, it will be deployed automatically. (<i>Via organization-deployer-app</i>)
Instances	status: <i>NEED_DEPLOY / TO_DEPLOY / DEPLOYED / TO_DELETED</i> api_key: api key credits: request times left for api private_ip: can have access to kai api service db_password: password to connect to postgresDB	When a new instance is inserted into database, it will be deployed automatically.(<i>Via instance-deployer-app</i>)



Organization






has multiple



Instances



Azure Resources

Resource group name: kai-studio		
NAME		Type
 <u>kai-studio.ai</u>	DNS zone	
 <u>kai-instances</u>	Azure Database for PostgreSQL flexible server	
 <u>filestoragekaistudio</u>	Storage account	
 <u>k8s-kaistudio</u>	Kubernetes service	
 kaistudio	Container registry	



App Services

NAME	Access	CPU	RAM	Pod	IMAGE	DESCRIPTION
aoi-loadbalancer-app	internal access	250m	500Mi	1 pod	kaistudio.azurecr.io/ aoi-loadbalancer :{tag}	Azure OpenAI Service
file-manager-app	https://fma.kai-studio.ai	250m	500Mi	2 pods	kaistudio.azurecr.io/ file-manager :{tag}	File Manager
instance-deployer-app	no http access	500m	500Mi	1 pod	kaistudio.azurecr.io/ instance-deployer :{tag}	Instance Deployer
instance-manager-app	https://ima.kai-studio.ai	250m	500Mi	2 pods	kaistudio.azurecr.io/ instance-manager :{tag}	Instance Manager
organization-deployer-app	no http access	250m	500Mi	1 pod	kaistudio.azurecr.io/ organization-deployer :{tag}	Organization Deployer
kai-studio-resource-check	no http access	250m	500Mi	1 pod	kaistudio.azurecr.io/ kai-studio-resource-check :{tag}	Check KAI Studio resource on Azure periodically / once per week
kai-studio-ui-app	https://app.kai-studio.ai	250m	200Mi	1 pod	kaistudio.azurecr.io/ kai-studio-ui :{tag}	Front user/client app interface
organization-manager-app	https://oma.kai-studio.ai	250m	500Mi	1 pod	kaistudio.azurecr.io/ organization-manager :{tag}	Organization Manager

