



Red Hat Advanced Cluster Management for Kubernetes 2.3

故障排除

故障排除

故障排除

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律通告

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Troubleshooting.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

摘要

对 Red Hat Advanced Cluster Management for Kubernetes 进行故障排除

目录

第 1 章 故障排除	5
1.1. 记录的故障排除	5
1.2. 运行 MUST-GATHER 命令进行故障排除	6
1.2.1. must-gather 情境	6
1.2.2. must-gather 过程	6
1.2.3. 在断开连接的环境中的 must-gather	7
1.3. 对安装状态处于安装或待处理状态进行故障排除	7
1.3.1. 症状：处于待处理状态	7
1.3.2. 解决问题：调整 worker 节点大小	8
1.4. 重新安装失败的故障排除	8
1.4.1. 症状：重新安装失败	8
1.4.2. 解决问题：重新安装失败	8
1.5. 离线集群的故障排除	9
1.5.1. 症状：集群状态为离线	9
1.5.2. 解决问题：集群状态为离线	9
1.6. 对带有待处理导入状态的集群进行故障排除	9
1.6.1. 症状：集群处于待处理导入状态	9
1.6.2. 鉴别问题：集群处于待处理导入状态	9
1.6.3. 解决问题：集群处于待处理导入状态	10
1.7. VMWARE VSPHERE 上创建集群的故障排除	10
1.7.1. 受管集群创建失败并显示证书 IP SAN 错误	10
1.7.1.1. 症状：Managed 集群创建失败并显示证书 IP SAN 错误	10
1.7.1.2. 鉴别问题：管理的集群创建失败并显示证书 IP SAN 错误	10
1.7.1.3. 解决问题：管理的集群创建失败，并显示证书 IP SAN 错误	10
1.7.2. 受管集群创建失败并显示未知证书颁发机构	10
1.7.2.1. 症状：管理集群创建失败并显示未知证书颁发机构	10
1.7.2.2. 鉴别问题：Managed 集群创建失败并显示未知证书颁发机构	10
1.7.2.3. 解决问题：管理的集群创建失败并显示未知证书颁发机构	11
1.7.3. 受管集群创建带有过期证书失败	11
1.7.3.1. 情况：集群创建失败并显示过期的证书	11
1.7.3.2. 鉴别问题：管理的集群创建失败并显示过期的证书	11
1.7.3.3. 解决问题：管理的集群创建失败并显示过期的证书	11
1.7.4. 受管集群创建失败且没有标记权限	11
1.7.4.1. 症状：管理集群创建失败且没有足够特权进行标记	11
1.7.4.2. 鉴别问题：Managed 集群创建会失败，没有足够权限进行标记	11
1.7.4.3. 解决问题：管理的集群创建没有足够权限进行标记	11
1.7.5. 受管集群创建失败并显示无效的 dnsVIP	11
1.7.5.1. 症状：受管集群创建失败并显示无效的 dnsVIP	12
1.7.5.2. 鉴别问题：Managed 集群创建失败并显示无效的 dnsVIP	12
1.7.5.3. 解决问题：受管集群创建失败并显示无效的 dnsVIP	12
1.7.6. 受管集群创建带有不正确的网络类型失败	12
1.7.6.1. 症状：集群创建失败并显示不正确的网络类型	12
1.7.6.2. 鉴别问题：管理的集群创建失败并显示不正确的网络类型	12
1.7.6.3. 解决问题：受管集群创建失败并显示不正确的网络类型	12
1.7.7. 受管集群创建失败并显示磁盘更改错误	12
1.7.7.1. 症状：因为错误处理磁盘更改导致添加 VMware vSphere 受管集群失败	12
1.7.7.2. 鉴别问题：添加 VMware vSphere 受管集群会因为处理磁盘更改出错而失败	13
1.7.7.3. 解决问题：因为错误处理磁盘更改导致 VMware vSphere 受管集群失败	13
1.8. OPENSIFT CONTAINER PLATFORM 版本 3.11 集群导入失败的故障排除	13
1.8.1. 症状：OpenShift Container Platform 版本 3.11 集群导入失败	13
1.8.2. 鉴别问题：OpenShift Container Platform 版本 3.11 集群导入失败	13

1.8.3. 解决问题：OpenShift Container Platform 版本 3.11 集群导入失败	13
1.9. 证书更改后导入的集群离线故障排除	14
1.9.1. 症状：证书更改后集群处于离线状态	14
1.9.2. 鉴别问题：证书更改后集群处于离线状态	14
1.9.3. 解决问题：证书更改后集群处于离线状态	15
1.10. 删除集群后命名空间会保留	15
1.10.1. 症状：删除集群后命名空间被保留	15
1.10.2. 解决问题：删除集群后命名空间被保留	15
1.11. 导入集群时出现自动 AUTO-IMPORT-SECRET-EXISTS 错误	16
1.11.1. 症状：导入集群时出现 Auto import secret exists 错误	16
1.11.2. 解决问题：导入集群时出现 Auto import secret exists 错误	16
1.12. 集群状态从离线变为可用的故障排除	16
1.12.1. 症状：集群状态从离线变为可用	17
1.12.2. 解决问题：集群状态从离线变为可用	17
1.13. 集群在控制台中有待处理或失败状态的故障排除	17
1.13.1. 症状：集群在控制台中有待处理或失败状态	17
1.13.2. 鉴别问题：集群在控制台中显示待处理或失败状态	17
1.13.3. 解决问题：集群在控制台中显示待处理或失败状态	18
1.14. 应用程序 GIT 服务器连接故障排除	18
1.14.1. 症状：Git 服务器连接	18
1.14.2. 解决问题：Git 服务器连接	18
1.15. GRAFANA 故障排除	20
1.15.1. 症状：Grafana explorer 网关超时	20
1.15.2. 解决问题：配置 multicloud-console 路由	20
1.16. 未使用放置规则选择本地集群的故障排除	20
1.16.1. 症状：未选择对本地集群进行故障排除	21
1.16.2. 解决问题：未选择对本地集群进行故障排除	21
1.17. 对应用程序 KUBERNETES 部署版本进行故障排除	22
1.17.1. 症状：应用程序部署版本	22
1.17.2. 解决：应用程序部署版本	22
1.18. 独立订阅内存故障排除	22
1.18.1. 症状：独立订阅内存	23
1.18.2. 解决问题：独立订阅内存	23
1.19. 带有降级条件的 KLUSTRLET 故障排除	24
1.19.1. 症状：Klusterlet 处于降级状况	24
1.19.2. 鉴别问题：Klusterlet 处于降级状况	24
1.19.3. 解决问题：Klusterlet 处于降级状况	24
1.20. 受管集群中的 KLUSTRLET 应用程序管理器故障排除	25
1.20.1. 症状：受管集群中的 Klusterlet 应用程序管理器	25
1.20.2. 解决问题：受管集群中的 Klusterlet 应用程序管理器	25
1.21. OBJECT STORAGE 频道 SECRET 故障排除	26
1.21.1. 症状：对象存储频道 secret	26
1.21.2. 解决问题：对象存储频道 secret	26
1.22. 对可观察性功能进行故障排除	27
1.22.1. 症状：MultiClusterObservability 资源处于没有就绪的状态	27
1.22.2. 解决这个问题：MultiClusterObservability 资源处于没有就绪的状态	27
1.23. OPENSIFT 监控服务故障排除	27
1.23.1. 症状：OpenShift 监控服务未就绪	27
1.23.2. 解决问题：OpenShift 监控服务未就绪	28
1.24. 在 MANAGEDCLUSTER 资源中不正确的标签值	28
1.24.1. 症状：managedcluster 资源中有不正确的标签值	28
1.24.2. 解决问题：managedcluster 资源中有不正确的标签值	28
1.25. 搜索聚合器 POD 状态的故障排除	28

1.25.1. 症状 1：搜索聚合器 pod 处于 Not Ready 状态	28
1.25.2. 解决问题：搜索聚合器 pod 处于 Not Ready 状态	29
1.25.3. 症状 2：搜索 redisgraph pod 处于待处理状态	29
1.25.4. 解决问题：搜索处于待处理状态的 redisgraph pod	29
1.26. METRICS-COLLECTOR 故障排除	29
1.26.1. 症状：metrics-collector 无法验证 observability-client-ca-certificate	29
1.26.2. 解决问题：metrics-collector 无法验证 observability-client-ca-certificate	30

