

排除常见数据层(CDL)故障

目录

[1. 简介](#)

[2. 概述](#)

[3. 组成部分](#)

[4. 配置演练](#)

[5. 故障排除](#)

[5.1 Pod故障](#)

[5.2 CDL如何从会话密钥获取会话信息](#)

[5.3 CDL Pod未启用](#)

[5.4 Mirror Maker面板处于初始化状态](#)

[5.5 CDL索引复制不正确](#)

[5.6 CDL操作失败，但连接成功](#)

[5.7从CDL提前或推迟清除记录的通知](#)

[6. 警报](#)

[7. 最常见的问题](#)

[7.1 cdlReplicationError](#)

[7.2 cdlRemoteConnectionFailure & GRPC Connections Remote Site](#)

[8. 格拉法纳](#)

1. 简介

本文将介绍在SMF环境中对通用数据层(CDL)进行故障排除的基础知识。您可以在[此链接](#)上找到的[文档](#)。

2. 概述

思科通用数据层(CDL)是用于所有云本地应用的高性能下一代KV(Key-value)数据存储层。

CDL当前用作具有HA (高可用性) 和Geo HA功能的状态管理组件。

CDL提供：

- 跨不同网络功能(NF)的通用数据存储层。
- 低延迟读写 (在内存会话存储中)
- 当报告对同一会话的DoS (拒绝服务) 攻击时，通知NF阻止用户。
- 高可用性 — 至少具有2个副本的本地冗余。
- 具有2个站点的地理冗余。
- 没有主要/次要概念所有插槽均可用于写操作。由于不进行主选举，从而缩短故障切换时间。

3. 组成部分

- 终端 : (cdl-ep-session-c1-d0-7c79c87d65-xpm5v)
 - CDL终端是Kubernetes(K8s)POD。它被部署用于向NF客户端显示gRPC over HTTP2接口，用于处理数据库服务请求，并充当北向应用程序的入口点。
- 插槽 : (cdl-slot-session-c1-m1-0)
 - CDL终端支持多个插槽微服务。这些微服务是部署的K8s POD，用于向思科数据存储显示内部gRPC接口
 - 每个插槽POD具有有限数量的会话。这些会话是字节数组格式的实际会话数据
- 索引 : (cdl-index-session-c1-m1-0)
 - 索引微服务保存索引相关的数据
 - 然后，此索引数据用于从插槽微服务检索实际会话数据
- ETCD:(etcd-smf-etcd-cluster-0)
 - CDL使用ETCD (开源密钥值存储) 作为数据库服务发现。当Cisco Data Store EP启动、关闭或关闭时，会导致发布状态添加事件。因此，会向订阅这些事件的每个POD发送通知。此外，当添加或删除关键事件时，它会刷新本地映射。
- 卡夫卡 : (卡夫卡-0)
 - Kafka POD在本地副本之间和跨站点复制数据以编制索引。对于跨站点复制，Kafak使用MirrorMaker。
- Mirror Maker:(mirror-maker-0)
 - Mirror Maker POD将索引数据异地复制到远程CDL站点。它从远程站点获取数据并将其发布到本地Kafka站点，以便获取相应的索引实例。

示例 :

