

# 容器安全服务

## 故障处理



腾讯云

## 【 版权声明 】

©2013–2024 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分內容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

## 【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

## 【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

## 【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

## 文档目录

### 故障处理

Linux 客户端离线排查

集群接入排查

# 故障处理

## Linux 客户端离线排查

最近更新时间：2023-09-26 21:05:07

本文档将指导您进行 Linux 客户端离线排查，包括客户端进程未启动排查及网络故障排查。

### 说明

镜像安全扫描报离线提示时，需根据镜像名称/ID 定位关联主机后再进行客户端离线排查。

## 客户端进程未启动排查

1. 请查询容器安全服务进程是否存在。输入：`ps -ef|grep YD`。

- 正常状态下，容器安全服务存在两个进程，如下图所示：

```
[root@VM_145_42_centos ~]# ps -ef|grep YD
root      2890    2857    0 11:05 pts/0    00:00:00 grep YD
root      9059      1    0 Oct30 ?        00:00:41 /usr/local/qcloud/YunJing/YDEyes/YDService
root     14340      1    0 Oct23 ?        00:00:58 /usr/local/qcloud/YunJing/YDLive/YDLive
```

- 如果进程不存在，可能存在以下情况：

- 服务器未安装容器安全服务或者客户端已被卸载，请根据 [快速入门](#) 安装指引，进行客户端安装。
- 客户端可能出现异常冲突或者崩溃，导致进程没有启动。

2. 若服务器已安装容器安全服务或者客户端，可采用以下方法排查客户端进程未启动原因：

- 可查看客户端日志，存放路径：`/usr/local/qcloud/YunJing/log`。
- 可执行命令：`sh /usr/local/qcloud/YunJing/startYD.sh` 启动容器安全服务服务。

## 网络故障排查

如果进程存在，但容器安全服务不在线，大部分原因是网络不通，请按照以下操作进行排查：

1. 如果无法访问容器安全服务域名，可以尝试修改 DNS。可以通过执行如下命令行，排查容器安全服务域名是否可以访问：

- VPC 网络和黑石服务器环境：`telnet s.yd.tencentyun.com 5574`。  
正常情况下：返回如下图所示结果。

```
[root@VM_0_10_centos ~]# telnet s.yd.tencentyun.com 5574
Trying 169.254.0.55...
Connected to s.yd.tencentyun.com.
Escape character is '^]'.
```

如果无法访问：

- 1.1.1 可以尝试修改 `dns nameserver` 字段：`vim /etc/resolv.conf` `nameserver 183.60.83.19`

```
nameserver 183.60.82.98
```

1.1.2 修改完成后，重新执行 `telnet s.yd.tencentyun.com 5574` 检测能否连通。

```
[root@VM_0_7_centos ~]# cat /etc/resolv.conf
options timeout:1 rotate
; generated by /usr/sbin/dhclient-script
nameserver 183.60.83.19
nameserver 183.60.82.98
```

1.1.3 如果可以连通，等待几分钟后（时间长短根据网络情况而定），控制台将能看到对应服务器重新上线。

○ 基础网络环境（非 VPC 上的服务器）：`telnet s.yd.qcloud.com 5574`。

正常情况下：返回如下图所示结果。

```
[root@VM-28-45-centos ~]# telnet s.yd.qcloud.com 5574
Trying 10.53.78.111...
Connected to s.yd.qcloud.com.
Escape character is '^'.
```

**如果无法访问：**

1.1.1 可以尝试修改 `dns nameserver` 字段：`vim /etc/resolv.conf`，先把原有的 `nameserver` 字段注释，再新增 `nameserver` 字段，具体的 `nameserver ip` 相关内容，请参见 [内网服务](#)。

1.1.2 修改完成后，重新执行 `telnet s.yd.qcloud.com 5574` 检测能否连通。

1.1.3 如果可以连通，等待几分钟后（时间长短根据网络情况而定），控制台将能看到对应服务器重新上线。

2. 防火墙策略限制，需要在 Linux 客户端开放 TCP 端口：5574、8080、80、9080。

3. 如果容器安全服务进程存在，且不是由于网络原因导致的客户端离线，请打包客户端日志（日志路径：`/usr/local/qcloud/Yunjing/log`）并 [提交工单](#) 进行反馈。

# 集群接入排查

最近更新时间：2023-10-09 17:19:21

在集群接入后，系统将在该集群中创建名为 `tcss` 的命名空间。在 `tcss` 命名空间下，系统将安装名称为 `init-tcss-agent` 的 Job 类型工作负载，以及名称为 `tcss-asset` 的 Deployment 类型工作负载。在 `kube-system` 命名空间下，系统将安装名为 `yunjing-agent` 的 DaemonSet 类型工作负载。需确保以上三个工作负载都能正常运行。

