

# 当无法访问互联网时，请排除故障码头工人容器

## 目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[问题](#)

[解决方案](#)

## 简介

当无法访问互联网时，本文描述如何排除故障一个码头工人容器在代理服务器背后。

## 先决条件

### 要求

Cisco 建议您了解以下主题：

- Linux接口
- 虚拟机环境

### 使用的组件

本文档中的信息基于以下软件版本：

- CloudCenter版本4.x
- CloudCenter管弦乐队(CCO)

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

## 背景信息

如果您的企业要求代理对互联网的访问，您必须配置码头工人容器。

## 问题

当码头工人容器不能到达互联网时，这是再次产生问题的步骤。

当root用户设法运行在CCO的core\_installer.bin：

```
[root@localhost tmp]# ./core_installer.bin centos7 vmware cco
```

此错误将出现：

```
[root@localhost tmp]# ./core_installer.bin centos7 vmware cco
Verifying archive integrity... All good.
Uncompressing Core Installer V 4.5.2.....
.....
Installing Module: sysupdate
Installing Module: gateway
Installing Module: ntp
Installing Module: jdk8
Installing Module: tomcat8
Installing Module: gwtomcatapr
Installing Module: gwmongodb
Installing Module: docker
Failed in docker. Check /root/cliqr_modules.log for more info
[root@localhost tmp]#
```

## 解决方案

步骤1.运行core\_installer.bin文件以这些参数创建核心文件夹。

```
[root@localhost]# /core_installer.bin --noexec --keep
```

步骤2.导航到核心文件夹。

```
[root@localhost]# cd core
```

第三步：从核心文件夹，请运行setup.sh脚本安装码头工人。

```
[root@localhost core]# /setup.sh centos7 vmware docker
```

脚本失效与在码头工人“失败的”此错误。

步骤4.修改Dockerfile。

```
[root@localhost core]# vi docker/cliqr-container-worker/Dockerfile
```

步骤5.添加在Dockerfile的ENV部分的代理服务器信息。

```
ENV JAVA_VERSION 1.7.0
ENV http_proxy http://proxy.company.com
ENV https_proxy https://proxy.company.com
```

**注意：**用实际代理服务器地址替换proxy.company.com。

步骤6.创建码头工人服务的一个systemd下落在目录。

```
ENV JAVA_VERSION 1.7.0
ENV http_proxy http://proxy.company.com
ENV https_proxy https://proxy.company.com
```

步骤7.创建码头工人httpproxy.conf文件。

```
ENV JAVA_VERSION 1.7.0
ENV http_proxy http://proxy.company.com
ENV https_proxy https://proxy.company.com
```

步骤8.添加代理服务器信息。

```
ENV JAVA_VERSION 1.7.0
ENV http_proxy http://proxy.company.com
ENV https_proxy https://proxy.company.com
```

步骤 9 如果有与联系您需要，不用代理的内部码头工人注册，请添加他们在NO\_PROXY环境变量：

```
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
```

**注意：**用实际代理服务器地址替换proxy.company.com。

步骤10.保存配置文件并且重新加载码头工人服务。

```
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
```

步骤11.构件工作人员对最新的镜像在这些命令帮助下。

```
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
```

步骤 12重新启动码头工人服务。

```
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
```

步骤 13如果码头工人容器配置，请测试。

```
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
```

一旦码头工人容器配置，您将需要从core\_installer.bin的安装(如果安装CCO)。

步骤 14修改cliqr\_modules.conf。

```
Environment="HTTP_PROXY=http://proxy.company.com"
Environment="HTTPS_PROXY=https://proxy.company.com"
Environment="NO_PROXY=localhost,127.0.0.1,docker-registry.company.com"
```

步骤 15添加码头工人在文件结束时。这告诉core\_installer.bin码头工人安装。

```
sysupdate
gateway
ntp
jdk8
tomcat8
gwtomcatapr
gwmongodb
docker
```

步骤 16重新运行core\_installer.bin完成安装。

```
sysupdate
gateway
ntp
jdk8
```

tomcat8  
gwtomcatapr  
gwmongodb  
**docker**