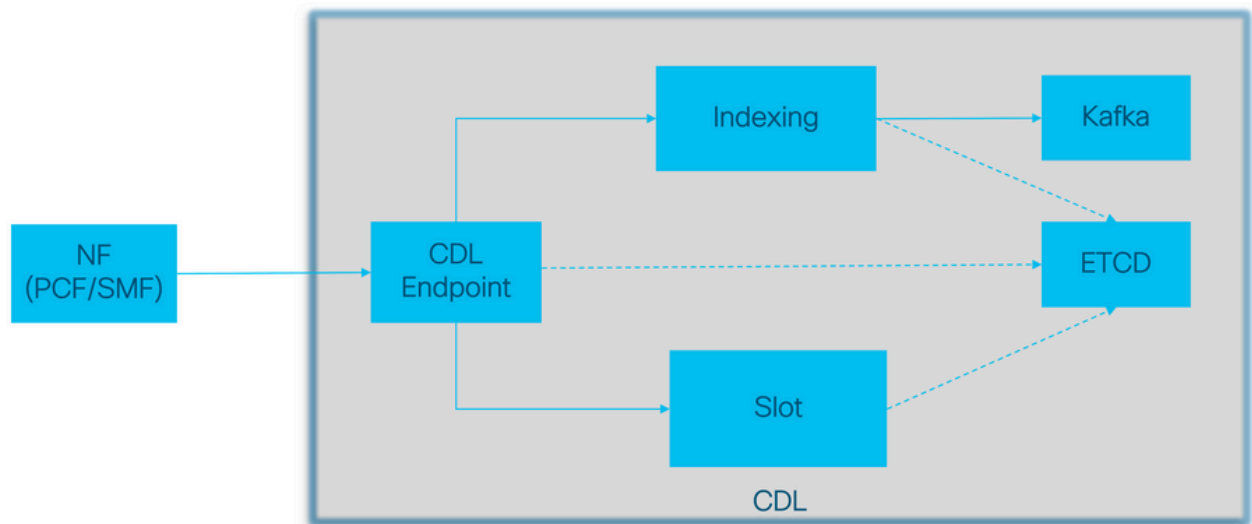
```

master-1:~$ kubectl get pods -n smf-smf -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP
cdl-ep-session-c1-d0-7889db4d87-5m1n5 1/1     Running   0           80d   192.168.
cdl-ep-session-c1-d0-7889db4d87-8q7hg 1/1     Running   0           80d   192.168.
cdl-ep-session-c1-d0-7889db4d87-fj2nf 1/1     Running   0           80d   192.168.
cdl-ep-session-c1-d0-7889db4d87-z6c2z 1/1     Running   0           34d   192.168.
cdl-ep-session-c1-d0-7889db4d87-z7c89 1/1     Running   0           80d   192.168.
cdl-index-session-c1-m1-0             1/1     Running   0           80d   192.168.
cdl-index-session-c1-m1-1             1/1     Running   0           80d   192.168.
cdl-index-session-c1-m2-0             1/1     Running   0           49d   192.168.
cdl-index-session-c1-m2-1             1/1     Running   0           80d   192.168.
cdl-index-session-c1-m3-0             1/1     Running   0           80d   192.168.
cdl-index-session-c1-m3-1             1/1     Running   0           80d   192.168.
cdl-index-session-c1-m4-0             1/1     Running   0           80d   192.168.
cdl-index-session-c1-m4-1             1/1     Running   0           49d   192.168.
cdl-slot-session-c1-m1-0              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m1-1              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m2-0              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m2-1              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m3-0              1/1     Running   0           34d   192.168.
cdl-slot-session-c1-m3-1              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m4-0              1/1     Running   0           34d   192.168.
cdl-slot-session-c1-m4-1              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m5-0              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m5-1              1/1     Running   0           34d   192.168.
cdl-slot-session-c1-m6-0              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m6-1              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m7-0              1/1     Running   0           80d   192.168.
cdl-slot-session-c1-m7-1              1/1     Running   0           49d   192.168.
cdl-slot-session-c1-m8-0              1/1     Running   0           49d   192.168.

