123Kubernetes 系列（一一七）多个Kubernetes集群资源集中展示

如今拥有多个 Kubernetes 集群很常见。组织选择拥有多个集群的原因有很多，包括：

- 隔离不同的环境：例如，您可能有一个用于开发的集群，一个用于预生产，一个用于生产。这有助于防止一个环境中的问题影响其他环境。
- 为了满足合规性要求：某些组织可能有合规性要求，规定某些应用程序必须部署到单独的集群。
- 支持不同地域：您可以将应用程序部署到不同地域的不同集群，以提高不同地域用户的性能和可用性。

拥有一个集中位置来查看资源摘要的重要性

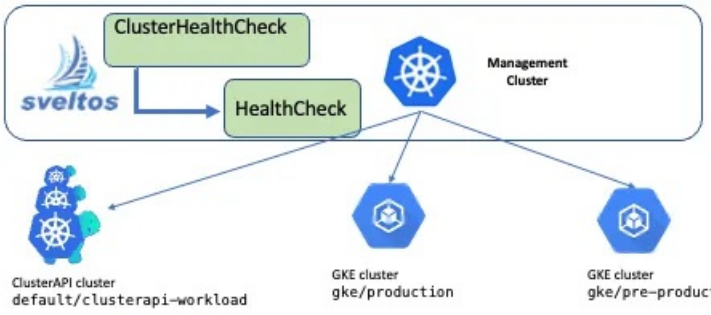
有效管理多个 Kubernetes 集群需要一个集中位置来查看资源摘要。以下是这一点如此重要的一些关键原因：

1. 集中可见性：中央位置提供资源摘要的统一视图，使您可以在一个位置监控和可视化所有集群的资源。这简化了跨多个集群的问题检测、趋势识别和问题故障排除。
2. 高效的故障排除和问题解决：通过集中的资源视图，您可以在出现问题时快速识别受影响的集群，将其与其他集群进行比较，并缩小潜在原因的范围。对资源状态和依赖关系的全面概述可以实现高效的故障排除和更快的问题解决。
3. 增强的安全性和合规性：集中的资源可见性增强了安全性和合规性监控。它使您能够监控集群配置、识别安全漏洞并确保所有集群一致遵守合规性标准。您可以从一个位置轻松跟踪和管理访问控制、网络策略和其他安全相关方面。

Sveltos 解决方案

Sveltos （https://github.com/projectsveltos）是一个开源项目，提供 Kubernetes 附加组件控制器，可简化跨多个 Kubernetes 集群的附加组件的部署和管理。

它还可以从中央位置收集和显示有关托管集群中资源的信息。




```
→  sveltosctl git:(main) KUBECONFIG=/tmp/config/mgmt_kubeconfig ./bin/sveltosctl show resources --kind=deployment
+-----------------------------+-----------------------------+-----------+-------------------+-----------------------+
|           CLUSTER           |             GVK             | NAMESPACE |       NAME        |        MESSAGE        |
+-----------------------------+-----------------------------+-----------+-------------------+-----------------------+
| default/clusterapi-workload | apps/v1, Kind=Deployment    | nginx     | nginx-deployment  | Image: nginx:1.14.2   |
| gke/pre-production          |                             | nginx     | nginx-deployment  | Image: nginx:latest   |
| gke/production              |                             | nginx     | nginx-deployment  | Image: nginx:1.14.2   |
+-----------------------------+-----------------------------+-----------+-------------------+-----------------------+

→  sveltosctl git:(main) KUBECONFIG=/tmp/config/mgmt_kubeconfig ./bin/sveltosctl show resources --kind=pod --namespace=nginx
+-----------------------------+-------------+-----------+-------------------------------------+-----------------------+
|           CLUSTER           |     GVK     | NAMESPACE |                NAME                 |        MESSAGE        |
+-----------------------------+-------------+-----------+-------------------------------------+-----------------------+
| default/clusterapi-workload | /v1, Kind=Pod | nginx     | nginx-deployment-85996f8dbd-7tctq   | Deployment: nginx     |
|                             |             | nginx     | nginx-deployment-85996f8dbd-tz4gd   | Deployment: nginx     |
| gke/pre-production          |             | nginx     | nginx-deployment-c4f7848dc-6jtwg    | Deployment: nginx     |
|                             |             | nginx     | nginx-deployment-c4f7848dc-trllk    | Deployment: nginx     |
| gke/production              |             | nginx     | nginx-deployment-676cf9b46d-k84pb   | Deployment: nginx     |
|                             |             | nginx     | nginx-deployment-676cf9b46d-mmbl4   | Deployment: nginx     |
+-----------------------------+-------------+-----------+-------------------------------------+-----------------------+
```


为此，Sveltos 使用三个主要概念：

1. ClusterHealthCheck CRD，用于告诉Sveltos从哪一组集群中收集信息；
2. HealthCheck CRD，用于告诉Sveltos要收集哪些资源；

3. Sveltosctl 是用于显示收集的信息的 CLI。

例如，您可以使用 ClusterHealthCheck 指定您希望 Sveltos 从在 env=prod 环境中运行的所有 Kubernetes 集群收集信息。然后，您可以使用 HealthCheck 指定您希望 Sveltos 收集有关这些集群中所有 Pod、部署和服务的信息。最后，您可以使用 Sveltosctl 以人类可读的格式显示收集的信息。

