

# 共通データ層(CDL)のトラブルシューティング

## 内容

### [1. 概要](#)

### [2. 概要](#)

### [3. コンポーネント](#)

### [4. 構成の段階的検証](#)

### [5. トラブルシューティング](#)

#### [5.1 ポッドの障害](#)

#### [5.2 CDLセッションキーからセッション情報を取得する方法](#)

#### [5.3 CDLポッドが起動していない](#)

#### [5.4 Mirror Makerポッドがinit状態にある](#)

#### [5.5 CDLインデックスが正しく複製されない](#)

#### [5.6 CDL操作は失敗するが、接続は成功する](#)

#### [5.7 レコードのパーシステンス通知がCDLから早く、または遅れた](#)

### [6. アラート](#)

### [7. 最も一般的な問題](#)

#### [7.1 cdlReplicationError](#)

#### [7.2 cdlRemoteConnectionFailureおよびGRPC Connections Remote Site](#)

#### [ハグラフィアナ](#)

## 1. 概要

この記事では、SMF環境でのCommon Data Layer(CDL)のトラブルシューティングの基本について説明します。この[リンク](#)からドキュメントを参照できます。

## 2. 概要

Cisco Common Data Layer(CDL)は、すべてのクラウドネイティブアプリケーション向けの高性能な次世代KV(Key-value)データストアレイヤです。

CDLは現在、HA (ハイアベイラビリティ) およびGeo HA機能を備えた状態管理コンポーネントとして使用されています。

CDLには次の機能があります。

- さまざまなネットワーク機能(NF)に共通のデータストア層。
- 低遅延の読み取りおよび書き込み (メモリセッションストレージ内)
- 同じセッションに対するDoS (サービス拒否) 攻撃が報告されたときに、サブスクリバをブロックするようにNFに通知します。
- 高可用性: 少なくとも2つのレプリカを使用するローカルの冗長性。
- 2サイトの地理的冗長性

- すべてのスロットで書き込み操作に使用できるプライマリ/セカンダリ/の概念はありません。プライマリの選出が行われないため、フェールオーバー時間が短縮されます。

### 3.コンポーネント

- エンドポイント : cdl-ep-session-c1-d0-7c79c87d65-xpm5v
  - CDLエンドポイントはKubernetes(K8s)PODです。これは、データベースサービス要求を処理するためにNFクライアントに向けてHTTP2インターフェイス経由でgRPCを公開するために展開され、ノースバウンドアプリケーションのエントリポイントとして機能します。
- スロット : (cdl-slot-session-c1-m1-0)
  - CDLエンドポイントは、複数のスロットマイクロサービスをサポートします。これらのマイクロサービスは、Cisco Data Storeに向けて内部gRPCインターフェイスを公開するために導入されたK8s PODです
  - 各スロットPODは、限定された数のセッションを保持します。これらのセッションは、バイト配列形式の実際のセッションデータです
- インデックス : (cdl-index-session-c1-m1-0)
  - インデックスマイクロサービスは、インデックス関連データを保持します
  - このインデックスデータは、スロットマイクロサービスから実際のセッションデータを取得するために使用されます
- ETCD: (etcd-smf-etcd-cluster-0)
  - CDLは、DBサービス検出としてETCD ( オープンソースのキー値ストア ) を使用します。Cisco Data Store EPが起動、強制終了、またはシャットダウンされると、パブリッシュ状態によってイベントが追加されます。したがって、通知は、これらのイベントに加入している各PODに送信されます。さらに、キーイベントが追加または削除されると、ローカルマップが更新されます。
- Kafka: (kafka-0)
  - Kafka PODは、ローカルレプリカ間およびサイト間でデータを複製してインデックスを作成します。サイト間のレプリケーションには、KafakはMirrorMakerを使用します
- ミラーメーカー : (mirror-maker-0)
  - Mirror Maker PODは、インデックス作成データをリモートCDLサイトにgeoレプリケートします。リモートサイトからデータを取得し、適切なインデックス作成インスタンスが取得できるようにローカルのKafkaサイトに公開します。