第 1 章 故障排除

在使用故障排除指南前，您可以运行 `oc adm must-gather` 命令来收集详情、日志和步骤来调试问题。如需了解更多详细信息，请参阅[运行 must-gather 命令进行故障排除](#)。

另外，查看您基于角色的访问。详情请参阅[基于角色的访问控制](#)。

1.1. 记录的故障排除

查看与 Red Hat Advanced Cluster Management for Kubernetes 故障排除相关的主题：

安装

与原始安装相关的任务，请查看[安装](#)。

- [对安装状态处于安装或待处理状态进行故障排除](#)
- [重新安装失败的故障排除](#)

集群管理

与原始集群管理相关的任务，请查看[管理集群](#)。

- [离线集群的故障排除](#)
- [对带有待处理导入状态的集群进行故障排除](#)
- [证书更改后导入的集群离线故障排除](#)
- [集群状态从离线变为可用的故障排除](#)
- [VMware vSphere 上创建集群的故障排除](#)
- [集群在控制台中带有待处理或失败状态的故障排除](#)
- [OpenShift Container Platform 版本 3.11 集群导入失败的故障排除](#)
- [带有降级条件的 Klusterlet 故障排除](#)
- [受管集群中的 Klusterlet 应用程序管理器故障排除](#)
- [Object storage 频道 secret 故障排除](#)
- [managedcluster 资源故障排除](#)
- [删除集群后命名空间会保留](#)
- [导入集群时出现自动 Auto-import-secret-exists 错误](#)

应用程序管理

与原始应用程序管理相关的任务，查看[管理应用程序](#)。

- [对应用程序 Kubernetes 部署版本进行故障排除。](#)
- [对独立订阅内存问题进行故障排除。](#)

- [应用程序 Git 服务器连接故障排除](#)。
- [未选择本地集群的故障排除](#)

监管

要获得原始的安全指南，请查看[风险和合规](#)。

控制台可观察性

控制台观察功能包括搜索和 Visual Web Terminal，以及标头和导航功能。要查看原始的可观察指南，使用[控制台中的 Observability](#)。

- [grafana 故障排除](#)
- [对可观察性功能进行故障排除](#)
- [OpenShift 监控服务故障排除](#)
- [在 managedcluster 资源中不正确的标签值](#)
- [搜索聚合器 pod 状态的故障排除](#)
- [metrics-collector 故障排除](#)

1.2. 运行 MUST-GATHER 命令进行故障排除

要进行故障排除，参阅可以使用 **must-gather** 命令进行调试的用户情景信息，然后使用这个命令进行故障排除。

需要的访问权限：集群管理员

1.2.1. must-gather 情境

- **场景一：**如果您的问题已被记录，使用 [已记录的故障排除](#) 文档部分进行解决。这个指南按照产品的主要功能进行组织。
在这种情况下，您可以参阅本指南来查看您的问题的解决方案是否在文档中。例如，在创建集群时出现问题，您可能在这个指南的 [管理集群](#) 部分中找到解决方案。
- **情况 2：**如果这个指南中没有与您的问题相关的内容，运行 **must-gather** 命令并使用输出来调试问题。
- **情况 3：**无法使用 **must-gather** 命令的输出结果无法帮助解决您的问题，请向红帽支持提供您的输出。

1.2.2. must-gather 过程

请参阅以下流程来使用 **must-gather** 命令：

1. 了解 **must-gather** 命令以及按照 Red Hat OpenShift Container Platform 文档中的 [收集集群数据](#) 所需的先决条件。
2. 登录到您的集群。对于通常的用例，在登录到您的 *hub* 集群时运行 **must-gather** 命令。
备注：要检查您的受管集群，找到位于 **cluster-scoped-resources** 目录中的 **gather-managed.log** 文件：

```
<your-directory>/cluster-scoped-resources/gather-managed.log>
```

检查 JOINED 和 AVAILABLE 栏没有被设置为 **True** 的受管集群。您可以在这些没有以 **True** 状态连接的集群中运行 **must-gather** 命令。

3. 添加用于收集数据和目录的 Red Hat Advanced Cluster Management for Kubernetes 镜像。运行以下命令，在其中提供您要插入的镜像和输出目录：

```
oc adm must-gather --image=registry.redhat.io/rhacm2/acm-must-gather-rhel8:v2.3.0 --dest-dir=<directory>
```

4. 进入您指定的目录查看输出。输出以以下级别进行组织：
 - 两个对等级别：**cluster-scoped-resources** 和 **namespace** 资源。
 - 每个对等级别下的子类：用于 cluster-scope 和 namespace-scoped 资源的自定义资源定义的 API 组。
 - 每个子类的下一级：按 **kind** 进行排序的 YAML 文件。

1.2.3. 在断开连接的环境中的 must-gather

在断开连接的环境中，按照以下步骤运行 **must-gather** 命令：

1. 在断开连接的环境中，将 RedHat operator 目录镜像镜像（mirror）到其 mirror registry 中。如需更多信息，请参阅在[断开连接的网络中安装](#)。
2. 运行以下命令以提取日志，从其 mirror registry 中引用镜像：

```
REGISTRY=registry.example.com:5000
IMAGE=$REGISTRY/rhacm2/acm-must-gather-rhel8@sha256:ff9f37eb400dc1f7d07a9b6f2da9064992934b69847d17f59e385783c071b9d8
oc adm must-gather --image=$IMAGE --dest-dir=./data
```

1.3. 对安装状态处于安装或待处理状态进行故障排除

安装 Red Hat Advanced Cluster Management 时，**MultiClusterHub** 仍然处于 **安装** 阶段，或者多个 pod 处于 **Pending** 状态。

1.3.1. 症状：处于待处理状态

因为安装了 **MultiClusterHub** 以及 **MultiClusterHub** 资源报告 **ProgressDeadlineExceeded** 中的 **status.components** 字段中的一个或多个组件，所以需要超过十分钟。集群上的资源限制可能是问题。

检查命名空间中的安装 **Multiclusterhub** 的 pod。您可能会看到 **Pending** 状态类似如下：

```
reason: Unschedulable
message: '0/6 nodes are available: 3 Insufficient cpu, 3 node(s) had taint {node-role.kubernetes.io/master:
  }, that the pod didn't tolerate.'
```

在这种情况下，worker 节点资源不足以在集群中运行该产品。

1.3.2. 解决问题：调整 worker 节点大小

如果您有这个问题，则需要使用更大或更多 worker 节点更新集群。有关 [调整集群大小](#) 的指南，请参阅集群大小调整。

1.4. 重新安装失败的故障排除

在重新安装 Red Hat Advanced Cluster Management for Kubernetes 时，pod 没有启动。

1.4.1. 症状：重新安装失败

在安装 Advanced Cluster Management 后，如果 pod 没有启动，这很可能是因为以前已安装了 Red Hat Advanced Cluster Management，在您进行最新安装时以前安装的一些组件没有被删除。

在本例中，pod 在完成安装过程没有启动。

1.4.2. 解决问题：重新安装失败

如果您有这个问题，请完成以下步骤：

1. 根据[卸载](#)中介绍的步骤，执行卸载过程来删除当前的组件。
2. 按照[安装 Helm](#) 中的内容，安装 Helm CLI 二进制版本 3.2.0 或更新版本。
3. 确保您的 Red Hat OpenShift Container Platform CLI 被配置为运行 **oc** 命令。如需有关如何配置 **oc** 命令的更多信息，请参阅 OpenShift Container Platform 文档中的 [OpenShift CLI 入门](#)。
4. 将以下脚本复制到一个文件中：

```
#!/bin/bash
ACM_NAMESPACE=<namespace>
oc delete mch --all -n $ACM_NAMESPACE
helm ls --namespace $ACM_NAMESPACE | cut -f 1 | tail -n +2 | xargs -n 1 helm delete --
namespace $ACM_NAMESPACE
oc delete apiservice v1beta1.webhook.certmanager.k8s.io v1.admission.cluster.open-cluster-
management.io v1.admission.work.open-cluster-management.io
oc delete clusterimageset --all
oc delete configmap -n $ACM_NAMESPACE cert-manager-controller cert-manager-
cainjector-leader-election cert-manager-cainjector-leader-election-core
oc delete consolelink acm-console-link
oc delete crd klusterletaddonconfigs.agent.open-cluster-management.io
placementbindings.policy.open-cluster-management.io policies.policy.open-cluster-
management.io userpreferences.console.open-cluster-management.io
searchservices.search.acm.com
oc delete mutatingwebhookconfiguration cert-manager-webhook cert-manager-webhook-
v1alpha1 ocm-mutating-webhook managedclustermutators.admission.cluster.open-cluster-
management.io
oc delete oauthclient multicloudingress
oc delete rolebinding -n kube-system cert-manager-webhook-webhook-authentication-reader
oc delete scc kui-proxy-scc
oc delete validatingwebhookconfiguration cert-manager-webhook cert-manager-webhook-
v1alpha1 channels.apps.open.cluster.management.webhook.validator application-webhook-
validator multiclusterhub-operator-validating-webhook ocm-validating-webhook
```