```

cdl-slot-session-c1-m8-1	1/1	Running	0	80d	192.168.
etcd-smf-smf-etcd-cluster-0	2/2	Running	0	80d	192.168.11.17
etcd-smf-smf-etcd-cluster-1	2/2	Running	0	48d	192.168.7.59
etcd-smf-smf-etcd-cluster-2	2/2	Running	1	34d	192.168.11.66
georeplication-pod-0	1/1	Running	0	80d	10.10.1.
georeplication-pod-1	1/1	Running	0	48d	10.10.1.
grafana-dashboard-cdl-smf-smf-77bd69cff7-qbvmv	1/1	Running	0	34d	192.168.7.41
kafka-0	2/2	Running	0	80d	192.168.
kafka-1	2/2	Running	0	49d	192.168.
mirror-maker-0	1/1	Running	1	80d	192.168.
zookeeper-0	1/1	Running	0	34d	192.168.
zookeeper-1	1/1	Running	0	48d	192.168.
zookeeper-2	1/1	Running	0	48d	192.168.

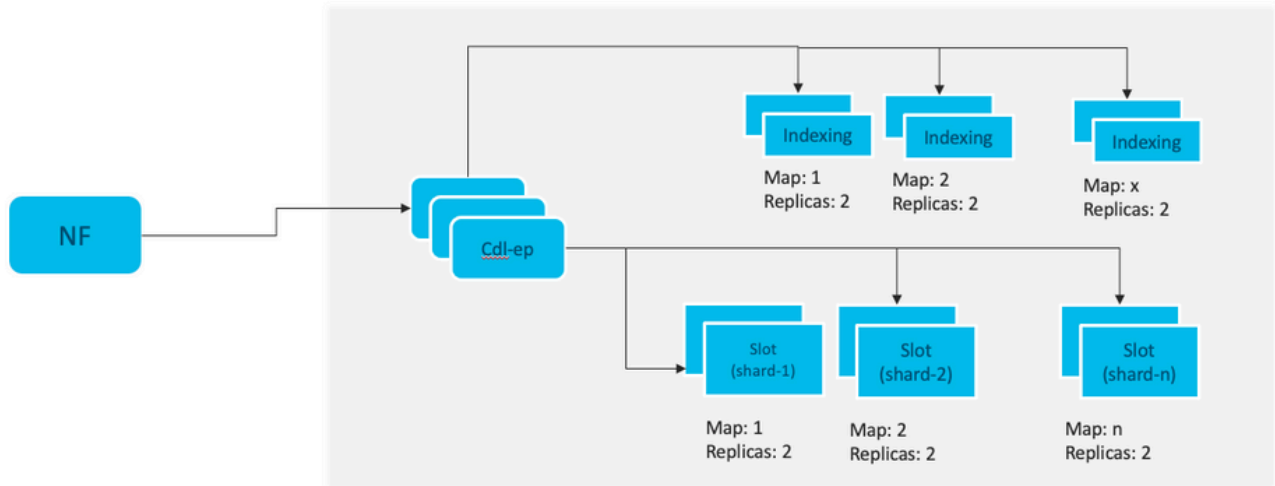
CDL Architecture



© 2017 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

CDL架构

CDL Architecture with primary components



x: Indexing maps /shards
n: Slot maps /shards
For HA purpose, each map should have 2 replicas minimum
Data distributed in multiple maps. Data in each map is different.
Each replica in a map has identical data for redundancy.

