TUESDAY-WEDNESDAY
JULY 13-14

ORGANISERS

•virtual event•
AGENDA

• Kubernetes Operator Patterns (~5min)
• RabbitMQ Cluster Operator
• Messaging Topology Operator
• Demo (~7min)
Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications.
kind: StatefulSet
StatefulSet

...THE CHALLENGES

• Day-2 operations: scaling, config changes, and upgrades
KUBERNETES OPERATOR PATTERN

- Extensions to Kubernetes that provide custom resources for management of applications, services and their components
- It makes use of custom resources and the concept of controllers to manage applications extensions to Kubernetes and encode product specific knowledge
RABBITMQ CLUSTER OPERATOR

Controller

Custom Resource Definition (CRD)
• Desired state defined in manifests
• Controllers analyze differences between observed and desired state
• Reconcile any difference
```yaml
apiVersion: rabbitmq.com/v1alpha1
kind: RabbitmqCluster
metadata:
  name: hello-world
```
```yaml
rabbitmq.yml
apiVersion: rabbitmq.com/v1beta1
kind: RabbitmqCluster
metadata:
  name: plugins
spec:
  rabbitmq:
    additionalConfig: |
      log.file.level = debug
    additionalPlugins:
      - rabbitmq_federation
      - rabbitmq_federation_management
      - rabbitmq_shovel
```

rabbitmq.yaml

```yaml
apiVersion: rabbitmq.com/v1beta1
kind: RabbitmqCluster
metadata:
  name: example
spec:
  replicas: 3
  service:
    type: LoadBalancer
  persistence:
    storageClassName: ssd
    storage: "50Gi"
  resources:
    requests:
      cpu: 4
      memory: 10Gi
    limits:
      cpu: 4
      memory: 10Gi
  tls:
    secretName: tls-secret
```
```yaml
apiVersion: rabbitmq.com/v1beta1
kind: RabbitmqCluster
metadata:
  name: override
spec:
  override:
    service:
      spec:
        ports:
        - name: additional-port
          protocol: TCP
          port: 12345

statefulSet:
  spec:
    template:
      spec:
        containers:
        - name: rabbitmq
          ports:
            - containerPort: 12345
              name: additional-port
              protocol: TCP
            - name: side-car
```
Support PVC expansion 😓 #606

ChunyiLyu merged 6 commits into main from persistence-resize on 18 Feb

Note to reviewers: remember to look at the commits in this PR and consider if they can be squashed

Summary Of Changes

1. update default PVC ‘persistence’ storage capacity in statefulSetBuilder
2. reconcilePVC() check if the default PVC ‘persistence’ storage capacity has updated; if yes:
   i. delete current statefulSet with delete propagation policy set to "Orphan"
   ii. update PVCs manually
   iii. recreate statefulSet (statefulSet will be automatically created by the CreateOrUpdate() call later as a result of its deletion in step 1; does not need to be done in reconcilePVC() explicitly)
<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalPorts</td>
<td>Test for documented examples (#590)</td>
<td>5 months ago</td>
</tr>
<tr>
<td>community-plugins</td>
<td>Provide container name for pod exec in test examples</td>
<td>2 months ago</td>
</tr>
<tr>
<td>custom-configuration</td>
<td>Test for documented examples (#590)</td>
<td>5 months ago</td>
</tr>
<tr>
<td>default-security-context</td>
<td>Allow SecurityContext and PodSecurityContext to be reset t...</td>
<td>12 days ago</td>
</tr>
<tr>
<td>federation-over-tls</td>
<td>Provide container name for pod exec in test examples</td>
<td>2 months ago</td>
</tr>
<tr>
<td>hello-world</td>
<td>Test for documented examples (#590)</td>
<td>5 months ago</td>
</tr>
<tr>
<td>import-definitions</td>
<td>Provide container name for pod exec in test examples</td>
<td>2 months ago</td>
</tr>
<tr>
<td>mTLS-inter-node</td>
<td>Provide container name for pod exec in test examples</td>
<td>2 months ago</td>
</tr>
<tr>
<td>mTLS</td>
<td>Link to the Troubleshooting TLS guide</td>
<td>3 months ago</td>
</tr>
<tr>
<td>multiple-disks</td>
<td>Provide container name for pod exec in test examples</td>
<td>2 months ago</td>
</tr>
<tr>
<td>plugins</td>
<td>Test for documented examples (#590)</td>
<td>5 months ago</td>
</tr>
<tr>
<td>pod-anti-affinity</td>
<td>Add new podAntiAffinity example and modify existing production...</td>
<td>6 days ago</td>
</tr>
<tr>
<td>production-ready</td>
<td>Add new podAntiAffinity example and modify existing production...</td>
<td>6 days ago</td>
</tr>
</tbody>
</table>
RabbitMQ Cluster Operator vs. Bitnami RabbitMQ Helm Chart vs. DIY RabbitMQ on Kubernetes
```yaml
apiVersion: rabbitmq.com/v1beta1
kind: RabbitmqCluster
name: tls
spec:
  replicas: 1
  rabbitmq:
    additionalPlugins:
      - rabbitmq_mqtt
      - rabbitmq_stomp
  tls:
    secretName: rabbitmq-tls-test-secret
```