以下に例を挙げます。

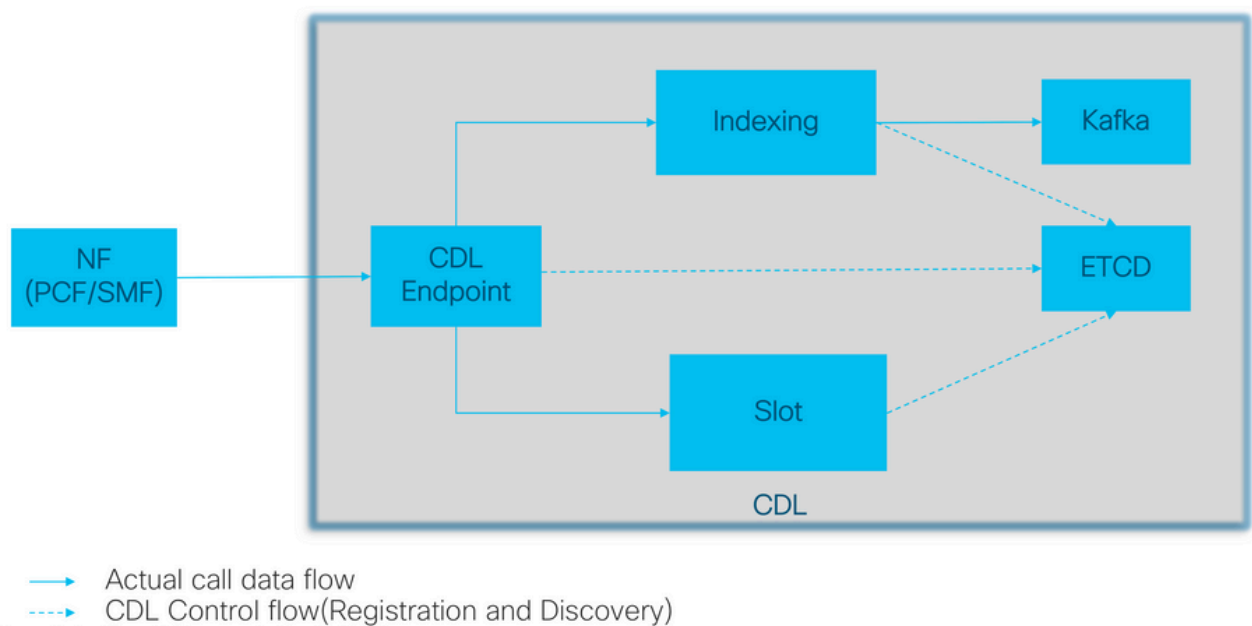
```

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.

```

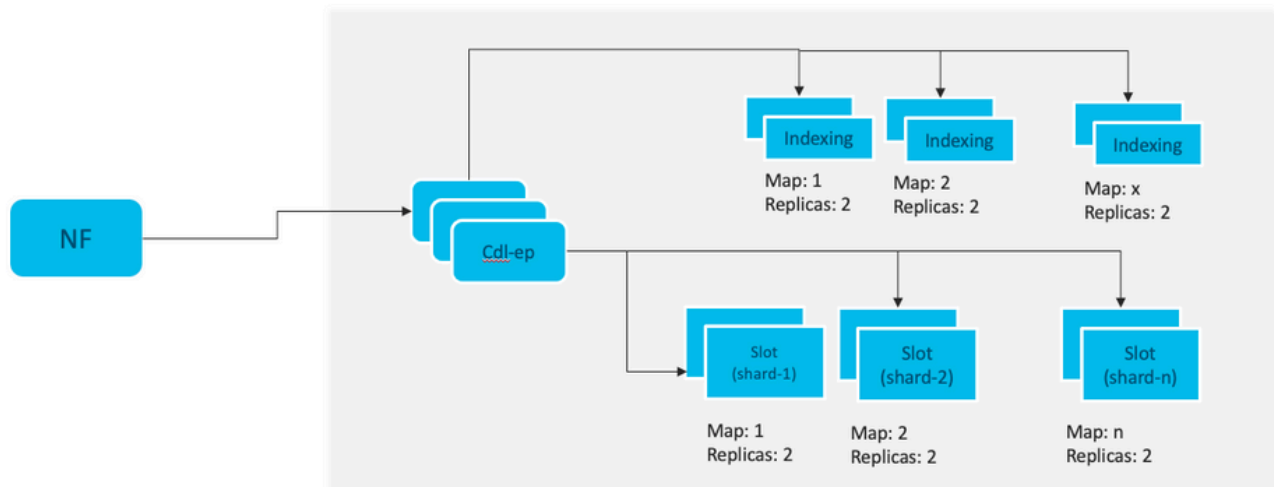
cdl-index-session-c1-m3-0	1/1	Running	0	80d	192.168.
cdl-index-session-c1-m3-1	1/1	Running	0	80d	192.168.
cdl-index-session-c1-m4-0	1/1	Running	0	80d	192.168.
cdl-index-session-c1-m4-1	1/1	Running	0	49d	192.168.
cdl-slot-session-c1-m1-0	1/1	Running	0	80d	192.168.
cdl-slot-session-c1-m1-1	1/1	Running	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 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：すべてのスロットで書き込み操作に使用できるプライマリ/セカンダリ概念はありません。プライマリの選出が行われないため、フェールオーバー時間が短縮されます。