注意：没有主要/次要概念所有插槽均可用于写操作。由于不进行主选举，从而缩短故障切换时间。

注意：默认情况下，部署CDL时，为db-ep部署2个复制副本，1个插槽映射（每个映射2个复制副本）和1个索引映射（每个映射2个复制副本）。

4. 配置演练

```
smf# show running-config cdl
cdl system-id 1 // unique across the site, system-id 1 is the primary site ID for
cdl node-type db-data // node label to configure the node affinity
cdl enable-geo-replication true // CDL GR Deployment with 2 RACKS
cdl remote-site 2
db-endpoint host x.x.x.x // Remote site cdl-ep configuration on site-1
db-endpoint port 8882
kafka-server x.x.x.x 10061 // Remote site kafka configuration on site-1
exit
kafka-server x.x.x.x 10061
exit
exit
cdl label-config session // Configures the list of label for CDL pods
endpoint key smi.cisco.com/node-type-3
endpoint value session
slot map 1
key smi.cisco.com/node-type-3
value session
exit
slot map 2
```

```

key smi.cisco.com/node-type-3
value session
exit
slot map 3
key smi.cisco.com/node-type-3
value session
exit
slot map 4
key smi.cisco.com/node-type-3
value session
exit
slot map 5
key smi.cisco.com/node-type-3
value session
exit
slot map 6
key smi.cisco.com/node-type-3
value session
exit
slot map 7
key smi.cisco.com/node-type-3
value session
exit
slot map 8
key smi.cisco.com/node-type-3
value session
exit
index map 1
key smi.cisco.com/node-type-3
value session
exit
index map 2
key smi.cisco.com/node-type-3
value session
exit
index map 3
key smi.cisco.com/node-type-3
value session
exit
index map 4
key smi.cisco.com/node-type-3
value session
exit
exit
cdl datastore session          /// unique with in the site
label-config session
geo-remote-site [ 2 ]
slice-names [ SMF1 SMF2 ]
endpoint cpu-request 2000
endpoint go-max-procs 16
endpoint replica 5           /// number of cdl-ep pods
endpoint external-ip x.x.x.x
endpoint external-port 8882
index cpu-request 2000
index go-max-procs 8
index replica 2             /// number of replicas per mop for cdl-index, can not be changed a
NOTE: If you need to change number of index replica, set the system mode to shutdown from respective op
index map 4                 /// number of mops for cdl-index
index write-factor 1       /// number of copies to be written before a successful response
slot cpu-request 2000
slot go-max-procs 8
slot replica 2             /// number of replicas per mop for cdl-slot

```

```

slot map      8          /// number of mops for cd1-slot
slot write-factor 1
slot metrics report-idle-session-type true
features instance-aware-notification enable true      /// This enables GR failover notification
features instance-aware-notification system-id 1
  slice-names [ SMF1 ]
exit
features instance-aware-notification system-id 2
  slice-names [ SMF2 ]
exit
exit
cd1 kafka replica 2
cd1 kafka label-config key smi.cisco.com/node-type-3
cd1 kafka label-config value session
cd1 kafka external-ip x.x.x.x 10061
exit
cd1 kafka external-ip x.x.x.x 10061
exit

```

5.故障排除

5.1 Pod故障

CDL的操作是简单明了的Key > Value db。

- 所有请求都进入cdl-endpoint pod。
- 在cdl-index pods中，我们存储密钥，轮询。
- 在cdl-slot中，我们存储值（会话信息），轮询。
- 我们为每个Pod映射（类型）定义备份（副本数量）。
- Kafka pod被用作运输公车。
- 镜像生成器用作到不同机架的传输总线（Geo冗余）。

每个的故障都可以转换为，也就是说，如果此类型/映射的所有Pod同时关闭：

- cdl-endpoint — 与CDL通信时出错
- cdl-index — 丢失会话数据的密钥
- cdl-slot — 丢失会话数据
- Kafka — 丢失Pod类型映射之间的同步选项
- 镜像生成器 — 丢失与其他地理位置还原和节点的同步