## 使用控制台进行排查

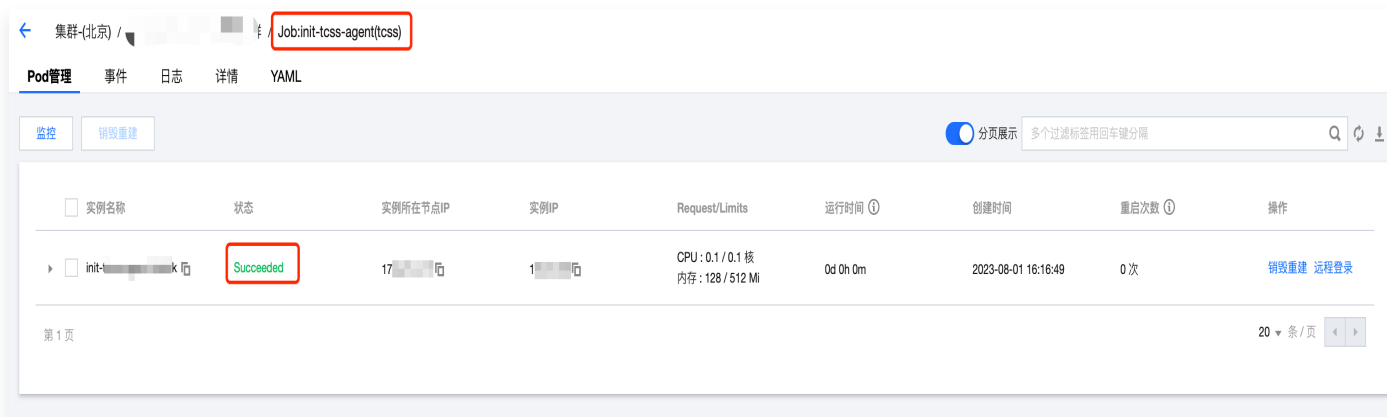
### Job 工作负载

检查 Job 工作负载下，名称为 `init-tcss-agent` 的 Pod 运行是否正常。

1. 登录 [容器服务控制台](#)，在左侧导航中，单击**集群**。
2. 在集群页面，单击**目标集群名称**，选择**工作负载 > Job**，搜索 `init-tcss-agent`。



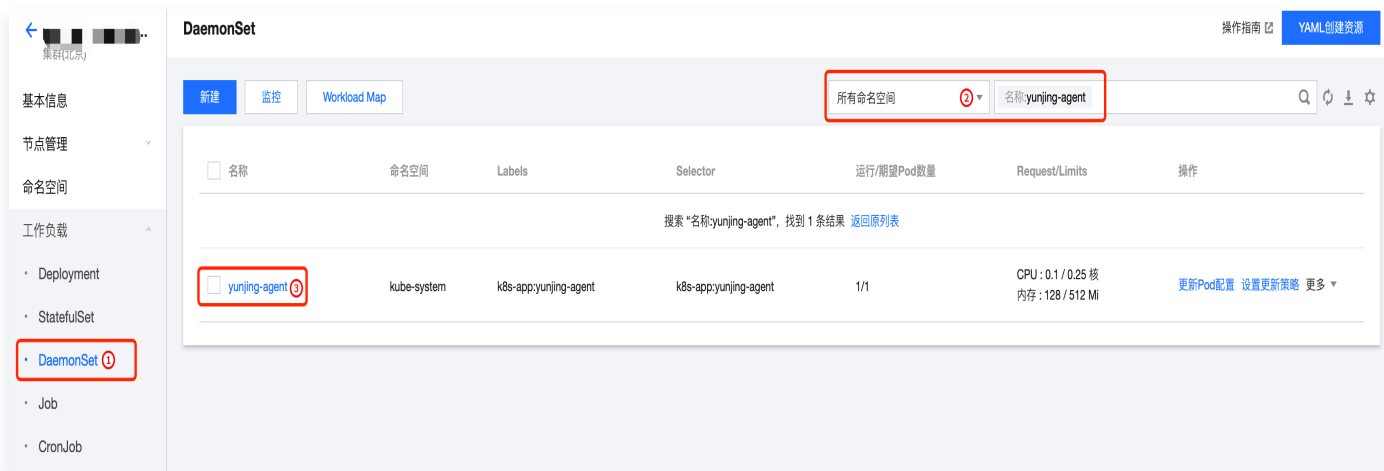
3. 单击**目标命名空间**，进入详情页面，状态列显示为 `Succeeded`，则表示名称为 `init-tcss-agent` 的 Pod 运行正常。



