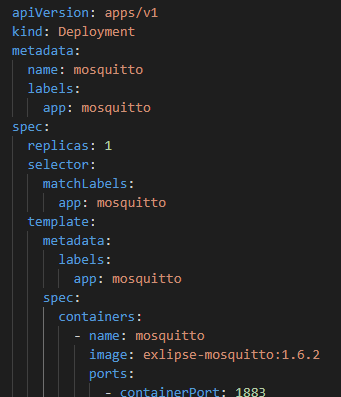
YAML – Vincular Secret y ConfigMap con contenedor

Para esto, debemos tomar como base lo siguiente:

* Ya tener dentro del ecosistema de **kubernetes** un **ConfigMap** y un **Secret**.
* Un archivo de **Deployment** para vincular con estos dos.

Vamos a partir de un archivo de **Deployment** básico; en este caso se decidió seguir un ejemplo que despliega **mosquitto:**



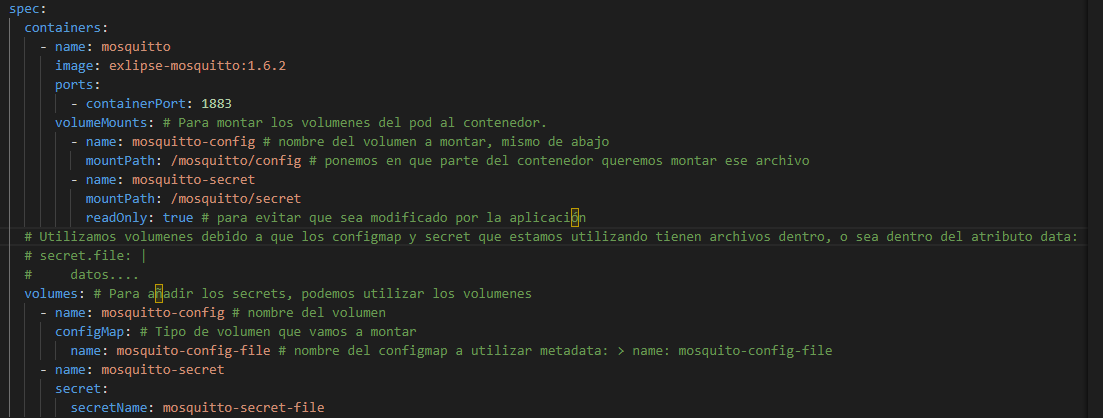
* Hay dos formas de realizar este procedimiento. La primera es mediante el uso de volúmenes; este caso sería si el **secret** o el **configmap** dentro del atributo **data**, apuntan a archivos:



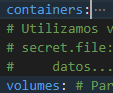
* El segundo caso es cuando el **configMap** o **secret**, posee una estructura interna de clave – valor:



Vamos primero con la configuración que correspondiente a la primera variante:



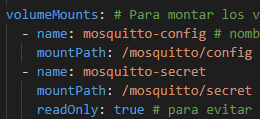
* Aunque en la imagen está, vamos a explicarlo a continuación:
  + Primero creamos los volúmenes del **POD**, esto se hace añadiendo la etiqueta **volumes** al mismo nivel que la de **container**:



* + Dentro de **volumes** debemos definir los datos correspondientes a cada **volumen**:
    - **name**: nombre que le daremos al volumen creado
    - **type**: depende del tipo de archivo que vayamos a crear, en este caso puede ser configMap o secret.
      * **name:** nombre que posee el **configMap** a montar, es importante que el nombre que se ponga aquí sea el mismo que posee el **configMap** creado.
    - **-**: el guion delante de la propiedad **name** se utiliza para hacer referencia a una sola configuración. Es decir, si tenemos dos volúmenes dentro de nuestra configuración, estos los separamos mediante el guion (-) delante del atributo **name**:

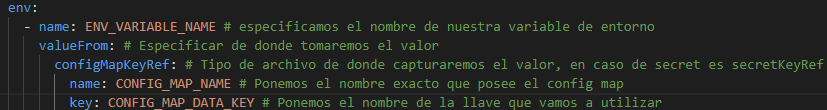


* + - * Es una forma de enumerar los diferentes conjuntos de atributos.
  + Ahora toca manipular el contenedor. La idea es utilizar los volúmenes creados en el **POD** y montarlos dentro del contenedor:

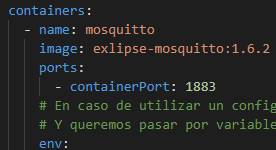


* + - El atributo **volumeMounts** nos permitirá decidir qué volúmenes del **POD** queremos montar en el contenedor.
      * Posteriormente enumeramos los contenedores a montar:
        + **name**: exactamente el mismo nombre del volumen del **POD** que vamos a utilizar.
        + **mountPath**: en que parte, dentro del contenedor, vamos a montar el/los archivo(s).
        + **readOnly**: por defecto está falso, pero si queremos evitar que el contenedor modifique alguno de estos valores, se debe poner el verdadero.

Ahora nos corresponde ver como sería la configuración con la segunda opción. La idea en este caso es utilizar los valores del **configMap** o **Secret** como variables de entorno del contenedor:



* Este es un poco más sencillo que el anterior.
  + Primero definiremos las variables de entorno dentro del container:



* + Dentro del atributo **env**, comenzamos a enumerar las diferentes variables de entorno:
    - **name**: el nombre de la variable de entorno para el contenedor.
    - **valueFrom**: permite definir de donde tomaremos la variable.
      * **type:** el tipo de archivo que vamos a utilizar. Puede ser **secretKeyRef** o **configMapKeyRef.**
        + **name**: nombre exacto del **secret** o el **configMap**.
        + **key:** nombre exacto de la llave que vamos a utilizar.