将脚本中的 `<namespace>` 替换为安装 Red Hat Advanced Cluster Management 的命名空间的名称。确保指定正确的命名空间，因为命名空间会被清理和删除。

5. 运行该脚本以删除以前安装中的内容。
6. 运行安装。参阅[在线安装](#)

1.5. 离线集群的故障排除

一些常见的原因会导致集群显示离线状态。

1.5.1. 症状：集群状态为离线

完成创建集群的步骤后，您无法从 Red Hat Advanced Cluster Management 控制台访问集群，集群的状态为**离线 (offline)**。

1.5.2. 解决问题：集群状态为离线

1. 确定受管集群是否可用。您可以在 Red Hat Advanced Cluster Management 控制台的 *Clusters* 区域中进行检查。
如果不可用，请尝试重启受管集群。
2. 如果受管集群状态仍处于离线状态，完成以下步骤：
 - a. 在 hub 集群上运行 `oc get managedcluster <cluster_name> -o yaml` 命令。将 `<cluster_name>` 替换为集群的名称。
 - b. 找到 `status.conditionss` 部分。
 - c. 检查 `type: ManagedClusterConditionAvailable` 信息并解决相关的问题。

1.6. 对带有待处理导入状态的集群进行故障排除

如果在集群的控制台上持续接收到 *Pending import (待处理到)* 信息时，请按照以下步骤排除此问题。

1.6.1. 症状：集群处于待处理导入状态

在使用 Red Hat Advanced Cluster Management 控制台导入一个集群后，出现在控制台中的集群带有 *Pending import* 状态。

1.6.2. 鉴别问题：集群处于待处理导入状态

1. 在受管集群中运行以下命令查看有问题的 Kubernetes pod 的名称：

```
kubectl get pod -n open-cluster-management-agent | grep klusterlet-registration-agent
```

2. 在受管集群中运行以下命令查找错误的日志条目：

```
kubectl logs <registration_agent_pod> -n open-cluster-management-agent
```

把 `registration_agent_pod` 替换为在第 1 步中获得的 pod 名称。

3. 在返回的结果中搜索显示有网络连接问题的内容。示例包括：**no such host**。

1.6.3. 解决问题：集群处于待处理导入状态

1. 通过在 hub 集群中输入以下命令来检索有问题的端口号：

```
oc get infrastructure cluster -o yaml | grep apiServerURL
```

2. 确保来自受管集群的主机名可以被解析，并确保建立到主机和端口的出站连接。
如果无法通过受管集群建立通信，集群导入就不完整。受管集群的集群状态将会是 *Pending import*。

1.7. VMWARE VSPHERE 上创建集群的故障排除

如果您在 VMware vSphere 上创建 Red Hat OpenShift Container Platform 集群时遇到问题，请查看以下故障排除信息以查看它们是否解决了您的问题。

注：当集群创建过程在 VMware vSphere 上失败时，您将无法使用该链接来查看日志。如果发生这种情况，您可以通过查看 **thehive-controllers** pod 的日志来找出问题。**hive-controllers** 日志位于 **hive** 命名空间中。

1.7.1. 受管集群创建失败并显示证书 IP SAN 错误

1.7.1.1. 症状：Managed 集群创建失败并显示证书 IP SAN 错误

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，并显示一个错误消息，显示证书 IP SAN 错误。

1.7.1.2. 鉴别问题：管理的集群创建失败并显示证书 IP SAN 错误

受管集群的部署失败，并在部署日志中返回以下错误：

```
time="2020-08-07T15:27:55Z" level=error msg="Error: error setting up new vSphere SOAP client:
Post https://147.1.1.1/sdk: x509: cannot validate certificate for xx.xx.xx.xx because it doesn't contain
any IP SANs"
time="2020-08-07T15:27:55Z" level=error
```

1.7.1.3. 解决问题：管理的集群创建失败，并显示证书 IP SAN 错误

使用 VMware vCenter 服务器完全限定主机名，而不是凭证中的 IP 地址。您还可以更新 VMware vCenter CA 证书以包含 IP SAN。

1.7.2. 受管集群创建失败并显示未知证书颁发机构

1.7.2.1. 症状：管理集群创建失败并显示未知证书颁发机构

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，因为证书由未知颁发机构签名。

1.7.2.2. 鉴别问题：Managed 集群创建失败并显示未知证书颁发机构

受管集群的部署失败，并在部署日志中返回以下错误：

```
Error: error setting up new vSphere SOAP client: Post https://vspherehost.com/sdk: x509: certificate signed by unknown authority"
```

1.7.2.3. 解决问题：管理的集群创建失败并显示未知证书颁发机构

确保您在创建凭证时从证书认证机构输入了正确的证书。

1.7.3. 受管集群创建带有过期证书失败

1.7.3.1. 情况：集群创建失败并显示过期的证书

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，因为证书已过期或者无效。

1.7.3.2. 鉴别问题：管理的集群创建失败并显示过期的证书

受管集群的部署失败，并在部署日志中返回以下错误：

```
x509: certificate has expired or is not yet valid
```

1.7.3.3. 解决问题：管理的集群创建失败并显示过期的证书

确保同步了 ESXi 主机上的时间。

1.7.4. 受管集群创建失败且没有标记权限

1.7.4.1. 症状：管理集群创建失败且没有足够特权进行标记

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，因为没有足够的权限进行标记。

1.7.4.2. 鉴别问题：Managed 集群创建会失败，没有足够权限进行标记

受管集群的部署失败，并在部署日志中返回以下错误：

```
time="2020-08-07T19:41:58Z" level=debug msg="vsphere_tag_category.category: Creating..."
time="2020-08-07T19:41:58Z" level=error
time="2020-08-07T19:41:58Z" level=error msg="Error: could not create category: POST
https://vspherehost.com/rest/com/vmware/cis/tagging/category: 403 Forbidden"
time="2020-08-07T19:41:58Z" level=error
time="2020-08-07T19:41:58Z" level=error msg=" on ../tmp/openshift-install-436877649/main.tf line
54, in resource \"vsphere_tag_category\" \"category\":"
time="2020-08-07T19:41:58Z" level=error msg=" 54: resource \"vsphere_tag_category\" \"category\"
{"
```

1.7.4.3. 解决问题：管理的集群创建没有足够权限进行标记

确保 VMware vCenter 所需的帐户权限正确。如需更多信息，请参阅[在安装过程中删除了镜像 registry](#)。

1.7.5. 受管集群创建失败并显示无效的 dnsVIP

1.7.5.1. 症状：受管集群创建失败并显示无效的 dnsVIP

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，因为存在无效的 dnsVIP。

1.7.5.2. 鉴别问题：Managed 集群创建失败并显示无效的 dnsVIP

如果您在尝试使用 VMware vSphere 部署新受管集群时看到以下消息，这是因为您有一个较老的 OpenShift Container Platform 发行版本镜像，它不支持 VMware Installer Provisioned Infrastructure (IPI)：

```
failed to fetch Master Machines: failed to load asset \\\"Install Config\\\": invalid \\\"install-config.yaml\\\" file: platform.vsphere.dnsVIP: Invalid value: \\\"\\\": \\\"\\\" is not a valid IP
```