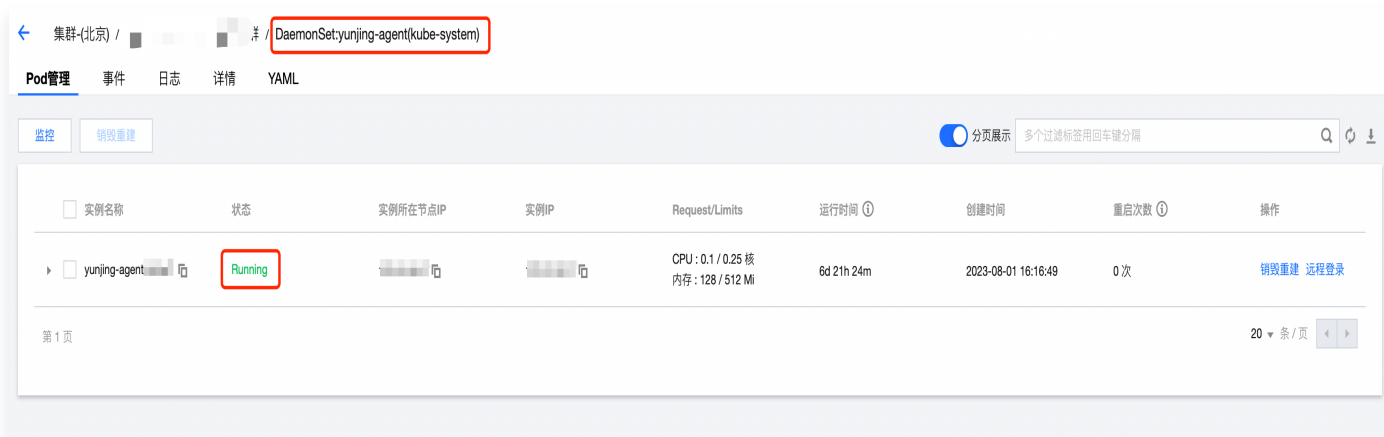
### DemonSet 工作负载

检查 DaemonSet 工作负载下，名称为 yunjing-agent 的 Pod 运行是否正常。

1. 登录 [容器服务控制台](#)，在左侧导航中，单击**集群**。
2. 在集群页面，单击**目标集群名称**，选择**工作负载 > DaemonSet**，搜索 `yunjing-agent`。



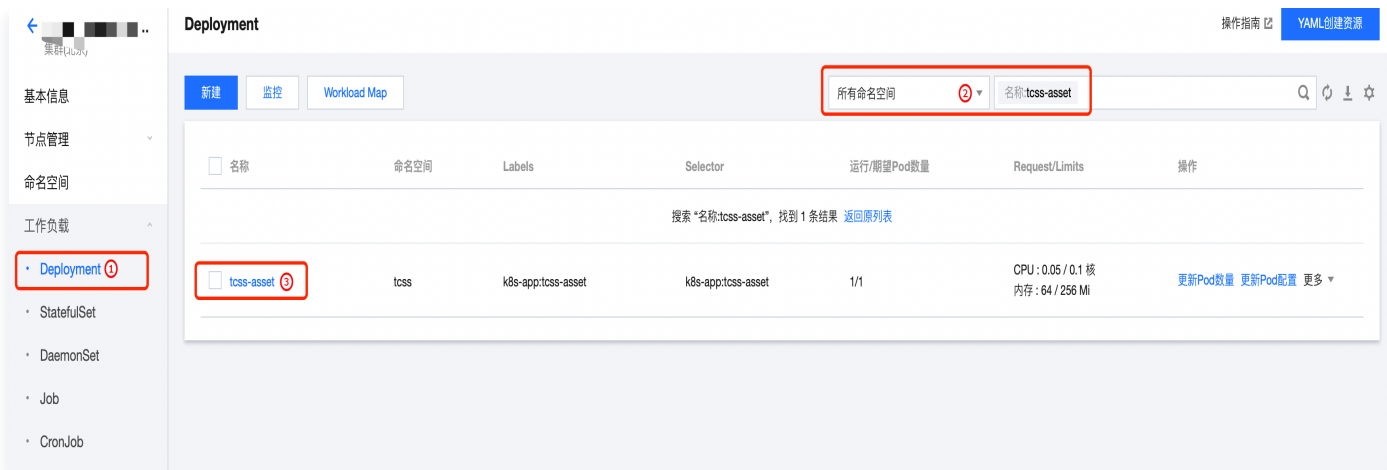
3. 单击**目标命名空间**，进入详情页面，状态列显示为 `Running`，则表示名称为 yunjing-agent 的 Pod 运行正常。



