

Begrijp de algoritme om beheer in UCCX-failover-scenario's te bepalen

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Normale motorfailover](#)

[herstel van de eilandmodus](#)

Inleiding

Dit document beschrijft de algoritme die wordt gebruikt om meesterschap te bepalen nadat de failover of de terugwinning uit eilandmodus is geïnitieerd in Unified Contact Center Express (UCCX).

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- UCCX
- failover-mechanisme

Gebruikte componenten

Dit document is niet beperkt tot specifieke software- en hardware-versies.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk levend is, zorg er dan voor dat u de mogelijke impact van om het even welke opdracht begrijpt.

Achtergrondinformatie

Twee scenario's waar een master moet worden gekozen zijn mogelijk en het algoritme dat wordt gebruikt om het meesterschap tussen de twee knooppunten te bepalen is verschillend voor elk scenario.

Normale motorfailover

Dit scenario wordt aangetroffen wanneer er geen master is (zoals wanneer een cluster is gestart) of er is één master, wanneer knooppunten in een HAoLAN- of HAoWAN-omgeving zonder eilandmodus zijn.

De algoritme watervallen om het meesterschap te bepalen (d.w.z. probeer 1, anders probeer 2 anders probeer 3, anders probeer 4 in gevallen van conflict):

Stap 1. Bepaal de status van de UCCX Engine van beide knooppunten, welke van beide in een betere status is - dat is de nieuwe master. Als beide gelijk zijn, ga dan naar stap 2.

Stap 2. Bepaal het hardwaremodel van de twee knooppunten. De betere hardware is de nieuwe meester. Als beide gelijk zijn, gaat u naar stap. Aangezien veel UCCX-installaties nu virtueel zijn, wordt deze stap niet vaak gebruikt.

Stap 3. Bepaal knooppunt 1, d.w.z. uitgever (het eerste UCCX-knooppunt geïnstalleerd). De nieuwe meester is het knooppunt van uitgevers. CVD is geprogrammeerd om Node 1 als standaardmaster te maken. Dit komt van de PrimaireEngineComputerName parameter in de cluster configuratie (ClusterSpecificConfig) op CET. Als deze waarde onjuist is, neemt Node2 altijd het meesterschap over. Raadpleeg : [CSCuw95068](#).

Stap 4. Als Stap 3 niet het juiste hostname van Uitgever kan bepalen, maak Node 2 als de master (Subscriber).

De logica is:

Stap 1 . Controleer de servicestatus van de knooppunten. Als Node 1 IN_SERVICE is en knooppunt 2 in PARTIAL_SERVICE is, wordt Node 1 de master. Als de staten hetzelfde zijn (IN_SERVICE of PARTIAL_SERVICE) ga naar STAP 2.

Stap 2 . De hardwarespecificatie van de 2 UCCX-knooppunten is ingeschakeld. De server met de betere specificatie krijgt het meesterschap. Als de hardwarespecificaties gelijk zijn, ga naar STAP 3.

Stap 3 . De PUBLISHER wordt de MASTER als de hostnaam van de PUBLISHER overeenkomt met de PrimaireEngineComputerName op CET (ClusterSpecificConfig). Als er geen MATCH is, ga dan naar stap 4.

Stap 4 . Maken Subscriber Master als de bovenstaande stap faalt.

herstel van de eilandmodus

Dit scenario wordt aangetroffen bij het herstel van de eilandmodus wanneer er twee kapiteins zijn. Als dit gebeurt, wordt het bovenstaande algoritme niet uitgevoerd. Het is eerder zo dat het UCCX Publisher-knooppunt (het eerste UCCX-knooppunt geïnstalleerd) onder de heerschappij blijft vallen en dat de abonnee het meesterschap behoudt.

Opmerking: Een belangrijk iets om op te merken is dat de hostnaam van het primaire knooppunt moet overeenkomen met de ingang PrimaireEngineComputerNam in het object ClusterSpecificConfig. Anders wordt het secundaire knooppunt gekozen als de master. Gebruik het CET-gereedschap om verbinding te maken met het primaire knooppunt, om te

controleren of de ingang juist is en om deze indien nodig te wijzigen.

Daarnaast is dit de manier om de gecontroleerde specifieke diensten te controleren wanneer het systeemknooppunt in een betere servicestatus verkeert, zoals vermeld in stap 1

- Engine Service
- Manager-component binnen de Engine

Als beide services IN_SERVICE zijn, dan wordt dit knooppunt beschouwd als meesterschap.

Dit is een fragment uit de logboeken waarin het algoritme wordt gebruikt om het scenario te verklaren: In dit scenario werd aan knooppunt 1 verteld dat hij de baas was voor de WAN-onderbreking; En toen het WAN terugkwam, werd Node 2 de MASTER.