1.7.5.3. 解决问题：受管集群创建失败并显示无效的 dnsVIP

从支持 VMware Installer Provisioned Infrastructure 的 OpenShift Container Platform 版本中选择一个发行镜像。

1.7.6. 受管集群创建带有不正确的网络类型失败

1.7.6.1. 症状：集群创建失败并显示不正确的网络类型

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，因为指定的网络类型不正确。

1.7.6.2. 鉴别问题：管理的集群创建失败并显示不正确的网络类型

如果您在尝试使用 VMware vSphere 部署新受管集群时看到以下消息，这是因为您有一个旧的 OpenShift Container Platform 镜像，它不支持 VMware Installer Provisioned Infrastructure (IPI)：

```
time="2020-08-11T14:31:38-04:00" level=debug msg="vsphereprivate_import_ova.import: Creating..."
time="2020-08-11T14:31:39-04:00" level=error
time="2020-08-11T14:31:39-04:00" level=error msg="Error: rpc error: code = Unavailable desc = transport is closing"
time="2020-08-11T14:31:39-04:00" level=error
time="2020-08-11T14:31:39-04:00" level=error
time="2020-08-11T14:31:39-04:00" level=fatal msg="failed to fetch Cluster: failed to generate asset \\\"Cluster\\\": failed to create cluster: failed to apply Terraform: failed to complete the change"
```

1.7.6.3. 解决问题：受管集群创建失败并显示不正确的网络类型

为指定的 VMware 集群选择一个有效的 VMware vSphere 网络类型。

1.7.7. 受管集群创建失败并显示磁盘更改错误

1.7.7.1. 症状：因为错误处理磁盘更改导致添加 VMware vSphere 受管集群失败

在 VMware vSphere 上创建新的 Red Hat OpenShift Container Platform 集群后，集群会失败，因为在处理磁盘更改时会出现错误。

1.7.7.2. 鉴别问题：添加 VMware vSphere 受管集群会因为处理磁盘更改出错而失败

日志中会显示类似以下内容的消息：

```
ERROR
ERROR Error: error reconfiguring virtual machine: error processing disk changes post-clone: disk.0:
ServerFaultCode: NoPermission: RESOURCE (vm-71:2000), ACTION (queryAssociatedProfile):
RESOURCE (vm-71), ACTION (PolicyIDByVirtualDisk)
```

1.7.7.3. 解决问题：因为错误处理磁盘更改导致 VMware vSphere 受管集群失败

使用 VMware vSphere 客户端为用户授予 *Profile-driven Storage Privileges* 的所有权限。

1.8. OPENSIFT CONTAINER PLATFORM 版本 3.11 集群导入失败的故障排除

1.8.1. 症状：OpenShift Container Platform 版本 3.11 集群导入失败

试图导入 Red Hat OpenShift Container Platform 版本 3.11 集群后，导入会失败，并显示类似以下内容的日志消息：

```
customresourcedefinition.apiextensions.k8s.io/klusterlets.operator.open-cluster-management.io
configured
clusterrole.rbac.authorization.k8s.io/klusterlet configured
clusterrole.rbac.authorization.k8s.io/open-cluster-management:klusterlet-admin-aggregate-clusterrole
configured
clusterrolebinding.rbac.authorization.k8s.io/klusterlet configured
namespace/open-cluster-management-agent configured
secret/open-cluster-management-image-pull-credentials unchanged
serviceaccount/klusterlet configured
deployment.apps/klusterlet unchanged
klusterlet.operator.open-cluster-management.io/klusterlet configured
Error from server (BadRequest): error when creating "STDIN": Secret in version "v1" cannot be
handled as a Secret:
v1.Secret.ObjectMeta:
v1.ObjectMeta.TypeMeta: Kind: Data: decode base64: illegal base64 data at input byte 1313, error
found in #10 byte of ...|dhruy45="}, "kind": "..., bigger context
...|tye56u56u568yu07i67i67i67o556574i"}, "kind": "Secret", "metadata": {"annotations": {"kube|...
```

1.8.2. 鉴别问题：OpenShift Container Platform 版本 3.11 集群导入失败

这通常是因为安装的 **kubectl** 命令行工具的版本为 1.11 或更早版本。运行以下命令，以查看您正在运行的 **kubectl** 命令行工具的版本：

```
kubectl version
```

如果返回的数据列出了 1.11 或更早版本，按照 [解决问题：OpenShift Container Platform 版本 3.11 集群导入失败](#) 中的内容进行解决。

1.8.3. 解决问题：OpenShift Container Platform 版本 3.11 集群导入失败

您可以通过完成以下步骤之一解决这个问题：

- 安装 **kubectl** 命令行工具的最新版本。
 1. 下载 **kubectl** 工具的最新版本：参阅 Kubernetes 文档中的 [按照和设置 kubectl](#)。
 2. 升级 **kubectl** 工具后再次导入集群。
- 运行包含导入命令的文件。
 1. 根据[使用 CLI 导入受管集群](#)进行操作。
 2. 在创建用于导入集群的命令时，将该命令复制到名为 **import.yaml** 的 YAML 文件中。
 3. 运行以下命令从文件中再次导入集群：

```
oc apply -f import.yaml
```

1.9. 证书更改后导入的集群离线故障排除

虽然支持安装自定义的 **apiserver** 证书，但在更改证书前导入的一个或多个集群会处于 **offline** 状态。

1.9.1. 症状：证书更改后集群处于离线状态

完成更新证书 **secret** 的步骤后，在线的一个或多个集群现在会在 Red Hat Advanced Cluster Management for Kubernetes 控制台中显示 **offline** 状态。

1.9.2. 鉴别问题：证书更改后集群处于离线状态

更新自定义 API 服务器证书信息后，在新证书前导入并运行的集群会处于 **offline** 状态。

表示证书有问题的错误会出现在离线受管集群的 **open-cluster-management-agent** 命名空间中的 pod 日志中。以下示例与日志中显示的错误类似：

work-agent 的日志：

```
E0917 03:04:05.874759    1 manifestwork_controller.go:179] Reconcile work test-1-klusterlet-
addon-workmgr fails with err: Failed to update work status with err Get "https://api.aaa-
ocp.dev02.location.com:6443/apis/cluster.management.io/v1/namespaces/test-1/manifestworks/test-
1-klusterlet-addon-workmgr": x509: certificate signed by unknown authority
E0917 03:04:05.874887    1 base_controller.go:231] "ManifestWorkAgent" controller failed to sync
"test-1-klusterlet-addon-workmgr", err: Failed to update work status with err Get "api.aaa-
ocp.dev02.location.com:6443/apis/cluster.management.io/v1/namespaces/test-1/manifestworks/test-
1-klusterlet-addon-workmgr": x509: certificate signed by unknown authority
E0917 03:04:37.245859    1 reflector.go:127] k8s.io/client-go@v0.19.0/tools/cache/reflector.go:156:
Failed to watch *v1.ManifestWork: failed to list *v1.ManifestWork: Get "api.aaa-
ocp.dev02.location.com:6443/apis/cluster.management.io/v1/namespaces/test-1/manifestworks?
resourceVersion=607424": x509: certificate signed by unknown authority
```

registration-agent 的日志：

```
I0917 02:27:41.525026    1 event.go:282] Event(v1.ObjectReference{Kind:"Namespace",
Namespace:"open-cluster-management-agent", Name:"open-cluster-management-agent", UID:"",
APIVersion:"v1", ResourceVersion:"", FieldPath:""}): type: 'Normal' reason:
'ManagedClusterAvailableConditionUpdated' update managed cluster "test-1" available condition to
"True", due to "Managed cluster is available"
```

```
E0917 02:58:26.315984    1 reflector.go:127] k8s.io/client-go@v0.19.0/tools/cache/reflector.go:156:
Failed to watch *v1beta1.CertificateSigningRequest: Get "https://api.aaa-
ocp.dev02.location.com:6443/apis/cluster.management.io/v1/managedclusters?
allowWatchBookmarks=true&fieldSelector=metadata.name%3Dtest-
1&resourceVersion=607408&timeout=9m33s&timeoutSeconds=573&watch=true": x509: certificate
signed by unknown authority
E0917 02:58:26.598343    1 reflector.go:127] k8s.io/client-go@v0.19.0/tools/cache/reflector.go:156:
Failed to watch *v1.ManagedCluster: Get "https://api.aaa-
ocp.dev02.location.com:6443/apis/cluster.management.io/v1/managedclusters?
allowWatchBookmarks=true&fieldSelector=metadata.name%3Dtest-
1&resourceVersion=607408&timeout=9m33s&timeoutSeconds=573&watch=true": x509: certificate
signed by unknown authority
E0917 02:58:27.613963    1 reflector.go:127] k8s.io/client-go@v0.19.0/tools/cache/reflector.go:156:
Failed to watch *v1.ManagedCluster: failed to list *v1.ManagedCluster: Get "https://api.aaa-
ocp.dev02.location.com:6443/apis/cluster.management.io/v1/managedclusters?
allowWatchBookmarks=true&fieldSelector=metadata.name%3Dtest-
1&resourceVersion=607408&timeout=9m33s&timeoutSeconds=573&watch=true": x509: certificate
signed by unknown authority
```