注：デフォルトでは、CDLはdb-ep用に2つのレプリカ、1つのスロットマップ（マップあたり2つのレプリカ）、1つのインデックスマップ（マップあたり2つのレプリカ）で導入されます。

## 4.構成の段階的検証

```
smf# show running-config cd1
cd1 system-id 1 // unique across the site, system-id 1 is the primary site ID for
cd1 node-type db-data // node label to configure the node affinity
cd1 enable-geo-replication true // CDL GR Deployment with 2 RACKS
cd1 remote-site 2
db-endpoint host x.x.x.x // Remote site cd1-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
cd1 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
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 cd1-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.トラブルシューティング

### 5.1ポッドの障害

CDLの操作は単純なキー>値dbです。

- すべての要求はcdl-endpoint podsに到達します。
- cdl-indexポッドには、キー、ラウンドロビンが格納されます。
- cdl-slotには、値（セッション情報）、ラウンドロビンが格納されます。
- ポッドマップ（タイプ）ごとにバックアップ（レプリカ数）を定義します。
- Kafkaポッドはトランスポートバスとして使用されます。
- ミラーメーカーは、異なるラックへの転送バスとして使用されます（地理的冗長性）。

それぞれの障害は、このタイプ/マップのすべてのポッドが同時にダウンした場合に、次のように変換できます。

- cdl-endpoint - CDLとの通信エラー
- cdl-index - セッションデータへのキーの喪失
- cdl-slot : セッションデータの損失
- Kafka : ポッドタイプマップ間の同期オプションの喪失
- ミラーメーカー - 他のgeo redundantノードとの同期が失われました

cdlポッドのログは迅速にロールオーバーされないため、関連するポッドからログを常に収集できます。したがって、ログを収集するための追加の値があります。

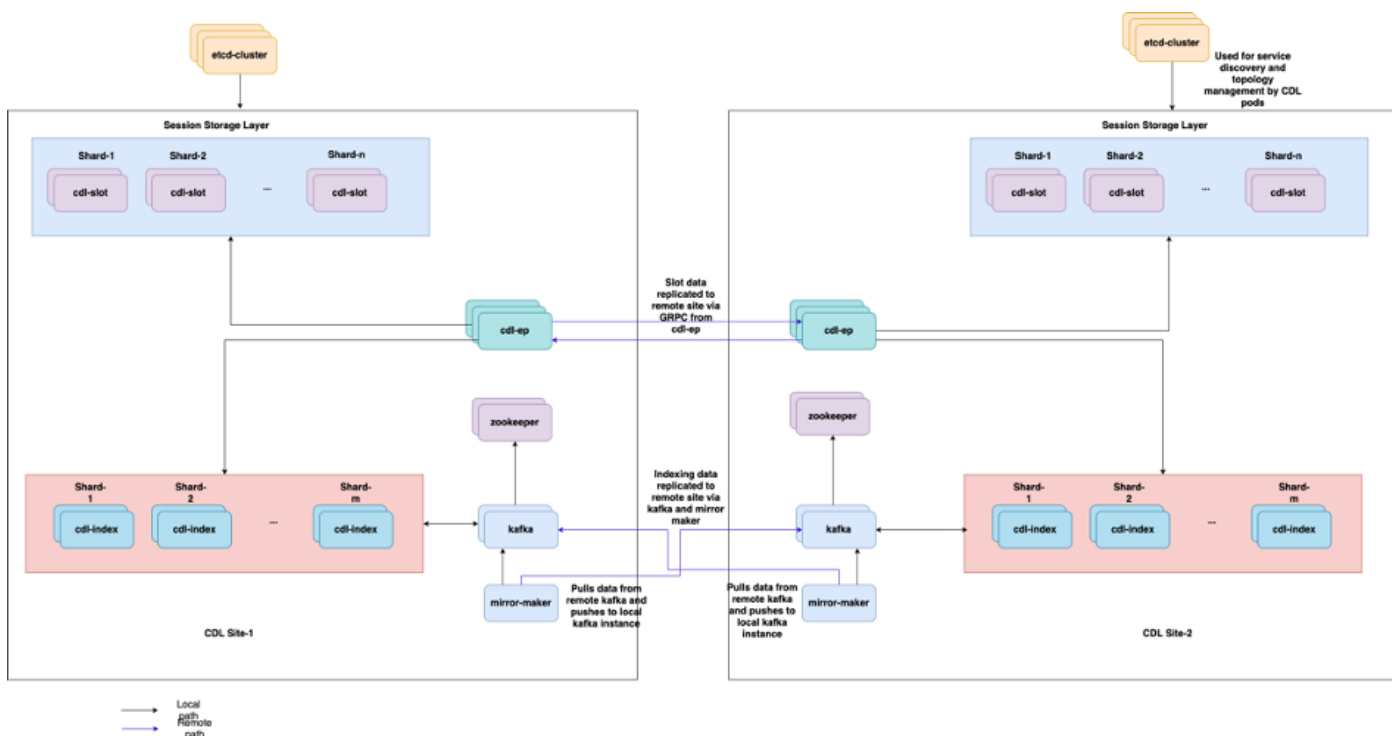
Remember tac-debugはスナップショットを時間内に収集し、ログは保存後にすべてのデータを出力します。

ポッドの説明