我们始终可以从相关Pod收集日志，因为cdl Pod日志的滚动更新速度不会太快，因此收集这些日志具有额外的价值。

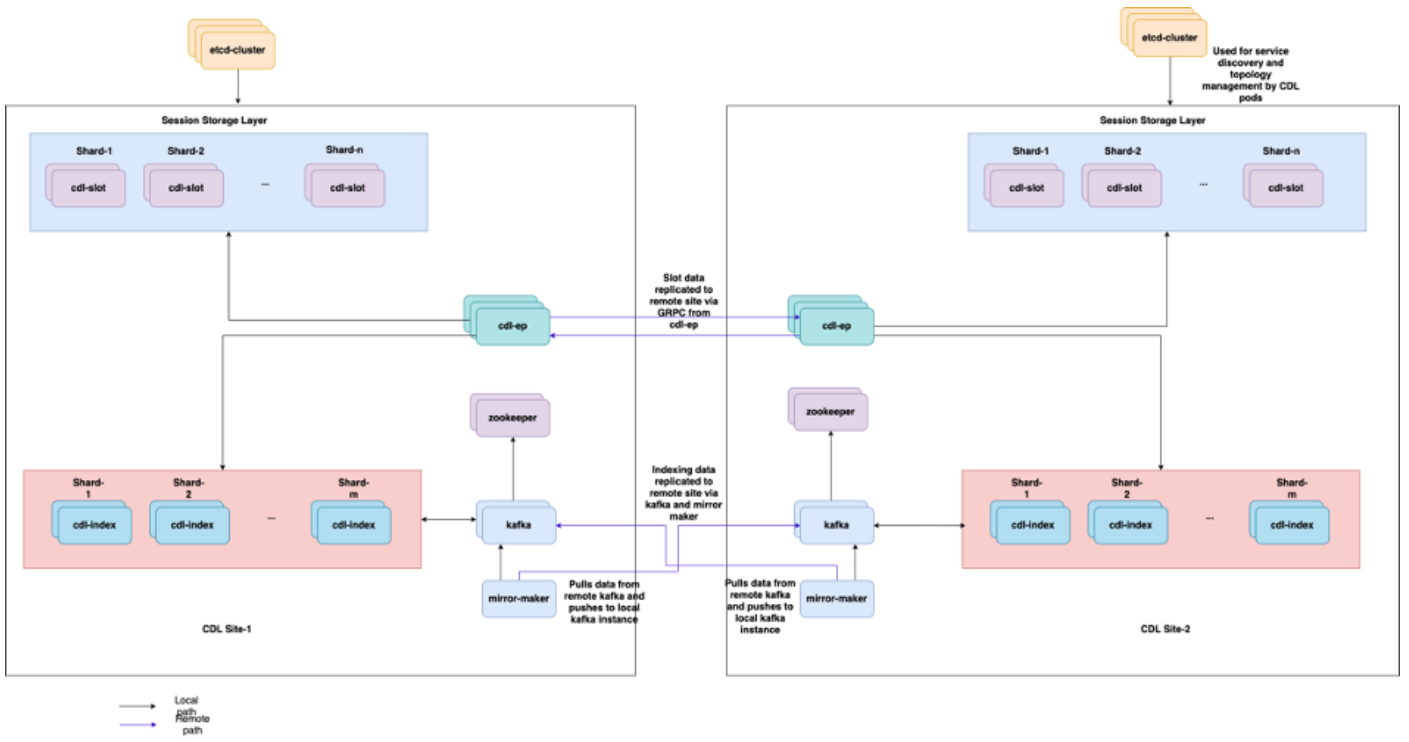
Remember tac-debug在注销时及时收集快照，并打印出所有数据，因为数据是存储的。

描述Pod

```
kubectl describe pod cd1-ep-session-c1-d0-7889db4d87-5m1n5 -n smf-rcdn
```

收集Pod日志

```
kubectl logs cdl-ep-session-c1-d0-7c79c87d65-xpm5v -n smf-rcdn
```



5.2 CDL如何从会话密钥获取会话信息

在CDL内部，每个会话都有一个名为unique-keys的字段，用于标识此会话。

如果比较show subscriber supi和cdl show sessions summary slice-name slice1 db-name session filter的会话打印输出