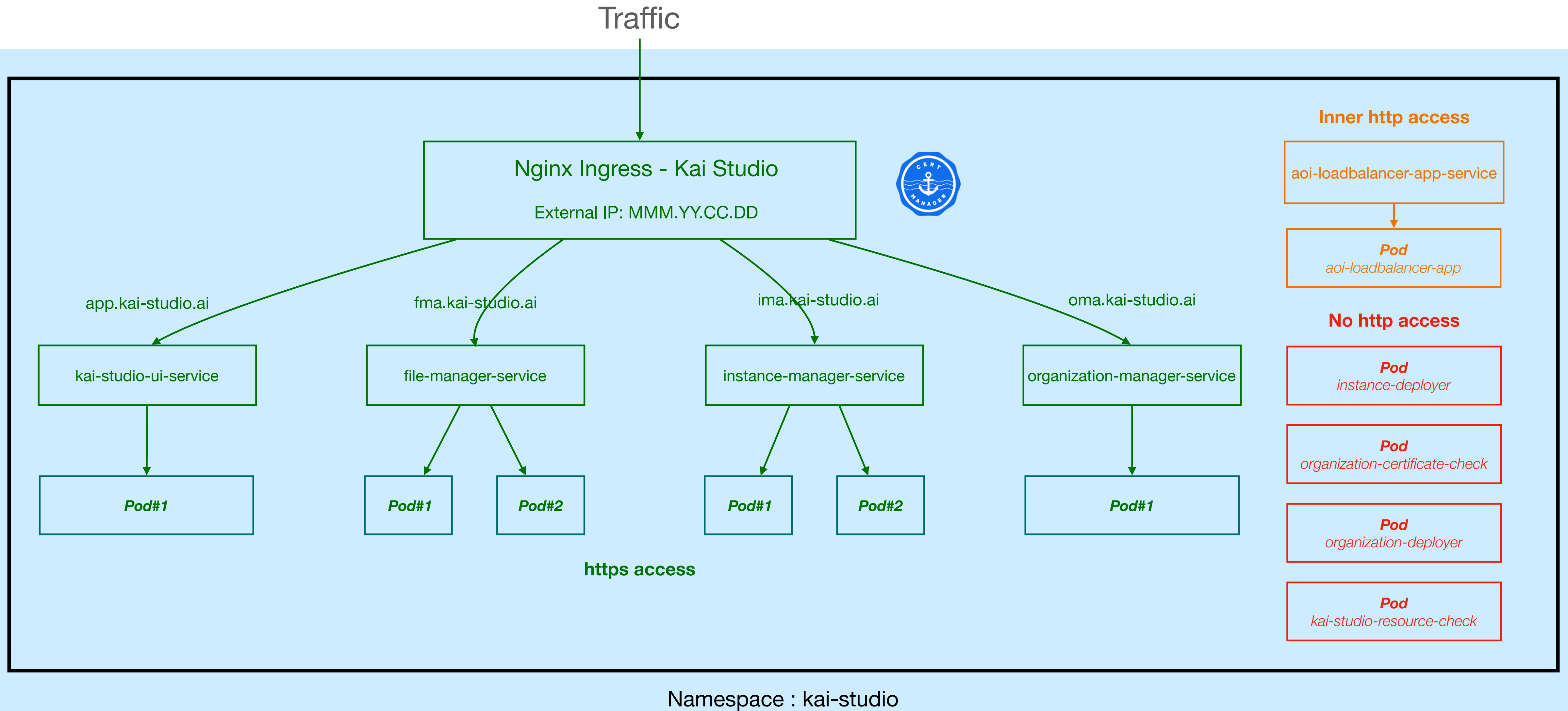


App Services

NAME	Access	CPU	RAM	Pod	IMAGE	DESCRIPTION
instance-api-gateway	https://{c1}.kai-studio.ai	250m	500Mi	2 pod	kaistudio.azurecr.io/instance-api-gateway:{tag}	API Gateway, c1 means its organization id.
kaiapi-\${instanceId1}	internal access	500m(request) 2000m(limit)	500Mi(request) 2000Mi(limit)	1 pod	kaistudio.azurecr.io/kaiapi:{tag}	API pod
kaiapi-\${instanceId2}	internal access	500m(request) 2000m(limit)	500Mi(request) 2000Mi(limit)	1 pod	kaistudio.azurecr.io/kaiapi:{tag}	API pod