以下是 `sveltosctl show resources` 命令的输出示例：

```
./sveltosctl show resources --kind=pod --namespace=nginx
+-----------------------------+--------------+-----------+----------------------------------+-------------------+
|           CLUSTER           |      GVK     | NAMESPACE |               NAME               |      MESSAGE      |
+-----------------------------+--------------+-----------+----------------------------------+-------------------+
| default/clusterapi-workload | /v1, Kind=Pod | nginx     | nginx-deployment-85996f8dbd-7tctq | Deployment: nginx |
|                             |              | nginx     | nginx-deployment-85996f8dbd-tz4gd | Deployment: nginx |
| gke/pre-production          |              | nginx     | nginx-deployment-c4f7848dc-6jtwg  | Deployment: nginx |
|                             |              | nginx     | nginx-deployment-c4f7848dc-trllk  | Deployment: nginx |
| gke/production              |              | nginx     | nginx-deployment-676cf9b46d-k84pb | Deployment: nginx |
|                             |              | nginx     | nginx-deployment-676cf9b46d-mmbl4 | Deployment: nginx |
+-----------------------------+--------------+-----------+----------------------------------+-------------------+
```

此输出显示来自所有托管集群。它包含集群的名称、资源的 GVK（组、版本、种类）、命名空间、资源的名称以及有关资源的消息。

在接下来的部分中，我们将通过具体示例详细介绍这三个概念。

ClusterHealthCheck

ClusterHealthCheck 实例是一个 Kubernetes 自定义资源 (CRD)，它指定 Sveltos 应从哪些集群收集信息。它有两个主要字段：

- clusterSelector：该字段是 Kubernetes 标签选择器，指定 Sveltos 应从哪些集群收集信息。例如，提供的示例中的 clusterSelector 字段指定 Sveltos 应从具有标签 env=fv 的所有集群收集信息。
- livenessCheck：该字段引用 HealthCheck 实例。 HealthCheck 实例包含有关要收集哪些资源的详细说明。

```yaml
apiVersion: lib.projectsveltos.io/v1alpha1
kind: ClusterHealthCheck
metadata:
  name: production
spec:
  clusterSelector: env=fv
  livenessChecks:
  - name: deployment
    type: HealthCheck
    livenessSourceRef:
      kind: HealthCheck
      apiVersion: lib.projectsveltos.io/v1alpha1
      name: deployment-replicas
  notifications:
  - name: event
    type: KubernetesEvent
```

当管理集群中创建 ClusterHealthCheck 实例时，Sveltos 将收集所有与 clusterSelector 匹配的托管集群中引用的 HealthCheck 中指定的信息。

HealthCheck

HealthCheck 实例是 Kubernetes 自定义资源 (CRD)，它指定 Sveltos 应收集哪些资源。

它的规格部分包含以下字段

- Group：该字段指定 HealthCheck 所针对的 Kubernetes 资源组。
- Version：此字段指定 HealthCheck 所针对的 Kubernetes 资源的版本。
- Kind：此字段指定 HealthCheck 所针对的 Kubernetes 资源的类型。
- 命名空间：该字段可用于按命名空间过滤资源。
- LabelFilters：该字段可用于按标签过滤资源。
- Script：该字段可以包含Lua脚本，它定义自定义健康检查。该脚本必须包含一个评估方法，该方法传递一个收集的资源实例 (obj)，并且必须返回一个结构体 (hs)。

```yaml
apiVersion: lib.projectsveltos.io/v1alpha1
kind: HealthCheck
metadata:
 name: deployment-replicas
spec:
 group: "apps"
 version: v1
 kind: Deployment
 script: |
   function evaluate()
     hs = {}
     hs.status = "Progressing"
     hs.message = ""
     if obj.spec.replicas == 0 then
       hs.ignore=true
       return hs
     end
     if obj.status ~= nil then
       if obj.status.availableReplicas ~= nil then
         if obj.status.availableReplicas == obj.spec.replicas then
           hs.status = "Healthy"
           hs.message = "All replicas " .. obj.spec.replicas .. " are healthy"
         else
           hs.status = "Progressing"
           hs.message = "expected replicas: " .. obj.spec.replicas .. " available: " .. obj.status.availableReplicas
         end
       end
       if obj.status.unavailableReplicas ~= nil then
          hs.status = "Degraded"
          hs.message = "deployments have unavailable replicas"
       end
     end
     return hs
   end
```

在上面的示例中，收集了所有命名空间中的 Deployment。请求副本设置为零的部署将被忽略。收集任何其他部署，并且消息指示所有请求的副本是否处于活动状态.