1.9.3. 解决问题：证书更改后集群处于离线状态

要在更新证书信息后手动恢复集群，请为每个受管集群完成以下步骤：

1. 再次手动导入集群。从 Red Hat Advanced Cluster Management 创建的 Red Hat OpenShift Container Platform 集群将每 2 小时重新同步，以便可以为这些集群跳过这一步。

- a. 在 hub 集群中，输入以下命令显示导入命令：

```
oc get secret -n ${CLUSTER_NAME} ${CLUSTER_NAME}-import -
ojsonpath='{.data.import\.yaml}' | base64 --decode > import.yaml
```

将 `CLUSTER_NAME` 替换为您要导入的受管集群的名称。

- b. 在受管集群中应用 `import.yaml` 文件：

```
oc apply -f import.yaml
```

1.10. 删除集群后命名空间会保留

当您删除受管集群时，该命名空间通常会作为移除集群过程的一部分被删除。在个别情况下，命名空间会和其中的一些工件一起被保留。在这种情况下，您必须手动删除命名空间。

1.10.1. 症状：删除集群后命名空间被保留

删除受管集群后，命名空间没有被删除。

1.10.2. 解决问题：删除集群后命名空间被保留

完成以下步骤以手动删除命名空间：

1. 运行以下命令以生成保留在 `<cluster_name>` 命名空间中的资源列表：

```
oc api-resources --verbs=list --namespaced -o name | grep -E
'^secrets|^serviceaccounts|^managedclusteraddons|^roles|^rolebindings|^manifestworks|^lease:
```

```
| ^managedclusterinfo|^appliedmanifestworks' | xargs -n 1 oc get --show-kind --ignore-not-found -n <cluster_name>
```

使用您要删除的集群的命名空间名称替换 *cluster_name*。

2. 输入以下命令来编辑列表，删除列表中状态不是 **Delete** 的资源：

```
| oc edit <resource_kind> <resource_name> -n <namespace>
```

将 *resource_kind* 替换为资源类型。将 *resource_name* 替换为资源的名称。将 *namespace* 替换为资源的命名空间的名称。

3. 在元数据中找到 **finalizer** 属性。
4. 使用 vi 编辑器的 **dd** 命令删除非 Kubernetes finalizer。
5. 输入 **:wq** 命令保存列表并退出 vi 编辑器。
6. 输入以下命令来删除命名空间：

```
| oc delete ns <cluster-name>
```

将 *cluster-name* 替换为您要删除的命名空间的名称。

1.11. 导入集群时出现自动 **AUTO-IMPORT-SECRET-EXISTS** 错误

集群导入失败，并显示出错信息：auto import secret exists。

1.11.1. 症状：导入集群时出现 **Auto import secret exists** 错误

当导入 hive 集群以进行管理时，会显示 **auto-import-secret already exists** 错误。

1.11.2. 解决问题：导入集群时出现 **Auto import secret exists** 错误

当您试图导入之前由 Red Hat Advanced Cluster Management 管理的集群时，会出现这种情况。如果出现这种情况，当您尝试重新导入集群时，secret 会发生冲突。

要临时解决这个问题，请完成以下步骤：

1. 在 hub 集群中运行以下命令来手工删除存在的 **auto-import-secret**：

```
| oc delete secret auto-import-secret -n <cluster-namespace>
```

将 **cluster-namespace** 替换为集群的命名空间。

2. 按照[将目标受管集群导入到 hub 集群](#)的步骤再次导入集群。

集群已导入。

1.12. 集群状态从离线变为可用的故障排除

在没有对环境或集群进行任何手工更改的情况下，受管集群的状态在 **offline**（离线）和 **available**（可用）间转换。

1.12.1. 症状：集群状态从离线变为可用

当将受管集群连接到 hub 集群的网络不稳定时，hub 集群所报告的受管集群的状态在离线和可用之间不断转换。

1.12.2. 解决问题：集群状态从离线变为可用

要尝试解决这个问题，请完成以下步骤：

1. 输入以下命令在 hub 集群上编辑 **ManagedCluster** 规格：

```
oc edit managedcluster <cluster-name>
```

将 *cluster-name* 替换为您的受管集群的名称。

2. 在 **ManagedCluster** 规格中增加 **leaseDurationSeconds** 的值。默认值为 5 分钟，但可能没有足够的时间来保持与网络问题的连接。为租期指定较长的时间。例如，您可以将这个值提高为 20 分钟。

1.13. 集群在控制台中带有待处理或失败状态的故障排除

如果您在控制台中看到您创建的集群的状态为 *Pending* 或 *Failed*，请按照以下步骤排除问题。

1.13.1. 症状：集群在控制台中带有待处理或失败状态

在使用 Red Hat Advanced Cluster Management 控制台创建一个新集群后，在控制台中集群会一直显示 *Pending* 或 *Failed* 状态。

1.13.2. 鉴别问题：集群在控制台中显示待处理或失败状态

如果集群显示 *Failed* 状态，进入集群的详情页面并使用提供的日志的链接。如果没有找到日志或集群显示 *Pending* 状态，请按照以下步骤检查日志：

- 流程 1

1. 在 hub 集群中运行以下命令，查看在命名空间中为新集群创建的 Kubernetes pod 的名称：

```
oc get pod -n <new_cluster_name>
```

使用您创建的集群名称替换 *new_cluster_name*。

2. 如果没有 pod 在列出的名称中包括 **provision** 字符串，则按照流程 2 继续进行。如果存在其标题中带有 **provision** 字符串的 pod，则在 hub 集群中运行以下命令以查看该 pod 的日志：

```
oc logs <new_cluster_name_provision_pod_name> -n <new_cluster_name> -c hive
```

将 *new_cluster_name_provision_pod_name* 替换为您创建的集群的名称，后接包含 **provision** 的 pod 名称。

3. 搜索日志中可能会解释造成问题的原因的错误信息。

- 流程 2

如果没有在其名称中带有 **provision** 的 pod，则代表问题在进程早期发生。完成以下步骤以查看日志：

1. 在 hub 集群中运行以下命令：

```
oc describe clusterdeployments -n <new_cluster_name>
```

使用您创建的集群名称替换 **new_cluster_name**。如需有关集群安装日志的更多信息，请参阅 Red Hat OpenShift 文档中的 [收集安装日志](#) 的内容。

2. 检查是否在资源的 *Status.Conditions.Message* 和 *Status.Conditions.Reason* 条目中存在有关此问题的额外信息。

1.13.3. 解决问题：集群在控制台中显示待处理或失败状态

在日志中找到错误后，确定如何在销毁集群并重新创建它之前解决相关的错误。

以下示例包括了一个选择不支持的区的日志错误，以及解决它所需的操作：

```
No subnets provided for zones
```

When you created your cluster, you selected one or more zones within a region that are not supported.在重新创建集群时完成以下操作之一以解决此问题：

- 在区域里（region）选择不同的区（zone）。
- 如果列出了其它区，则省略不支持的区。
- 为集群选择不同的区域。

在处理了日志中记录的问题后，销毁集群并重新创建它。

如需了解更多与创建集群相关的信息，请参阅 [创建集群](#)。

1.14. 应用程序 GIT 服务器连接故障排除

来自 **open-cluster-management** 命名空间的日志中显示克隆 Git 仓库失败。

1.14.1. 症状：Git 服务器连接

来自 **open-cluster-management** 命令空间中的订阅控制器 pod **multicluster-operators-hub-subscription-*<random-characters>*** 的日志显示克隆 Git 仓库失败。您会看到一个 **x509: certificate signed by unknown authority** 错误，或 **BadGateway** 错误。