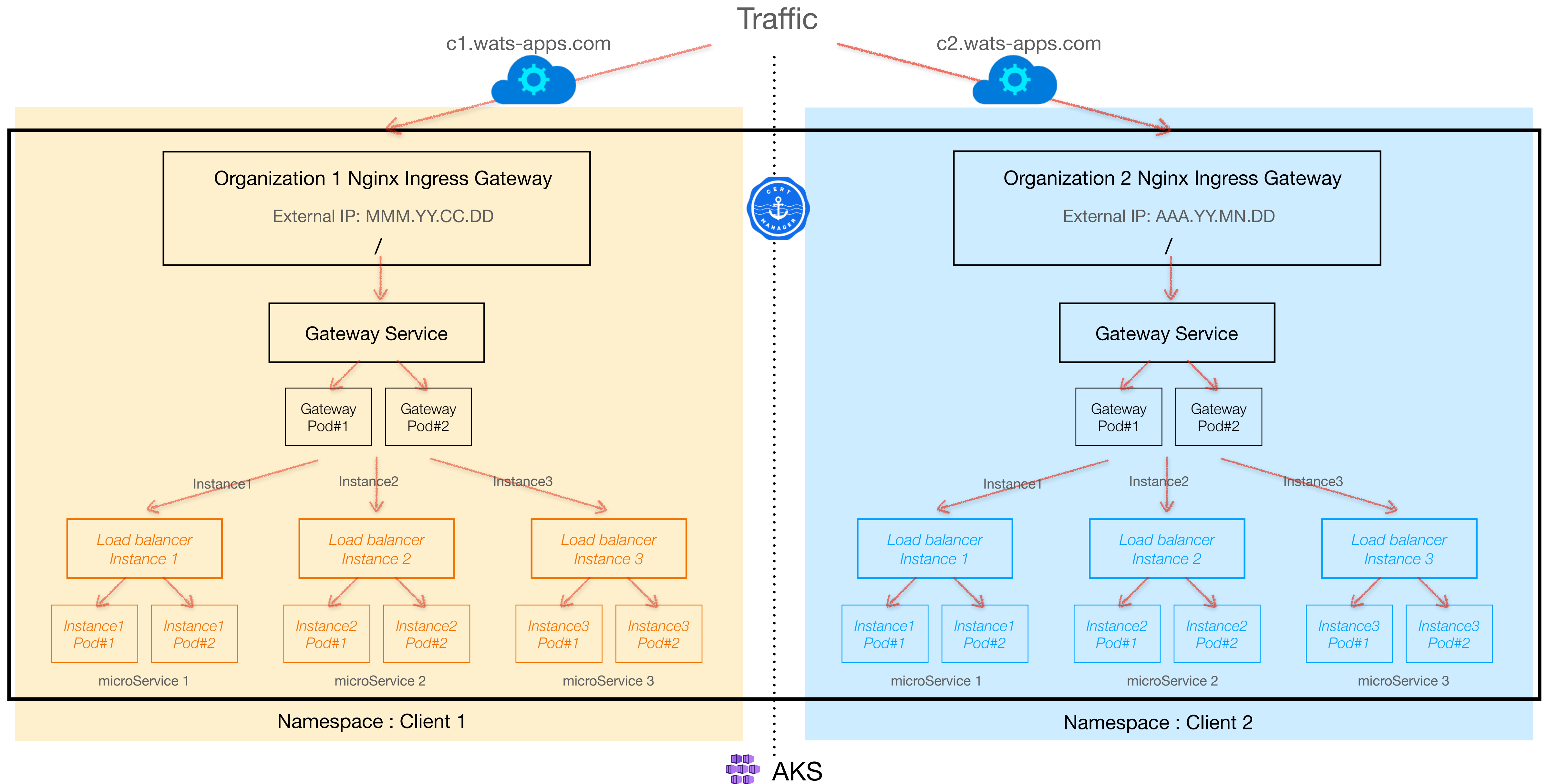


Architecture - KAI Studio





Architecture - Organization / Instance



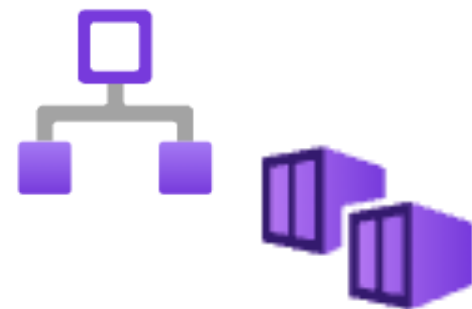


AKS Main Steps



Create namespace by client

- `kubectl create namespace client-example`



Create ingress/service/deployment of **Gateway**
+ service/deployment of **Application**
under client namespace

- `kubectl apply -f example-gateway-ingress.yaml -n client-example`
- `kubectl apply -f example-gateway-service.yaml -n client-example`
- `kubectl apply -f example-gateway-deployment.yaml -n client-example`
- `kubectl apply -f example-instance-service.yaml -n client-example`
- `kubectl apply -f example-instance-deployment.yaml -n client-example`



Register DNS + IP into Azure DNS

- `az network dns record-set a add-record...`



Generate CA by cert-manager

- `kubectl apply -f example-domain-cluster-issuer.yaml`
- `kubectl apply -f example-domain-certificate.yaml`

N.B: 1 cluster just needs 1 ClusterIssuer



Configurations



Azure OpenAI

```
export MAOI_HOST=http://aoi-loadbalancer-app-service.kai-studio
export MAOI_MAX_CONSUMPTION_PER_MINUTE="'30000'"
export MAOI_MAX_TOKEN_PER_REQUEST="'8150'"
export INSTANCE_BS_ID="xxxxxx"
export INSTANCE_KBS="xxxxxx"
```



PostgreSQL

```
export PG_HOST=host-example
export PG_USER=admin-example
export PG_PASS=password-example
export PG_DB=database-example
```



K8S

```
export DOCKER_SERVER=kaistudio.azurecr.io
export DOCKER_USERNAME=kaistudio
export DOCKER_PASSWORD=uF6ylKeIrxS3Qh0fAvly77yHglbGWQp7IMiNFvgDbV+ACRDkjeXE
export SERVICE_NAME=kaiapi-service-xxxx
export DEPLOYMENT_NAME=kaiapi-xxxx
export GATEWAY_SERVICE_NAME=gateway-service-xxxx
export INGRESS_CLASS_NAME=ingress-nginx-xxxx
```