这是 HealthCheck 收集 Pod 与部署关系的另一个示例。

```
apiVersion: lib.projectsveltos.io/v1alpha1
kind: HealthCheck
metadata:
 name: pod-in-deployment
spec:
 group: ""
 version: v1
 kind: Pod
 script: |
   function setContains(set, key)
    return set[key] ~= nil
   end

   function evaluate()
     hs = {}
     hs.status = "Healthy"
     hs.message = ""
     hs.ignore = true
     if obj.metadata.labels ~= nil then
       if setContains(obj.metadata.labels, "app") then
         if obj.status.phase == "Running" then
           hs.ignore = false
           hs.message = "Deployment: " .. obj.metadata.labels["app"]
         end
       end
     end
     return hs
   end
```

Sveltoctl

Sveltosctl (https://github.com/projectsveltos/sveltosctl)是 Sveltos 的 CLI。 sveltosctl show resources 命令显示收集的信息。

这里有些例子：

```
kubectl exec -it -n projectsveltos sveltosctl-0 -- ./sveltosctl show resources --kind=deployment
+----------------------------+-------------------------+----------------+---------------------------------------+----
|           CLUSTER          |           GVK           |   NAMESPACE    |                 NAME                  |
+----------------------------+-------------------------+----------------+---------------------------------------+----
| default/clusterapi-workload | apps/v1, Kind=Deployment | kube-system    | calico-kube-controllers               | All
|                            |                         | kube-system    | coredns                               | All
|                            |                         | kyverno        | kyverno-admission-controller          | All
|                            |                         | kyverno        | kyverno-background-controller         | All
|                            |                         | kyverno        | kyverno-cleanup-controller            | All
|                            |                         | kyverno        | kyverno-reports-controller            | All
|                            |                         | projectsveltos | sveltos-agent-manager                 | All
| gke/pre-production         |                         | gke-gmp-system | gmp-operator                          | All
|                            |                         | gke-gmp-system | rule-evaluator                        | All
|                            |                         | kube-system    | antrea-controller-horizontal-autoscaler | All
|                            |                         | kube-system    | egress-nat-controller                 | All
...
+----------------------------+-------------------------+----------------+---------------------------------------+----
```

```
./sveltosctl show resources
+----------------------------+-------------------------+-----------+-----------------+--------------------+
|           CLUSTER          |           GVK           | NAMESPACE |       NAME      |       MESSAGE      |
+----------------------------+-------------------------+-----------+-----------------+--------------------+
| default/clusterapi-workload | apps/v1, Kind=Deployment | nginx     | nginx-deployment | Image: nginx:1.14.2 |
| gke/pre-production         |                         | nginx     | nginx-deployment | Image: nginx:latest |
| gke/production             |                         | nginx     | nginx-deployment | Image: nginx:1.14.2 |
+----------------------------+-------------------------+-----------+-----------------+--------------------+
```

以下是过滤显示资源将显示的内容的可用选项：

```
--group=<group>: Show Kubernetes resources deployed in clusters matching this group. If not specified, all groups are cor
--kind=<kind>: Show Kubernetes resources deployed in clusters matching this Kind. If not specified, all kinds are conside
--namespace=<namespace>: Show Kubernetes resources in this namespace. If not specified, all namespaces are considered.
--cluster-namespace=<name>: Show Kubernetes resources in clusters in this namespace. If not specified, all namespaces are
--cluster=<name>: Show Kubernetes resources in the cluster with the specified name. If not specified, all cluster names a
```

审计

此功能还可用于审计目的。例如，如果我们在每个托管集群中都有 Kyverno，并且具有 Kyverno 审核策略来报告使用带有最新标签的镜像的每个 Pod，那么我们可以指示 Sveltos 使用以下 HealthCheck 实例从所有托管集群收集审核结果：

```yaml
apiVersion: lib.projectsveltos.io/v1alpha1
kind: HealthCheck
metadata:
 name: deployment-replicas
spec:
 collectResources: true
 group: wgpolicyk8s.io
 version: v1alpha2
 kind: PolicyReport
 script: |
   function evaluate()
     hs = {}
     hs.status = "Healthy"
     hs.message = ""
     for i, result in ipairs(obj.results) do
       if result.result == "fail" then
          hs.status = "Degraded"
          for j, r in ipairs(result.resources) do
             hs.message = hs.message .. " " .. r.namespace .. "/" .. r.name
          end
       end
     end
     if hs.status == "Healthy" then
       hs.ignore = true
     end
     return hs
   end
```

创建 HealthCheck 实例后，我们可以使用 `sveltosctl show resources` 命令查看所有托管集群的综合结果：

```
/sveltosctl show resources
+-----------------------------------+-------------------------------+-----------+-------------------------+----------
|              CLUSTER              |              GVK              | NAMESPACE |          NAME           |
+-----------------------------------+-------------------------------+-----------+-------------------------+----------
| default/sveltos-management-workload | wgpolicyk8s.io/v1alpha2,      | nginx     | cpol-disallow-latest-tag |  nginx/ng
|                                   | Kind=PolicyReport             |           |                         | nginx/ngi
|                                   |                               |           |                         | nginx/ngi
|                                   |                               |           |                         | nginx/ngi
+-----------------------------------+-------------------------------+-----------+-------------------------+----------
```