- 结合supi的ipv4会话地址= "1##imsi-123969789012404:10.0.0.3"
- ddn + ip4地址= "1##lab:10.0.0.3"
- 结合supi的ipv6会话地址= "1##imsi-123969789012404:2001:db0:0:2::"
- ddn + ipv6 address from session = "1##lab:2001:db0:0:2::"
- smfTeid还使用N4会话密钥= "1##293601283"在对UPF上的错误进行故障排除时，这非常有用，您可以搜索会话日志并查找与会话相关的信息。
- supi + ebi = "1##imsi-123969789012404:ebi-5"
- supi + ddn= "1##imsi-123969789012404:lab"

```
[smf/data] smf# cdl show sessions summary slice-name slice1 db-name session filter { condition match key
Sun Mar 19 20:17:41.914 UTC+00:00
message params: {session-summary cli session {0 100 1##/293601283 0 [{0 1##/293601283}] [] 0 0 false 40
session {
  primary-key 1##imsi-123969789012404:1
  unique-keys [ "1##imsi-123969789012404:10.0.0.3" "1##lab:10.0.0.3" "1##imsi-123969789012404:2001
  non-unique-keys [ "1##roaming-status:visitor-lbo" "1##ue-type:nr-capable" "1##supi:imsi-12396978
```

```

flags [ flag3:peerGtpuEpKey:10.10.10.150:20.0.0.1 session-state-flag:smf_active ]
map-id 2
instance-id 1
app-instance-id 1
version 1
create-time 2023-03-19 20:14:14.381940117 +0000 UTC
last-updated-time 2023-03-19 20:14:14.943366502 +0000 UTC
purge-on-eval false
next-eval-time 2023-03-26 20:14:14 +0000 UTC
session-types [ rat_type:NR wps:non_wps emergency_call:false pdu_type:ipv4v6 dnn:lab qos_5qi_1_rat_
data-size 2866
}
[smf/data] smf#

```

如果将其与SMF的打印输出进行比较：

```

[smf/data] smf# show subscriber supi imsi-123969789012404 gr-instance 1 namespace smf
Sun Mar 19 20:25:47.816 UTC+00:00
subscriber-details
{
  "subResponses": [
    [
      "roaming-status:visitor-lbo",
      "ue-type:nr-capable",
      "supi:imsi-123969789012404",
      "gpsi:msisdn-22331010101010",
      "pei:imei-123456789012381",
      "psid:1",
      "snssai:001000003",
      "dnn:lab",
      "emergency:false",
      "rat:nr",
      "access:3gpp access",
      "connectivity:5g",
      "udm-uecm:10.10.10.215",
      "udm-sdm:10.10.10.215",
      "auth-status:unauthenticated",
      "pcfGroupId:PCF-dnn=lab;",
      "policy:2",
      "pcf:10.10.10.216",
      "upf:10.10.10.150",
      "upfEpKey:10.10.10.150:10.10.10.202",
      "ipv4-addr:pool1/10.0.0.3",
      "ipv4-pool:pool1",
      "ipv4-range:pool1/10.0.0.1",
      "ipv4-startrange:pool1/10.0.0.1",
      "ipv6-pfx:pool1/2001:db0:0:2::",
      "ipv6-pool:pool1",
      "ipv6-range:pool1/2001:db0::",
      "ipv6-startrange:pool1/2001:db0::",
      "id-index:1:0:32768",
      "id-value:2/3",
      "chfGroupId:CHF-dnn=lab;",
      "chf:10.10.10.218",
      "amf:10.10.10.217",
      "peerGtpuEpKey:10.10.10.150:20.0.0.1",
      "namespace:smf",
    ]
  ]
}

```



```
    "nf-service:smf"  
  ]  
]  
}
```

检查SMF上的CDL状态：

```
cdl show status  
cdl show sessions summary slice-name <slice name> | more
```

5.3 CDL Pod未启用

如何识别

检查describe pod输出（容器/成员/状态/原因、事件）。

```
kubectl describe pods -n <namespace> <failed pod name>
```

如何修复

1. Pod处于挂起状态检查是否有任何标签值等于cdl/node-type复制副本数的k8s节点小于或等于标签值等于cdl/node-type值的k8s节点数

```
kubectl get nodes -l smi.cisco.com/node-type=<value of cdl/node-type, default value is 'session' i
```