1.14.2. 解决问题：Git 服务器连接

重要：如果您使用以前的版本，请进行升级。

1. 保存 [apps.open-cluster-management.io_channels_crd.yaml](#)，使用相同的文件名。
2. 在 Red Hat Advanced Cluster Management 集群中，运行以下命令以应用该文件：

```
oc apply -f apps.open-cluster-management.io_channels_crd.yaml
```

3. 在 **open-cluster-management** 命名空间中，编辑 **advanced-cluster-management.v2.2.0** CSV，运行以下命令并编辑：

```
oc edit csv advanced-cluster-management.v2.2.0 -n open-cluster-management
```

查找以下容器：

- **multicluster-operators-standalone-subscription**
- **multicluster-operators-hub-subscription**

使用以下内容替换容器镜像：

```
quay.io/open-cluster-management/multicluster-operators-subscription:2.2-PR337-91af6cb37d427d22160b2c055589a4418dada4eb
```

更新会在 **open-cluster-management** 命名空间中重新创建以下 pod:

- **multicluster-operators-standalone-subscription-<random-characters>**
- **multicluster-operators-hub-subscription-<random-characters>**

4. 检查新 pod 是否使用新 docker 镜像运行。运行以下命令，然后查找新 docker 镜像：

```
oc get pod multicluster-operators-standalone-subscription-<random-characters> -n open-cluster-management -o yaml
oc get pod multicluster-operators-hub-subscription-<random-characters> -n open-cluster-management -o yaml
```

1. 更新受管集群上的镜像。

在 hub 集群中，用实际的受管集群名称替换 **CLUSTER_NAME** 来运行以下命令：

```
oc annotate klusterletaddonconfig -n CLUSTER_NAME CLUSTER_NAME
klusterletaddonconfig-pause=true --overwrite=true
```

2. 运行以下命令，将 **CLUSTER_NAME** 替换为实际受管集群名称：

```
oc edit manifestwork -n CLUSTER_NAME CLUSTER_NAME-klusterlet-addon-appmgr
```

3. 找到 **spec.global.imageOverrides.multicluster_operators_subscription** 并将值设置为：

```
quay.io/open-cluster-management/multicluster-operators-subscription:2.2-PR337-91af6cb37d427d22160b2c055589a4418dada4eb
```

这会在受管集群上的 **open-cluster-management-agent-addon** 命名空间中重新创建 **klusterlet-addon-appmgr-<random-characters>** pod。

4. 检查新 pod 是否使用新 docker 镜像运行。
5. 当您通过控制台或 CLI 创建应用程序时，手动在频道 spec 中添加 'insecureSkipVerify: true'。请参见以下示例：

```
apiVersion: apps.open-cluster-management.io/v1
kind: Channel
metadata:
labels:
name: sample-channel
namespace: sample
```

```
spec:
  type: GitHub
  pathname: <Git URL>
  insecureSkipVerify: true
```

1.15. GRAFANA 故障排除

当您在 Grafana explorer 中查询一些耗时的指标时，您可能会遇到一个**网关超时**错误。

1.15.1. 症状：Grafana explorer 网关超时

如果您在 Grafana explorer 中查询一些耗时的指标时遇到 **Gateway Time-out** 错误，则超时可能是由 **open-cluster-management** 命名空间中的 **multicloud-console** 路由造成的。

1.15.2. 解决问题：配置 *multicloud-console* 路由

如果您有这个问题，请完成以下步骤：

1. 验证 Grafana 的默认配置是否有预期的超时设置：

- a. 要验证 Grafana 的默认超时设置，请运行以下命令：

```
oc get secret grafana-config -n open-cluster-management-observability -o jsonpath="{.data.grafana\.ini}" | base64 -d | grep dataproxy -A 4
```

应显示以下超时设置：

```
[dataproxy]
timeout = 300
dial_timeout = 30
keep_alive_seconds = 300
```

- b. 要验证 Grafana 的默认数据源查询超时时间，请运行以下命令：

```
oc get secret/grafana-datasources -n open-cluster-management-observability -o jsonpath="{.data.datasources\.yaml}" | base64 -d | grep queryTimeout
```

应显示以下超时设置：

```
queryTimeout: 300s
```

2. 如果您验证了 Grafana 的默认配置有预期的超时设置，那么您可以运行以下命令在 **open-cluster-management** 命名空间中配置 **multicloud-console** 路由：

```
oc annotate route multicloud-console -n open-cluster-management --overwrite haproxy.router.openshift.io/timeout=300s
```

刷新 Grafana 页面，并尝试再次查询指标。**网关超时**错误不再显示。

1.16. 未使用放置规则选择本地集群的故障排除

受管集群使用放置规则选择，但不会选择 **local-cluster**（也被管理的 hubb 集群）。放置规则用户没有在 **local-cluster** 命名空间中创建可部署资源的权限。

1.16.1. 症状：未选择对本地集群进行故障排除

所有受管集群都使用放置规则选择，但 **local-cluster** 不是。放置规则用户没有在 **local-cluster** 命名空间中创建可部署资源的权限。

1.16.2. 解决问题：未选择对本地集群进行故障排除

要解决这个问题，您需要在 **local-cluster** 命名空间中授予可部署的管理权限。完成以下步骤：

1. 确认受管集群列表包含 **local-cluster**，且放置规则的 **decisions** 列表不显示本地集群。运行以下命令并查看结果：

```
% oc get managedclusters
```

```
NAME          HUB ACCEPTED  MANAGED CLUSTER URLS  JOINED  AVAILABLE
AGE
local-cluster true          True   True   56d
cluster1     true          True   True   16h
```

```
apiVersion: apps.open-cluster-management.io/v1
kind: PlacementRule
metadata:
  name: all-ready-clusters
  namespace: default
spec:
  clusterSelector: {}
status:
  decisions:
  - clusterName: cluster1
    clusterNamespace: cluster1
```

2. 在 **.yaml** 文件中创建一个 **Role**，以便在 **local-cluster** 命名空间中授予可部署资源的管理权限。请参见以下示例：

```
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: deployables-admin-user-zisis
  namespace: local-cluster
rules:
- apiGroups:
  - apps.open-cluster-management.io
  resources:
  - deployables
  verbs:
  - '*'
```

3. 创建 **RoleBinding** 资源，向放置规则用户授予 **local-cluster** 命名空间的访问权限。请参见以下示例：

```
apiVersion: rbac.authorization.k8s.io/v1
```

```

kind: RoleBinding
metadata:
  name: deployables-admin-user-zisis
  namespace: local-cluster
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: deployables-admin-user-zisis
  namespace: local-cluster
subjects:
- kind: User
  name: zisis
  apiGroup: rbac.authorization.k8s.io

```

1.17. 对应用程序 KUBERNETES 部署版本进行故障排除

可能不支持带有已弃用 Kubernetes **apiVersion** 的受管集群。有关已弃用 API 版本的详情，请参阅 [Kubernetes 问题](#)。

1.17.1. 症状：应用程序部署版本

如果 Subscription YAML 文件中的一个或多个应用程序资源使用弃用的 API，您可能会收到与以下类似的错误信息：

```

failed to install release: unable to build kubernetes objects from release manifest: unable to recognize
"": no matches for
kind "Deployment" in version "extensions/v1beta1"

```

或者，带有新 Kubernetes API 版本的 YAML 文件名为 **old.yaml**，您可能会收到以下错误：

```

error: unable to recognize "old.yaml": no matches for kind "Deployment" in version
"deployment/v1beta1"

```

1.17.2. 解决：应用程序部署版本

1. 更新资源中的 **apiVersion**。例如，如果 subscription YAML 文件中显示 *Deployment* kind 的错误，您需要将 **apiVersion** 从 **extensions/v1beta1** 更新至 **apps/v1**。请参见以下示例：

```

apiVersion: apps/v1
kind: Deployment