```
kubectl describe pod cd1-ep-session-c1-d0-7889db4d87-5mln5 -n smf-rcdn
```

## ポッドログの収集

```
kubectl logs cd1-ep-session-c1-d0-7c79c87d65-xpm5v -n smf-rcdn
```



## 5.2 CDLセッションキーからセッション情報を取得する方法

CDL内の各セッションには、このセッションを識別する一意キーと呼ばれるフィールドがあります。

show subscriber supiおよびcdl show sessions summary slice-name slice1 db-name session filterからのセッションプリントアウトを比較します。

- supi = "1##imsi-123969789012404:10.0.0.3"と組み合わせられたipv4セッションアドレス
- ddn + ip4アドレス = "1##lab:10.0.0.3"
- supi = "1##imsi-123969789012404:2001:db0:0:2::"と組み合わせられたipv6セッションアドレス
- ddn + ipv6 address from session = "1##lab:2001:db0:0:2::"
- smfTeidもN4 Session Key = "1##293601283"これはUPFのエラーをトラブルシューティングするとき非常に便利です。セッションログを検索してセッション関連情報を見つけることができます。
- supi + ebi = "1##imsi-123969789012404:ebi-5"
- supi + ddn = "1##imsi-123969789012404:lab"

```
[smf/data] smf# cd| show sessions summary slice-name slice1 db-name session filter { condition match ke
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#
```

これをSMFからの出力と比較すると、次のようになります。

