

Overslaan:

De volgende stappen zijn op het cluster reeds voorbereid

In het hoofdstuk 'Certificaten aanmaken' slaan we de volgende stappen over:

- Cert-manager installeren
- CA Issuer maken
- Private key maken
- Certificaat maken
- Secret maken

En pakken het verhaal dus weer op bij 'Extern certificaat aanmaken'

In hoofdstukken 5 en 6 slaan we de stap 'intern certificaat installeren' over.

Aangepast:

Namespace aamaken (hoofdstuk 1)

Omdat we met meerdere mensen op hetzelfde cluster werken bij deze training gebruiken we niet de namespace NLX, maar een persoonlijke namespace, dus met je eigen naam.

Extern certificaat aanmaken (hoofdstuk 2)

Voor deze tutorial gebruiken we en subdomein van nlx-enableu.commonground.nu, dus Common Name zou bijvoorbeeld <naam>-inway.nlx-enableu.commonground.nu kunnen zijn.

Daarnaast moeten we bij het aanmaken van het secret een ding wijzigen in het commando:

- In de optie `--namespace=nlx`, vervangen we NLX door de in namespace die we aangemaakt hebben.

Postgresql installeren (hoofdstuk 3)

In plaats van met Helm op onze eigen computer installeren we Postgres via Dashkube.

Als we het cluster openen, gaan we naar installations en klikken op "Add softare", vervolgens zoeken we op postgresql.

Als we bitnami postgresql gevonden hebben, openen we het venster 'Install' en passen het veld 'namespace' aan naar onze eigen namespace. Daarna voegen we in het tabblad 'Values' het veld postgresqlDatabase toe met de waarde postgresqlDatabase=nlx_management.

The screenshot shows a 'Software' window titled 'nlx-management installation wizard'. It contains a form with the following fields:

- name ***: nlx-management
- namespace ***: robert-example
- chart repository ***: https://charts.commonground.nl/
- chart ***: nlx-management
- cluster ***: CYSO Cluster

At the bottom right, there are two buttons: 'Close' and 'Install'.

Intern certificaat installeren (hoofdstuk 4)

In plaats van het door NLX in de handleiding meegeleverde bestand (management-internal-tls.yaml) gebruiken we het in de set aanwezige bestand internal-tls.yaml, waarin we [namespace] vervangen door onze eigen namespace.

NLX management chart (hoofdstuk 4)

Hier wijken we af van de gevolgde procedure door Dashkube te gebruiken in plaats van helm op onze eigen computer.

Als we het cluster openen, gaan we naar installations en klikken op "Add software", vervolgens zoeken we op nlx-management.

Als we NLX Management gevonden hebben, openen we het venster 'Install' en passen het veld 'namespace' aan naar onze eigen namespace.

Vervolgens openen we het tabblad 'Values', en maken de volgende aanpassingen:

- In de categorie postgresql passen we de volgende velden aan:
 - postgresql.database: 'nlx_management'
 - postgresql.hostname: postgresql
 - postgresql.sslMode: disable
 - postgresql.username: postgres
 - postgresql.password: het wachtwoord van postgres, op te vragen met het volgende commando: `kubectl get secret --namespace <namespace> postgresql -o jsonpath="{.data.postgresql-password}" | base64 -decode`
- In de categorie ingress passen we de volgende velden aan:
 - ingress.enabled: true
 - ingress.class: nginx
- Vervolgens moeten we de volgende velden handmatig toevoegen:
 - tls.organization.existingSecret: external-tls
 - tls.internal.existingSecret: <namespace>-nlx-internal-tls
 - tls.organization.rootCertificatePEM: regel 2 van het bijgeleverde bestand roots.txt
 - tls.internal.rootCertificatePEM: regel 5 van datzelfde bestand
 - config.directoryInspectionHostname: directory-inspection-api.demo.nlx.io
 - config.directoryRegistrationHostname: directory-registration-api.demo.nlx.io
 - config.enableBasicAuth: true
 - ingress.annotations.cert-manager.io/cluster-issuer: letsencrypt-prod
 - ingress.hosts[0]: <naam>.nlx-enableu.commonground.nu
 - ingress.tls[0].secretName: nlx-management-ingress.tls
 - ingress.tls[0].hosts[0]: <naam>.nlx-enableu.commonground.nu

NLX management administrator aanmaken (hoofdstuk 4)

In plaats van het uitvoeren van de in de handleiding gevolgde procedure openen we nu in Dashkube de NLX management-installatie en klikken op 'Bash' in het menu. Er opent nu een bash scherm waarmee we commando's kunnen uitvoeren op de container. In het invoerveld voeren we dan het volgende commando in (waarin we de gegevens tussen <> aanpassen naar de gewenste gegevens):

```
nlx-management-api create-user --email <user@example.com> --password  
<password> --role admin
```

Inway chart installeren (hoofdstuk 5)

Hier kijken we af van de gevolgde procedure door Dashkube te gebruiken in plaats van helm op onze eigen computer.

Als we het cluster openen, gaan we naar installations en klikken op "Add software", vervolgens zoeken we op nlx-inway.

Als we NLX Inway gevonden hebben, openen we het venster 'Install' en passen het veld 'namespace' aan naar onze eigen namespace.

Vervolgens openen we het tabblad 'Values', en maken de volgende aanpassingen:

- In de categorie TLS passen we de volgende velden aan:
 - `tls.internal.existingSecret`: <namespace>- nlx-internal-tls
 - `tls.internal.rootCertificatePEM`: regel 5 van het bijgeleverde bestand roots.txt
 - `tls.organization.existingSecret`: external-tls
 - `tls.organization.rootCertificatePEM`: regel 2 van het bijgeleverde bestand roots.txt
- In de categorie config passen we de volgende velden aan:
 - `config.name`: <naam>-inway
 - `config.selfAddress`: het adres dat we in [Extern certificaat aanmaken \(hoofdstuk 2\)](#) als common name hebben gebruikt, aangevuld met :443
 - `config.managementAPI.address`: nlx-management-api.<namespace>.svc.cluster.local:443
 - `config.directoryRegistrationHostname`: directory-registration-api.demo.nlx.io
- In de categorie transactionLog passen we het volgende veld aan:
 - `transactionLog.enabled`: false

Outway chart installeren (hoofdstuk 5)

Hier kijken we af van de gevolgde procedure door Dashkube te gebruiken in plaats van helm op onze eigen computer.

Als we het cluster openen, gaan we naar installations en klikken op "Add software", vervolgens zoeken we op nlx-inway.

Als we NLX Inway gevonden hebben, openen we het venster 'Install' en passen het veld 'namespace' aan naar onze eigen namespace.

Vervolgens openen we het tabblad 'Values', en maken de volgende aanpassingen:

- De volgende velden moeten we toevoegen:
 - `config.managementAPI.address`: nlx-management-api.<namespace>.svc.cluster.local:443
 - `config.directoryInspectionHostname`: directory-inspection-api.demo.nlx.io
 - `transactionLog.enabled`: false
 - `tls.organization.existingSecret`: external-tls
 - `tls.internal.existingSecret`: <namespace>- nlx-internal-tls
 - `tls.organization.rootCertificatePEM`: regel 2 van het bijgeleverde bestand roots.txt
 - `tls.internal.rootCertificatePEM`: regel 5 van datzelfde bestand
 - `https.enabled`: false