

Installing ELERA

After the ELERA platform dependencies are installed, the GKE cluster can support the installation.

Obtain Your ELERA Container Registry Key

The ELERA container registry is a private registry and requires a password to pull the Helm chart and the Docker images. This should have been provided to you by Toshiba. You will need it in the next two sections.

Create a Kubernetes Secret for ELERA OCI Registry

The ELERA Container Registry is a private container registry that requires a token for access. This describes how to create and deploy a Kubernetes secret using the ELERA Container Registry key.

Set environment variables - Create two environment variables and replace the ELERA Container Registry Username and Key and the support email placeholders:

```
username=<ELERA Id>
password=<ELERA key>
support_email=<support email>
```

Generate new Kubernetes secret:

```
kubectl create secret docker-registry regcred \
--docker-server=tcgseleraacr.azurecr.io \
--docker-username=$username \
--docker-password=$password \
--docker-email=$support_email \
-o yaml --dry-run \
> elera-secret.yaml
```

Deploy the Secret to the Kubernetes Cluster

This section describes how to deploy the new secret, `elera-secret.yaml`, to the Kubernetes cluster.

Change to the ELERA-demo namespace:

```
kubectl config set-context --current --namespace=elera-demo
```

Deploy the secret:

```
kubectl apply -f elera-secret.yaml
```

Use `kubectl` to verify the secret was created:

```
$ kubectl get secret/regcred
NAME      TYPE                      DATA  AGE
regcred   kubernetes.io/dockerconfigjson  1      31h
```

Enabling Helm OCI Registry Support

ELERA is installed using the Helm package manager. The ELERA Helm chart is stored with ELERA container images in Toshiba's private ELERA Container Registry.

Note: Helm will migrate over to using OCI registries replacing current Helm repositories. These features are currently available as preview. These features need to be enabled within Helm.

Add the ELERA Container Registry to Helm

This section describes adding the ELERA registry to the local Helm environment.

Enable OCI registry support:

```
$ export HELM_EXPERIMENTAL_OCI=1
```

If they aren't already defined, create two environment variables replacing the ELERA Container Registry Username and Key and the support email placeholders:

```
username=<ELERA Id>
password=<ELERA key>
```

Login into the ELERA OCI Registry:

```
echo $password | helm registry login tgcseleraacr.azurecr.io \
--username $username --password-stdin
```

Download and Export the Helm Chart

Download the ELERA Helm chart and export it to the local environment.

Download or pull the ELERA chart from the ELERA container registry

```
$ helm chart pull tgcseleraacr.azurecr.io/elera-platform: 1.0.0
```

Export the ELERA Helm chart to the local directory. When the export is finished, a new elera-platform folder is created containing the uncompressed ELERA Helm chart.

```
$ helm chart export tgcseleraacr.azurecr.io/elera-platform: 1.0.0
```

Run `helm chart list` to confirm access to artifacts.

This is an example of command and run results:

```
$ helm chart list
REF                                     NAME                VERSION DIGEST  SIZE    CREATED
elera-platform:1.0.0                 elera-platform      1.0.0  6200bb4  6.7 KiB  25
hours tgcseleraacr.azurecr.io/elera-platform:1.0.0  elera-platform      1.0.0  6200bb4  7.4 KiB
23 minutes
```

Install the ELERA Platform to the Kubernetes Cluster

This is the final step for installing the ELERA platform to the Kubernetes cluster. The default install uses a dummy domain name of example.com or you can provide your own domain name. Do either step 1 or step 2:

Use the Helm install command to install the ELERA platform to the Kubernetes cluster using the default domain name “example.com”.

```
helm install elera-platform-demo elera-platform \
--set usePrivateRepo=true
```

Or

Use the Helm install command for installing the ELERA platform to the Kubernetes cluster with your own domain name. After the ELERA Platform is installed, you need to work with your DNS hosting provider to update the domain name and make sure it points the ELERA platform’s IP address.

```
helm install elera-platform-demo elera-platform \
--set usePrivateRepo=true \
--set domainname=myawesomedomain.com
```

After loading the data into the local Mongo DB the ELERA platform is installed and running. You can view the status of the installation by running the following command.

```
$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
data-loader-6f46fcb6b6-d4tm7	1/1	Running	0	18s
elera-client-6f8c7bd47d-4hwzp	1/1	Running	0	22m
elera-platform-55d67ffb94-ln248	1/1	Running	0	22m
elera-platform-admin-ui-856c74c6fb-wkmfh	1/1	Running	0	22m
mongodb-5c8956b7b6-nnqk5	1/1	Running	0	22m
nginx-5cd67cb68c-pq2mf	1/1	Running	0	22m
rabbitmq-855dcf59c9-j9jts	1/1	Running	0	22m

Monitor the status of the data-loader pod. When the status is Completed, the demo store is available for use.

After Installation is Complete

The ELERA platform is now installed on the cluster. Use the following commands to generate your URL to ELERA:

```
kubectl get svc/nginx --no-headers=true \
    | awk '{printf "The main ELERA platform URL is\n", $4}'
http://%s/webpos/index.jsp?deviceId=DEV\_ELERA\n", \$4}
kubectl get svc/nginx --no-headers=true \
    | awk '{printf "The ELERA Admin console URL is\n", $4}'
http://%s/configurations-admin-ui\n", \$4}
```