```

2. 在受管集群中运行以下命令来验证可用版本：

```

kubectl explain <resource>

```

3. 检查 **VERSION**。

1.18. 独立订阅内存故障排除

multicluster-operators-standalone-subscription pod 会因为内存问题而定期重启。

1.18.1. 症状：独立订阅内存

当 Operator Lifecycle Manager (OLM) 部署所有 operator 而不仅仅是 multicluster-subscription-operator 时，**multicluster-operators-standalone-subscription** pod 会重启，因为没有为独立订阅容器分配足够内存。

multicluster-operators-standalone-subscription pod 的内存限值在 multicluster subscription community operator CSV 中增加到 2GB，但此资源限制设置会被 OLM 忽略。

1.18.2. 解决问题：独立订阅内存

1. 安装后，找到订阅 multicluster subscription community operator 的 operator 订阅 CR。运行以下命令：

```
% oc get sub -n open-cluster-management acm-operator-subscription
```

2. 编辑 operator 订阅自定义资源，添加 **spec.config.resources.yaml** 文件，以定义资源限值。
注：不要创建新的、订阅了同一个多集群订阅社区 operator 的订阅自定义资源。因为两个 operator 订阅都连接到一个 operator，operator Pod 会被 **"killed"** 并由两个 operator 订阅自定义资源重启。

请参阅以下更新的 **.yaml** 文件示例：

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: multicluster-operators-subscription-alpha-community-operators-openshift-
  marketplace
  namespace: open-cluster-management
spec:
  channel: release-2.2
  config:
    resources:
      limits:
        cpu: 750m
        memory: 2Gi
      requests:
        cpu: 150m
        memory: 128Mi
  installPlanApproval: Automatic
  name: multicluster-operators-subscription
  source: community-operators
  sourceNamespace: openshift-marketplace
```

3. 保存资源后，确保独立订阅 Pod 被重启为有 2GB 内存限制。运行以下命令：

```
% oc get pods -n open-cluster-management multicluster-operators-standalone-subscription-7c8cbf885f-c94kz -o yaml
```

```
apiVersion: v1
kind: Pod
...
spec:
  containers:
```

```

- image: quay.io/open-cluster-management/multicluster-operators-subscription:community-
2.2
...
resources:
  limits:
    cpu: 750m
    memory: 2Gi
  requests:
    cpu: 150m
    memory: 128Mi
...
status:
  qosClass: Burstable

```

1.19. 带有降级条件的 KLUSTRERLET 故障排除

Klusterlet 降级条件可帮助诊断受管集群中的 Klusterlet 代理的状态。如果 Klusterlet 处于 degraded 条件，受管集群中的 Klusterlet 代理可能会出错，需要进行故障排除。对于设置为 **True** 的 Klusterlet 降级条件，请参见以下信息。

1.19.1. 症状：Klusterlet 处于降级状况

在受管集群中部署 Klusterlet 后，**KlusterletRegistrationDegraded** 或 **KlusterletWorkDegraded** 条件会显示 *True* 的状态。

1.19.2. 鉴别问题：Klusterlet 处于降级状况

1. 在受管集群中运行以下命令查看 Klusterlet 状态：

```
kubectl get klusterlets klusterlet -oyaml
```

2. 检查 **KlusterletRegistrationDegraded** 或 **KlusterletWorkDegraded** 以查看该条件是否被设置为 **True**。请根据 [解决这个问题](#) 的内容处理降级问题。

1.19.3. 解决问题：Klusterlet 处于降级状况

请查看以下降级状态列表，以及如何尝试解决这些问题：

- 如果 **KlusterletRegistrationDegraded** 条件的状态为 *True* 且状况原因为：*BootstrapSecretMissing*，您需要在 **open-cluster-management-agent** 命名空间中创建一个 bootstrap secret。
- 如果 **KlusterletRegistrationDegraded** 条件显示为 *True*，且状况原因为 *BootstrapSecretError* 或 *BootstrapSecretUnauthorized*，则当前的 bootstrap secret 无效。删除当前的 bootstrap secret，并在 **open-cluster-management-agent** 命名空间中重新创建有效的 bootstrap secret。
- 如果 **KlusterletRegistrationDegraded** 和 **KlusterletWorkDegraded** 显示为 *True*，且状况原因为 *HubKubeConfigSecretMissing*，请删除 Klusterlet 并重新创建它。
- 如果 **KlusterletRegistrationDegraded** 和 **KlusterletWorkDegraded** 显示为 *True*，则状况原因为：*ClusterNameMissing*、*KubeConfigMissing*、*HubConfigSecretError* 或 *HubConfigSecretUnauthorized*，从 **open-cluster-management-agent** 命名空间中删除 hub 集群 kubeconfig secret。注册代理将再次引导以获取新的 hub 集群 kubecofnig secret。

- 如果 **KlusterletRegistrationDegraded** 显示 *True*，且状况原因为 *GetRegistrationDeploymentFailed* 或 *UnavailableRegistrationPod*，您可以检查条件信息以获取问题详情并尝试解决。
- 如果 **KlusterletWorkDegraded** 显示为 *True*，且状况原因为 *GetWorkDeploymentFailed*，或 *UnavailableWorkPod*，您可以检查条件消息以获取问题详情并尝试解决。

1.20. 受管集群中的 KLUSTERLET 应用程序管理器故障排除

当您从 Red Hat Advanced Cluster Management for Kubernetes 升级时，Red Hat OpenShift Container Platform 受管集群版本 4.5 和 4.6 的 **klusterlet-addon-appmgr** pod 会被 **OOMKilled**。

1.20.1. 症状：受管集群中的 Klusterlet 应用程序管理器

您会收到 Red Hat OpenShift Container Platform 管理的集群版本 4.5 和 4.6 中的 **klusterlet-addon-appmgr** pod 的错误：**OOMKilled**。

1.20.2. 解决问题：受管集群中的 Klusterlet 应用程序管理器

对于 Red Hat Advanced Cluster Management for Kubernetes 2.1.x 和 2.2，您需要手动将 pod 的内存限值增加到 **8Gb**。请查看以下步骤。

1. 在 hub 集群中，注解 **klusterletaddonconfig** 来暂停复制。使用以下命令：

```
oc annotate klusterletaddonconfig -n ${CLUSTER_NAME} ${CLUSTER_NAME}
klusterletaddonconfig-pause=true -- overwrite=true
```

2. 在 hub 集群中，缩减 **klusterlet-addon-operator**。使用以下命令：

```
oc edit manifestwork ${CLUSTER_NAME}-klusterlet-addon-operator -n ${CLUSTER_NAME}
```

3. 找到 **klusterlet-addon-operator** 部署，将 **replicas: 0** 添加到 spec 以进行缩减。

```
- apiVersion: apps/v1
  kind: Deployment
  metadata:
    labels:
      app: cluster1
      name: klusterlet-addon-operator
      namespace: open-cluster-management-agent-addon
  spec:
    replicas: 0
```

在受管集群中，**open-cluster-management-agent-addon/klusterlet-addon-operator** pod 将被终止。

4. 登录到受管集群，手动增加 **appmgr** pod 中的内存限值。运行以下命令：

```
% oc edit deployments -n open-cluster-management-agent-addon klusterlet-addon-appmgr
```

例如，如果限制为 5G，将限制增加到 8G。

```
resources:
  limits:
    memory: 2Gi -> 8Gi
  requests:
    memory: 128Mi -> 256Mi
```

1.21. OBJECT STORAGE 频道 SECRET 故障排除

如果更改 **SecretAccessKey**，Object 存储频道的订阅将无法自动获取更新的 secret，您会收到一个错误。

1.21.1. 症状：对象存储频道 secret

Object 存储频道的订阅无法自动获取更新的 secret。这可能会阻止订阅 operator 协调并将资源从对象存储部署到受管集群的过程。

1.21.2. 解决问题：对象存储频道 secret

您需要手动输入凭证来创建 secret，然后引用频道中的 secret。

1. 注解订阅 CR，以便生成一个协调的单个订阅 operator。请参阅以下 **data** 规格：

```
apiVersion: apps.open-cluster-management.io/v1
kind: Channel
metadata:
  name: deva
  namespace: ch-obj
  labels:
    name: obj-sub
spec:
  type: ObjectBucket
  pathname: http://ec2-100-26-232-156.compute-1.amazonaws.com:9000/deva
  sourceNamespaces:
    - default
  secretRef:
    name: dev
---
apiVersion: v1
kind: Secret
metadata:
  name: dev
  namespace: ch-obj
  labels:
    name: obj-sub
data:
  AccessKeyID: YWRtaW4=
  SecretAccessKey: cGFzc3dvcmRhZG1pbG==
```

2. 运行 **oc annotate** 进行测试：

