Quarkus - Kubernetes Config

Quarkus includes the `kubernetes-config` extension which allows developers to use Kubernetes ConfigMaps and Secrets as a configuration source, without having to mount them into the Pod running the Quarkus application.

Configuration

Once you have your Quarkus project configured you can add the `kubernetes-config` extension by running the following command in your project base directory.

```
./mvnw quarkus:add-extension -Dextensions="kubernetes-config"
```

This will add the following to your `pom.xml`:

```
<dependency>
    <groupId>io.quarkus</groupId>
    <artifactId>quarkus-kubernetes-config</artifactId>
</dependency>
```

Usage

The extension works by reading ConfigMaps and Secrets directly from the Kubernetes API server using the Kubernetes Client.

The extension understands the following types of ConfigMaps and Secrets as input sources:

- ConfigMaps and Secrets that contain literal data (see this for an example on how to create one)
- ConfigMaps and Secrets created from files named `application.properties`, `application.yaml` or `application.yml` (see this for an example on how to create one).

You have to explicitly enable the retrieval of ConfigMaps and Secrets by setting `quarkus.kubernetes-config.enabled=true`. The default is `false` in order to make it easy to test the application locally.

Afterwards, set the `quarkus.kubernetes-config.configmaps` property to configure which ConfigMaps should be used. Set the `quarkus.kubernetes-config.secrets` property to configure which Secrets should be used. To access ConfigMaps and Secrets from a specific namespace, you can set the `quarkus.kubernetes-config.namespace` property.

Priority of obtained properties

The properties obtained from the ConfigMaps and Secrets have a higher priority than (i.e. they

override) any properties of the same name that are found in `application.properties` (or the YAML equivalents), but they have lower priority than properties set via Environment Variables or Java System Properties.

Kubernetes Permissions

Since reading ConfigMaps involves interacting with the Kubernetes API Server, when RBAC is enabled on the cluster, the ServiceAccount that is used to run the application needs to have the proper permissions for such access.

Thankfully, when using the `kubernetes-config` extension along with the Kubernetes extension, all the necessary Kubernetes resources to make that happen are automatically generated.

Secrets

By default, the Kubernetes extension doesn't generate the necessary resources to allow accessing secrets. Set `quarkus.kubernetes-config.secrets.enabled=true` to generate the necessary role and corresponding role binding.

Configuration Reference

🔒 Configuration property fixed at build time - All other configuration properties are overridable at runtime

Configuration property	Type	Default
`quarkus.kubernetes-config.enabled` If set to true, the application will attempt to look up the configuration from the API server	boolean	`false`
`quarkus.kubernetes-config.fail-on-missing-config` If set to true, the application will not start if any of the configured config sources cannot be located	boolean	`true`
`quarkus.kubernetes-config.config-maps` ConfigMaps to look for in the namespace that the Kubernetes Client has been configured for	list of string	
`quarkus.kubernetes-config.secrets` Secrets to look for in the namespace that the Kubernetes Client has been configured for. If you use this, you probably want to enable `quarkus.kubernetes-config.secrets.enabled`.	list of string	

quarkus.kubernetes-config.namespace		
Namespace to look for config maps and secrets. If this is not specified, then the namespace configured in the kubectl config context is used. If the value is specified and the namespace doesn't exist, the application will fail to start.	string	

quarkus.kubernetes-config.namespace

Namespace to look for config maps and secrets. If this is not specified, then the namespace configured in the kubectl config context is used. If the value is specified and the namespace doesn't exist, the application will fail to start.