- Additional protocol plugins
- TLS Secret
spec:
    containers:
    - name: rabbitmq
      volumeMounts:
      - mountPath: /etc/rabbitmq-tls/
        name: rabbitmq-tls
        readOnly: true
      volumes:
      - name: rabbitmq-tls
        projected:
          defaultMode: 400
          sources:
          - secret:
            name: rabbitmq-tls-test-secret
            optional: true
Additional ports for StatefulSet

- name: rabbitmq
  ports:
  - containerPort: 1883
    name: amqps
    protocol: TCP
  - containerPort: 61613
    name: stomp
    protocol: TCP
  - containerPort: 5671
    name: amqps
    protocol: TCP
  - containerPort: 15671
    name: management-tls
    protocol: TCP
  - containerPort: 15691
    name: prometheus-tls
    protocol: TCP
  - containerPort: 8883
    name: mqtts
    protocol: TCP
  - containerPort: 61614
    name: stomp
    protocol: TCP
  - containerPort: 6161
    name: management-tls
    protocol: TCP

Additional ports for Service

- name: mqtt
  port: 1883
  protocol: TCP
  targetPort: 1883
- name: mqtts
  port: 8883
  protocol: TCP
  targetPort: 8883
- name: stomp
  port: 61613
  protocol: TCP
  targetPort: 61613
- name: stomps
  port: 61614
  protocol: TCP
  targetPort: 61614
- name: prometheus-tls
  port: 15691
  protocol: TCP
  targetPort: 15691
- name: management-tls
rabbitmq.conf TLS configurations

```conf
ssl_options.cert = /etc/rabbitmq/mq-tls/tls.crt
ssl_options.key = /etc/rabbitmq/mq-tls/tls.key
listeners.ssl = [::]:5671
management.ssl = [::]:15671
prometheus.ssl = [::]:9115
mqtt.listeners.ssl = [::]:1883
```

---

I'm not sure what these configurations do. 🤔🤔🤔
RABBITMQ CLUSTER OPERATOR

- Encode RabbitMQ specific knowledge
- `rabbitmq-queues rebalance`
- Enable rabbitmq plugins dynamically
- Graceful termination with `rabbitmq-upgrade drain` and `rabbitmq-upgrade await_online_quorum_plus_one`
RABBITMQ CLUSTER OPERATOR

- Supports rolling upgrades of RabbitMQ
- Customizable with StatefulSet and Service override
- Volume expansions
- Reasonable defaults
- Easy to manage and configure
- Encode RabbitMQ operation knowledge such as queue rebalance and graceful termination
- Developed and maintained by RabbitMQ team 😊
MESSAGING TOPOLOGY OPERATOR

• Manage RabbitMQ topology objects within a RabbitMQ cluster
• Work with RabbitMQ created by RabbitMQ Cluster Operator
MESSAGING TOPOLOGY OPERATOR

- Create topology objects: queues, exchanges, users, bindings, policies, and vhosts
- Configure: shovel, federations, and user permissions
```yaml
quorum-queue.yaml

---
apiVersion: rabbitmq.com/v1beta1
kind: Queue
metadata:
  name: example
spec:
  name: example  # queue name
  vhost: "/test-vhost"  # vhost; defaults to '/'
  type: quorum
  autoDelete: false
  durable: true
  rabbitmqClusterReference:
    name: test
```
fanout-exchange.yaml

---
apiVersion: rabbitmq.com/v1beta1
kind: Exchange
metadata:
  name: fanout
spec:
  name: fanout-exchange  # name of the exchange
  vhost: "/test-vhost"  # default to '/' if not provided
  type: fanout  # default to 'direct' if not provided
  autoDelete: false
  durable: true
  rabbitmqClusterReference:
    name: test  # name of the rabbitmqCluster
<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
<th>Last Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>bindings</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>exchanges</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>federations</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>permissions</td>
<td>Example uses spec.userReference for permissions crd</td>
<td>11 days ago</td>
</tr>
<tr>
<td>policies</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>queues</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>shovels</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>users</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>vhosts</td>
<td>Remove hardcoded namespace from all examples</td>
<td>Last month</td>
</tr>
<tr>
<td>README.md</td>
<td>Link tutorial in README</td>
<td>3 days ago</td>
</tr>
</tbody>
</table>

**Messaging Topology Operator examples**
Messaging Topology Operator vs. importing definitions vs. declaring objects by applications? 😐
WHAT IS GITOPS?

• Used for Kubernetes cluster management and application delivery
• Use Git as the source of truth for declarative infrastructure and applications
• Infrastructure as Code
• Argo CD (GitOps tool)
• RabbitMQ Cluster Operator and Messaging Topology Operator
• Two RabbitMQ clusters
• RabbitMQ Cluster 1 will be 3 vhosts, 3 queues, and 3 exchanges created
• RabbitMQ Cluster 2 will have federation plugin enabled with federations created between 2 queues (in different vhosts)
Useful links

- RabbitMQ Cluster Operator
- Messaging Topology Operator
- RabbitMQ community slack #kubernetes
- RabbitMQ Cluster Operator examples
- Messaging Topology Operator examples
- Kubernetes operator pattern
- CNCF 2020 survey report
- Argo CD kubecon talk
- GitOps Demo Repo
Thank you!