```
oc annotate appsub -n <subscription-namespace> <subscription-name> test=true
```

运行命令后，您可以进入 Application 控制台以验证资源是否已部署到受管集群。或者您可以登录到受管集群，以查看应用程序资源是否在给定命名空间中创建。

1.22. 对可观察性功能进行故障排除

安装可观察组件后，该组件可能会卡住，并显示 **Installing** 状态。

1.22.1. 症状：MultiClusterObservability 资源处于没有就绪的状态

如果在安装并创建 Observability 自定义资源定义 (CRD) 后，可观察性 (observability) 状态会一直处于 **Installing** 状态，则可能是因为没有为 **spec:storageConfig:storageClass** 参数定义值。或者，可观察性组件自动找到默认的 **storageClass**，但如果没有存储值，则组件会停留在 **Installing** 状态下。

1.22.2. 解决这个问题：MultiClusterObservability 资源处于没有就绪的状态

如果您有这个问题，请完成以下步骤：

1. 验证是否安装了可观察性组件：

- a. 要验证 **multicluster-observability-operator**，请运行以下命令：

```
kubectl get pods -n open-cluster-management|grep observability
```

- b. 要验证是否存在正确的 CRD，请运行以下命令：

```
kubectl get crd|grep observ
```

在启用组件前，必须显示以下 CRD:

```
multiclusterobservabilities.observability.open-cluster-management.io
observabilityaddons.observability.open-cluster-management.io
observatoria.core.observatorium.io
```

2. 如果您为裸机集群创建自己的 storageClass，请参阅 [如何在集群中或从集群中创建 NFS 置备程序](#)。
3. 为确保可观察性组件可以找到默认的 storageClass，更新 **multicluster-observability-operator** CRD 中的 **storageClass** 参数。您的参数可能类似以下值：

```
storageclass.kubernetes.io/is-default-class: "true"
```

安装完成后，可观察组件状态会被更新为 *Ready* 状态。如果安装无法完成，则会显示 *Fail* 状态。

1.23. OPENSIFT 监控服务故障排除

受管集群中的 Observability 服务需要从 OpenShift Container Platform 监控堆栈中提取指标数据。如果 OpenShift Container Platform 监控堆栈没有处于就绪状态，则不会安装 **metrics-collector**。

1.23.1. 症状：OpenShift 监控服务未就绪

endpoint-observability-operator-x pod 会检查 **prometheus-k8s** 服务是否在 **openshift-monitoring** 命名空间中可用。如果这个服务没有出现在 **openshift-monitoring** 命名空间中，则不会部署 **metrics-collector**。您可能会收到以下出错信息：**Failed to get prometheus resource**。

1.23.2. 解决问题：OpenShift 监控服务未就绪

如果您有这个问题，请完成以下步骤：

1. 登录您的 OpenShift Container Platform 集群。
2. 访问 **openshift-monitoring** 命名空间，验证 **prometheus-k8s** 服务是否可用。
3. 在受管集群的 **open-cluster-management-addon-observability** 命名空间中重启 **endpoint-observability-operator-x** pod。

1.24. 在 MANAGEDCLUSTER 资源中不正确的标签值

当您导入受管集群时，默认会安装可观察组件。您的放置规则可能类似以下信息：

```
status:
  decisions:
  - clusterName: sample-managed-cluster
    clusterNamespace: sample-managed-cluster
```

如果放置规则中没有包括受管集群，则不会安装可观察性组件。

1.24.1. 症状：managedcluster 资源中有不正确的标签值

如果发现导入的集群没有被包括，则可能会禁用受管集群资源的可观察性服务。

请记住：当启用该服务时，**vendor:OpenShift** 标签会添加来代表目标受管集群。只有在 OpenShift Container Platform 受管集群中才支持可观察性服务。

1.24.2. 解决问题：managedcluster 资源中有不正确的标签值

如果有这个问题，请为目标受管集群启用可观察性服务，并在 **managedcluster** 资源中更新标签。

完成以下步骤：

1. 登录您的 Red Hat Advanced Cluster Management 集群。
2. 通过更新放置规则将 **observability** 参数值改为 **enabled**。运行以下命令：

```
oc edit placementrule -n open-cluster-management-observability
```

3. 运行以下命令来验证 OpenShift 是否已被列为目标受管集群的厂商：

```
oc get managedcluster <CLUSTER NAME> -o yaml
```

将 **metadata.labels.vendor** 参数值更新为 **OpenShift**。

1.25. 搜索聚合器 POD 状态的故障排除

search-aggregator 无法运行。

1.25.1. 症状 1：搜索聚合器 pod 处于 Not Ready 状态

如果更新了 **redisgraph-user-secret**，搜索聚合器 Pod 处于 **Not Ready** 状态。您可能会收到以下错误：

```
E0113 15:04:42.427931    1 pool.go:93] Error authenticating Redis client. Original error: ERR invalid password
W0113 15:04:42.428100    1 healthProbes.go:36] Unable to reach Redis.
E0113 15:04:44.265777    1 pool.go:93] Error authenticating Redis client. Original error: ERR invalid password
W0113 15:04:44.266003    1 healthProbes.go:36] Unable to reach Redis.
E0113 15:04:46.316869    1 pool.go:93] Error authenticating Redis client. Original error: ERR invalid password
W0113 15:04:46.317029    1 healthProbes.go:36] Unable to reach Redis.
```

1.25.2. 解决问题：搜索聚合器 pod 处于 Not Ready 状态

如果您遇到这个问题，删除 **search-aggregator** 和 **search-api** pod 以重启 pod。运行以下命令以删除前面提到的 pod。

```
oc delete pod -n open-cluster-management <search-aggregator>
oc delete pod -n open-cluster-management <search-api>
```

1.25.3. 症状 2：搜索 redisgraph pod 处于待处理状态

当它处于 **Pending** 状态时，**search-redisgraph** pod 无法运行。

1.25.4. 解决问题：搜索处于待处理状态的 redisgraph pod

如果您有这个问题，请完成以下步骤：

1. 使用以下命令检查 hub 集群命名空间中的 pod 事件：

```
oc describe pod search-redisgraph-0
```

2. 如果已创建了 **searchcustomization** CR，请检查存储类和存储大小是否有效，并检查是否创建了 PVC。运行以下命令列出 PVC：

```
oc get pvc <storageclassname>-search-redisgraph-0
```

3. 确保 PVC 可以绑定到 **search-redisgraph-0** pod。如果问题仍没有解决，删除 StatefulSet **search-redisgraph**。search operator 会重新创建 StatefulSet。运行以下命令：

```
oc delete statefulset -n open-cluster-management search-redisgraph
```

1.26. METRICS-COLLECTOR 故障排除

当 **observability-client-ca-certificate** secret 没有在受管集群中被重新刷新时，您可能会收到一个内部服务器错误。

1.26.1. 症状：metrics-collector 无法验证 observability-client-ca-certificate

可能有一个受管集群，其中的指标不可用。如果出现这种情况，您可能会从 **metrics-collector** 部署中收到以下错误：

error: response status code is 500 Internal Server Error, response body is x509: certificate signed by unknown authority (possibly because of "crypto/rsa: verification error" while trying to verify candidate authority certificate "observability-client-ca-certificate")

1.26.2. 解决问题：metrics-collector 无法验证 observability-client-ca-certificate

如果您有这个问题，请完成以下步骤：

1. 登录到受管集群。
2. 删除名为 **observability-controller-open-cluster-management.io-observability-signer-client-cert** 的 secret，该 secret 位于 **open-cluster-management-addon-observability** 命名空间中。运行以下命令：

```
oc delete observability-controller-open-cluster-management.io-observability-signer-client-cert -n open-cluster-management-addon-observability
```

注： **observability-controller-open-cluster-management.io-observability-signer-client-cert** 会自动使用新证书重新创建。

重新创建 **metrics-collector** 部署并更新 **observability-controller-open-cluster-management.io-observability-signer-client-cert** secret。