2. Pod处于CrashLoopBackOff故障状态检查etcd pod状态。如果etcd pod未运行，请修复etcd问题。

```
kubectl describe pods -n <namespace> <etcd pod name>
```

3. Pod处于ImagePullBack故障状态。检查是否可以访问helm存储库和映像注册表。检查是否配置了所需的代理和dns服务器。

5.4 Mirror Maker面板处于初始化状态

检查描述Pod输出和Pod日志

```
kubectl describe pods -n <namespace> <failed pod name>  
kubectl logs -n <namespace> <failed pod name> [-c <container name>]
```

如何修复

- 检查为Kafka配置的外部IP是否正确
- 通过外部IP检查远程站点kafka的可用性

5.5 CDL索引复制不正确

如何识别

从一个站点添加的数据无法从其他站点访问。

如何修复

- 检查本地系统id配置和远程站点配置。
- 检查CDL终端和kafka在每个站点之间的可达性。
- 检查每个站点上的映射、索引副本和插槽。它可以在所有站点上完全相同。

5.6 CDL操作失败，但连接成功

如何修复

- 检查所有Pod是否处于就绪和运行状态。
- 只有当索引Pod与对等副本（本地或远程，如果可用）同步完成时，它们才处于就绪状态
- 仅当与对等副本（本地或远程，如果可用）同步完成时，插槽Pod才会处于就绪状态
- 如果至少一个插槽和一个索引Pod不可用，则终端处于NOT in ready状态。即使尚未就绪，也会从客户端接受grpc连接。

5.7从CDL提前或推迟清除记录的通知

如何修复

- 在k8s集群中，所有节点都可以进行时间同步
- 检查所有k8s节点上的NTP同步状态。如果存在任何问题，请解决此问题。

```
chronyc tracking
chronyc sources -v
chronyc sourcestats -v
```

6. 警报

警报	严重级	摘要
cdlLocalRequestFailure	关键	如果5分钟以上的本地请求成功率低于90%，则触发警报
cdlRemoteConnectionFailure	关键	如果从终端Pod到远程站点的活动连接超过0分钟时间（超过5分钟），则发出警报（仅适用于启用GR的系统）
cdlRemoteRequestFailure	关键	如果传入的远程请求成功率小于90%超过5分钟，则触发警报（仅适用于启用了GR的系统）
cdlReplicationError	关键	如果cdl-global命名空间中的传出复制请求与本地请求的比率在超过5分钟内低于90%。（仅适用于启用了GR的系统）。升级活动期间预期会出现这些警报，因此您可以忽略它们。
cdlKafkaRemoteReplicationDelay	关键	如果kafka到远程站点的复制延迟超过5分钟，则超过10秒就会发出警报（仅适用于启用了GR的系统）
cdlOverloaded — 重大	重大	如果CDL系统达到配置的容量百分比（默认值为80%），则系统触发警报（仅当启用过载保护功能时）
cdlOverloaded — 严重	关键	如果CDL系统达到配置的容量百分比（默认值为90%），则系统触发警报（仅当启用过载保护功能时）
cdlKafkaConnectionFailure	关键	如果CDL索引Pod与kafka断开连接超过5分钟

7.最常见的问题

7.1 cdlReplicationError

此警报通常出现在启动运营中心或系统升级期间，尝试为其找到CR，尝试检查CEE出现的警报是否已清除。

7.2 cdlRemoteConnectionFailure & GRPC_Connections_Remote_Site

该说明适用于所有“cdlRemoteConnectionFailure”和“GRPC_Connections_Remote_Site”警报。

对于cdlRemoteConnectionFailure警报：

从CDL终端日志中，我们看到从CDL终端Pod到远程主机的连接已丢失：

```
2022/01/20 01:36:18.852 [ERROR] [RemoteEndpointConnection.go:572] [datastore.ep.session] Connection to remote
```

