ONAP secret management

WIP WIP WIP WIP WIP WIP WIP WIP WIP WIP WIP WIP

Problem statement

ONAP is quite a complicated application. It consists of several components and each component is often a set of micro services. All those components have to communicate with each other and many of them also needs to access the Database. This requires some secret material to be distributed at the deployment time. There are two types of secret materials provided at the deployment time:

1. Chart-internal secrets
2. Chart-external secrets

Chart-internal secret is a sensitive material that is used only within given chart and its subcharts and does not depend on any external system. Examples is a password to service database.

Chart-external secret is a sensitive material that cannot be produced within a chart and has to be delivered from the outside.

Currently all sensitive material that is chart-internal is constant and the same for every ONAP deployment (unless explicitly change by the deployer but it's not an easy task). On the other hand chart-external material is often set to some insecure defaults and charts never check whether they have been really provided by the deployer. Both cases are serious security issues as they allow to easily execute dictionary-based attack.

Solution

In order to improve current situation we propose to:

- Remove all default values for chart-external secrets
- Provide the infrastructure to fail deployment of chart if any of external secrets has not been provided by the deployer
- Provide the infrastructure to generate chart-internal secrets on per-deployment basis
    - Secret material cannot be perfectly random to ensure possibility of component upgrade
    - Instead of that it should be derived from secret material provided by the user at the deployment time (aka masterPasword)
    - To achieve this functionality derivePassword function from spring library which implements well known master password algorithm should be used
    - It should be documented that master password is extremely sensitive information and should be always provided from the command line in at the deployment type provided that user ensured that it's not going to be stored in shell history file
- Sensitive material should be provided to ONAP services as a kubernetes secret
- Every project may chose if they prefer to user in-memory volume or environment variables to obtain content of the secret
- Material inside a kubernetes secret should be stored in plain text
- Requirement towards underlying kubernetes cluster should be specified to ensure that the encryption at rest plugin is used and secrets are never written to the disk (etcd) in a plain text form
- Requirement towards all ONAP component should be created to ensure that sensitive information obtain from secret is never written to the logs
- Provide the infrastructure which allows user to provide password or any other sensitive material as a string or as a reference to already existing kubernetes secret