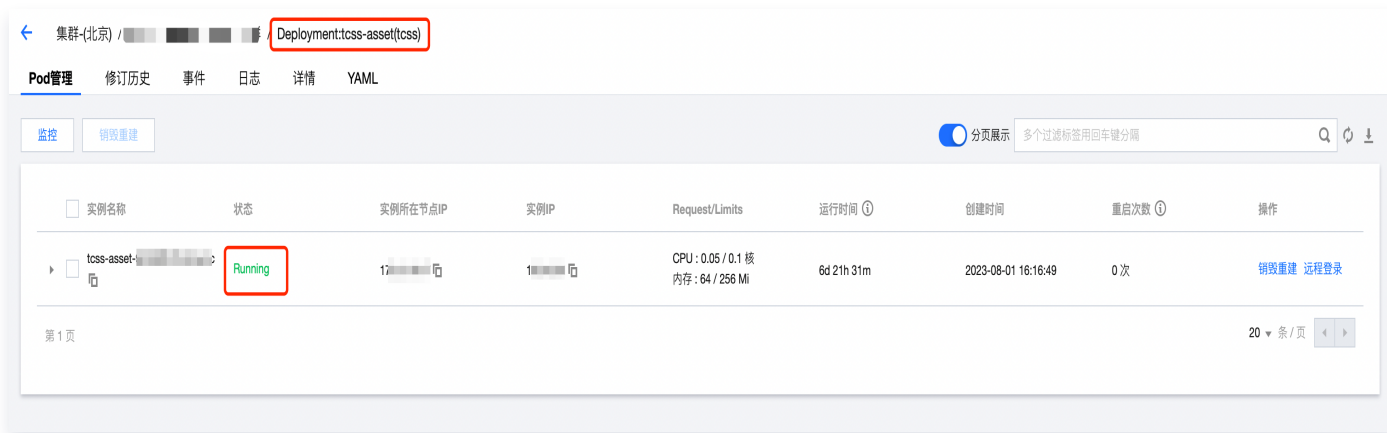
## Deployment 工作负载

检查 Deployment 工作负载下，名称为 tcss-asset 的 Pod 运行是否正常。

1. 登录 [容器服务控制台](#)，在左侧导航中，单击**集群**。
2. 在集群页面，单击**目标集群名称**，选择**工作负载 > Deployment**，搜索 `tcss-asset`。



3. 单击目标命名空间，进入详情页面，状态列显示为 `Running`，则表示名称为 `tcss-asset` 的 Pod 运行正常。



## 使用命令行进行排查

### Job 工作负载

1. 查看 Job 是否创建成功，执行命令：`kubectl get jobs -n tcss`。

```
[root@VM-0-17-tencentos ~]# kubectl get jobs -n tcss
NAME                COMPLETIONS  DURATION  AGE
init-tcss-agent    1/1          8s        9m27s
[root@VM-0-17-tencentos ~]#
```

2. 查看 Job 是否部署成功，执行命令：`kubectl get pods -n tcss | grep init-tcss-agent`。

```
[root@VM-0-17-tencentos ~]# kubectl get pods -n tcss | grep init-tcss-agent
init-tcss-agent-8jpkp    0/1    Completed    0    7m17s
[root@VM-0-17-tencentos ~]#
```

### DaemonSet 工作负载

1. 查看 DaemonSet 是否创建成功，执行命令：`kubectl get daemonset -A -l k8s-app=yunjing-agent`。



```
[root@VM-0-17-tencentos ~]# kubectl get daemonset -A -l k8s-app=yunjing-agent
NAMESPACE   NAME           DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
kube-system  yunjing-agent  1         1         1       1             1           <none>          30d
[root@VM-0-17-tencentos ~]#
```

2. 查看 DaemonSet 是否部署成功，执行命令：`kubectl get pods -A -l k8s-app=yunjing-agent`。

```
[root@VM-0-17-tencentos ~]# kubectl get pods -A -l k8s-app=yunjing-agent
NAMESPACE   NAME                               READY   STATUS    RESTARTS   AGE
kube-system  yunjing-agent-bl4w7              1/1     Running   0           30d
[root@VM-0-17-tencentos ~]#
```

## Deployment 工作负载

1. 查看 Deployment 是否创建成功，执行命令：`kubectl get deployment -n tcss`。

```
[root@VM-0-17-tencentos ~]# kubectl get deployment -n tcss
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
tcss-asset    1/1     1             1           15m
[root@VM-0-17-tencentos ~]#
```

2. 查看 Deployment 是否部署成功，执行命令：`kubectl get pods -n tcss | grep tcss-asset`。

```
[root@VM-0-17-tencentos ~]# kubectl get pods -n tcss | grep tcss-asset
tcss-asset-79c5c77756-zc5x8  1/1     Running   0           16m
[root@VM-0-17-tencentos ~]#
```