```
[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"
  ]
}
}
```

SMFのCDLステータスをチェックします。

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

## 5.3 CDLポッドが起動していない

### 識別方法

describe podsの出力(containers/member/State/Reason、events)を確認します。

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

### 修正方法

1. ポッドは保留状態です。ラベル値がcdl/node-typeの値と等しいk8sノードのレプリカ数が、ラベル値がcdl/node-typeの値と等しいk8sノードの数以下であるかどうかを確認します

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

2. Pods are in CrashLoopBackOff failure State etcd pods statusをチェックします。etcdポッドが動作していない場合は、etcdの問題を修正します。

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

3. Pod are in ImagePullBack failure state helmリポジトリとイメージレジストリにアクセスできるかどうかを確認します。必要なプロキシとdnsサーバが設定されているかどうかを確認します。

## 5.4 Mirror Makerポッドがinit状態にある

describe podsの出力とポッドログを確認します。

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

### 修正方法

- Kafka用に設定された外部IPが正しいかどうかを確認します
- 外部IPを使用してリモートサイトのkafkaが利用可能かどうかを確認します。

## 5.5 CDLインデックスが正しく複製されない

### 識別方法

あるサイトで追加されたデータは、他のサイトからアクセスできません。

### 修正方法

- ローカルシステムIDの設定とリモートサイトの設定を確認します。
- 各サイト間のCDLエンドポイントとkafkaの到達可能性を確認します。
- 各サイトのマップ、インデックスのレプリカ、およびスロットを確認します。すべてのサイトで同じにすることができます。

## 5.6 CDL操作は失敗するが、接続は成功する

### 修正方法

- すべてのポッドが準備完了状態で稼働状態であることを確認します。
- インデックス・ポッドは、ピア・レプリカとの同期が完了した場合にのみ準備完了状態になります（利用可能な場合はローカルまたはリモート）
- スロット・ポッドは、ピア・レプリカとの同期が完了した場合にのみ（使用可能な場合はローカルまたはリモート）レディ状態になります。
- 1つ以上のスロットと1つのインデックスポッドが使用できない場合、エンドポイントは準備完了状態ではありません。準備ができていない場合でも、grpc接続はクライアントから受け入れられます。

## 5.7レコードのページの通知がCDLから早く、または遅れた

### 修正方法

- k8sクラスタでは、すべてのノードを時間同期できます
- すべてのk8sノードのNTP同期ステータスをチェックします。問題がある場合は、それを修正します。

chronyc tracking  
chronyc sources -v  
chronyc sourcestats -v

## 6.アラート

ALARM	severity	概要
cdlLocalRequestFailure	critical	ローカル要求の成功率が90 %未満で5分以上続くと、はアラームをトリガーします
cdlRemoteConnectionFailure	critical	エンドポイントポッドからリモートサイトへのアクティブな接続が5分以上続いて0に達した場合は、アラームが発生します ( GR対応システムのみ )
cdlRemoteRequestFailed	critical	着信リモート要求の成功率が5分以上90 %未満の場合、はアラームをトリガーします ( GR対応システムのみ )
cdlReplicationError	critical	cdl-global名前空間のローカル要求に対する出力方向のレプリケーション要求の比率が5分以上経過した後で、90%を下回った場合 ( GR対応システムのみ )。これらのアラートはアップグレードアクティビティ中に発生することが予想されるため、無視しても構いません。
cdlKafkaRemoteReplicationDelay	critical	リモートサイトへのkafkaレプリケーションの遅延が5分を超えて10秒を超えると、アラームが発生します ( GR対応システムのみ )
cdlOverload – メジャー	major	CDLシステムが容量の設定済みパーセンテージ ( デフォルトは80 % ) に達すると、システムはアラームをトリガーします ( 過負荷保護機能が有効になっている場合のみ )
cdlOverload – 重大	critical	CDLシステムがその容量の設定済みのパーセンテージ ( デフォルトは90 % ) に達すると、システムはアラームをトリガーします ( 過負荷保護機能が有効になっている場合のみ )

ALARM	severity	概要
cdlKafkaConnectionFailure	critical	CDLインデックスポッドがkafkaから5分以上切断されている場合

## 7.最も一般的な問題

### 7.1 cdlReplicationError

このアラートは通常、オペレーションセンターまたはシステムアップグレードの起動時に表示され、そのCRを検索し、アラートのCEEの発生を確認しようとし、すでにクリアされています。

### 7.2 cdlRemoteConnectionFailureおよびGRPC\_Connections\_Remote\_Site

この説明は、すべての「cdlRemoteConnectionFailure」アラートと「GRPC\_Connections\_Remote\_Site」アラートに適用されます。

cdlRemoteConnectionFailureアラート：

CDLエンドポイントログに、CDLエンドポイントポッドからリモートホストへの接続が失われたことが記録されています。

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

CDLエンドポイントポッドはリモートサーバに接続しようとしていますが、リモートホストによって拒否されています。

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

リモートホストが5分間到達不能な状態が続いたため、次のようにアラートが発生しました。

```
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 cd1-ep-session-c1-d0-6d86f55945-pxfx9
summary "CDL endpoint connections from pod cd1-ep-session-c1-d0-6d86f55945-pxfx9 and namespace smf-rcdn
labels [ "alertname: cd1RemoteConnectionFailure" "cluster: smf-data-rcdn_cee" "monitor: prometheus" "na
annotations [ "summary: CDL endpoint connections from pod cd1-ep-session-c1-d0-6d86f55945-pxfx9 and nam
```

リモートホストへの接続は02:10:32に成功しました：

```
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
```

CDLリモートサイトのSMFに存在する設定：

```
cd1 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
```

アラートGRPC\_Connections\_Remote\_Siteの場合：

「GRPC\_Connections\_Remote\_Site」も同じCDLエンドポイントポッドからであるため、同じ説明が適用されます。

```
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" "
```

CDLエンドポイントのポッドログから、リモートホストへの接続が拒否されたときにアラートが開始されました。

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

リモートサイトへの接続が成功したときにアラートがクリアされました：

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] Cd] 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

## ハグラフィアナ

CDLダッシュボードは、すべてのSMF展開に含まれています。

## 翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。