我们可以看到CDL终端Pod尝试连接到远程服务器，但被远程主机拒绝：

```
2022/01/20 01:37:08.730 [WARN] [checksum.go:836] [datastore.checksum.session] Could not get checksum from remote
2022/01/20 01:37:08.732 [WARN] [checksum.go:836] [datastore.checksum.session] Could not get checksum from remote
2022/01/20 01:37:08.742 [WARN] [checksum.go:836] [datastore.checksum.session] Could not get checksum from remote
2022/01/20 01:37:08.742 [WARN] [checksum.go:836] [datastore.checksum.session] Could not get checksum from remote
2022/01/20 01:37:08.752 [WARN] [checksum.go:836] [datastore.checksum.session] Could not get checksum from remote
2022/01/20 01:37:08.754 [WARN] [checksum.go:836] [datastore.checksum.session] Could not get checksum from remote
```

由于远程主机在5分钟内仍无法访问，因此警报引发如下：

```
alerts history detail cdlRemoteConnectionFailure f5237c750de6
severity critical
type "Processing Error Alarm"
startsAt 2025-01-21T01:41:26.857Z
endsAt 2025-01-21T02:10:46.857Z
source cdl-ep-session-c1-d0-6d86f55945-pxfx9
summary "CDL endpoint connections from pod cdl-ep-session-c1-d0-6d86f55945-pxfx9 and namespace smf-rcdn-tee"
labels [ "alertname: cdlRemoteConnectionFailure" "cluster: smf-data-rcdn_tee" "monitor: prometheus" "namespace: smf-rcdn-tee" ]
annotations [ "summary: CDL endpoint connections from pod cdl-ep-session-c1-d0-6d86f55945-pxfx9 and namespace smf-rcdn-tee" ]
```

在02:10:32:

```
2022/01/20 02:10:32.702 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote endpoint closed
2022/01/20 02:10:38.923 [WARN] [RemoteEndpointConnection.go:563] [datastore.ep.session] CdI status changed
2022/01/20 02:10:38.927 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote endpoint closed
2022/01/20 02:10:38.934 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote endpoint closed
2022/01/20 02:10:38.934 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote endpoint closed
2022/01/20 02:10:38.935 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote endpoint closed
```

CDL远程站点的SMF中存在配置：

```
cdl remote-site 2
db-endpoint host 10.10.10.141
```

```
db-endpoint port 8882
kafka-server 10.10.19.139 10061
exit
kafka-server 10.10.10.140 10061
exit
exit
```

对于警报GRPC_Connections_Remote_Site:

同样的解释同样适用于“GRPC_Connections_Remote_Site”，因为它也来自同一个CDL终端Pod。

```
alerts history detail GRPC_Connections_Remote_Site f083cb9d9b8d
severity critical
type "Communications Alarm"
startsAt 2025-01-21T01:37:35.160Z
endsAt 2025-01-21T02:11:35.160Z
source cd1-ep-session-c1-d0-6d86f55945-pxfx9
summary "GRPC connections to remote site are not equal to 4"
Labels [ "alertname: GRPC_Connections_Remote_Site" "cluster: smf-data-rcdn_cee" "monitor: prometheus" ]
```

从CDL终端Pod日志，当与远程主机的连接被拒绝时，警报开始：

```
2022/01/20 01:36:18.852 [ERROR] [RemoteEndpointConnection.go:572] [datastore.ep.session] Connection to r
```

当成功连接到远程站点时，警报被清除：

```
2022/01/20 02:10:32.702 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote
2022/01/20 02:10:38.923 [WARN] [RemoteEndpointConnection.go:563] [datastore.ep.session] Cd1 status chang
2022/01/20 02:10:38.927 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote
2022/01/20 02:10:38.934 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote
2022/01/20 02:10:38.934 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote
2022/01/20 02:10:38.935 [WARN] [RemoteEndpointConnection.go:437] [datastore.ep.session] Stream to remote
```

8. 格拉法纳

CDL控制面板是每个SMF部署的一部分。

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。