Wanneer de WAN-link naar beneden is gegaan:

Eerst waren beide knooppunten Meester. Node 1 was de Master; Node 2 werd ook de baas:

Cisco Unified CCX Engine op knooppunt 2 verandert uw kennis van onjuist in waar

```
3162: Dec 15 12:41:17.607 IST %MCVD-CLUSTER_MGR-7-UNK:JavaService167
```

Dit is ook het moment waarop het Node een crash van het andere knooppunt vermoedt:

```
3111: Dec 15 12:41:17.481 IST %MCVD-CVD-4-HEARTBEAT_SUSPECT_NODE_CRASH:CVD suspects node crash:
state=Heartbeat State,nodeInfo=Node id=1 ip=172.30.72.2 convId=69 cmd=16 viewLen=1,dt=1022
```

Toen de WAN-link terugkwam en de CONVERGENTIE begon:

```
9777: Dec 15 12:42:28.859 IST %MCVD-CVD-4-MASTER_DETECTS_NODE_JOIN:More than one master
detected, when processing node join: name=Cisco Unified CCX Database,nodeId=2,masterCnt=1
9778: Dec 15 12:42:28.859 IST %MCVD-CVD-7-UNK:Split after network partition is detected, new
nodeId=2
```

```
Node id=002, addresses=[172.30.83.2], MAC addresses=[279f2d5ba86d], compName=UCCXSUB, state=IN
SERVICE, en=true, rmiPort=6999, masterPort=1994
```

```
VersionInfo: [ Version=8.5.1.11003-32, crsRelease=8.5.1.11003-32, crsServiceRelease=,
crsEngineeringSpecial=, dbEdition=IDS, dbVersion=V11, installTime=1348139852000,
upgradeTime=1348139852000, jtapiClientVersion=8.6(2.10000)-2 ]
```

```
cT=969, uT=969, rT=528, serVer=3, cvdVer=3, points=0
```

```
Component201: type=CRS Historical Datastore, state=IN SERVICE, en=true, prim=false, node=002,
activationTime=1348141153000, parent=null, uT=492, rT=193, rootDir /opt/cisco/uccx,
version=8.5.1.11003-32, serVer=1
```

```
Service163: name=Cisco Unified CCX Database, Feature Service, isActivationSupported=false,
node=002, state=IN SERVICE, master, parent=null, type=DB Services, logDir:
/common/informix/crs/???, en=true, uT=928, rT=0, version=8.5.1.11003-32, serVer=4
```

```
Component202: type=Cisco Recording, state=IN SERVICE, en=true, prim=false, node=002,
activationTime=1348140987000, parent=null, uT=439, rT=198, rootDir /opt/cisco/uccx,
version=8.5.1.11003-32, serVer=1
```

```
9823: Dec 15 12:42:38.866 IST %MCVD-CLUSTER_MGR-7-UNK:Post Convergence Event:
CONVERGENCE_STARTED, name=Cisco Unified CCX Engine
```

```
9824: Dec 15 12:42:38.866 IST %MCVD-CLUSTER_MGR-7-UNK:Cl Mgr: Cisco Unified CCX Engine
Convergence Started
9825: Dec 15 12:42:38.866 IST %MCVD-CLUSTER_MGR-7-UNK:try to process
MasterConvergenceCompletedCmdImpl: name Cisco Unified CCX Engine, nodeId=1, type=MASTER_DROPPED,
uniqueId=66, master=false, updateTick=3101, baseTick=3100, nodeCurrentTick=3101
9826: Dec 15 12:42:38.866 IST %MCVD-CLUSTER_MGR-7-UNK:process MasterConvergenceCompletedCmdImpl:
name Cisco Unified CCX Engine, nodeId=1, type=MASTER_DROPPED, uniqueId=66, master=false,
updateTick=3101, baseTick=3100, nodeCurrentTick=3101
9827: Dec 15 12:42:38.866 IST %MCVD-CLUSTER_MGR-7-UNK:JavaService66: Cisco Unified CCX Engine on
node 1 change master from true to false
```

Toen begon de Convergentie. Om deze reden wordt het eerder toegelichte algoritme gebruikt om de master te selecteren. Let op de status van beide knooppunten:

```
Node id=001, addresses=[172.30.72.2], MAC addresses=[95eab6e4c4cb], compName=UCCXPUB,
state=PARTIAL SERVICE, en=true, rmiPort=6999, masterPort=1994
  VersionInfo: [ Version=8.5.1.11003-32, crsRelease=8.5.1.11003-32, crsServiceRelease=,
crsEngineeringSpecial=, dbEdition=IDS, dbVersion=V11, installTime=1348064353000,
upgradeTime=1348064353000, jtapiClientVersion=8.6(2.10000)-2 ]
  cT=3275, uT=3275, rT=534, serVer=3, cvdVer=3, points=0

Node id=002, addresses=[172.30.83.2], MAC addresses=[279f2d5ba86d], compName=UCCXSUB, state=IN
SERVICE, en=true, rmiPort=6999, masterPort=1994
  VersionInfo: [ Version=8.5.1.11003-32, crsRelease=8.5.1.11003-32, crsServiceRelease=,
crsEngineeringSpecial=, dbEdition=IDS, dbVersion=V11, installTime=1348139852000,
upgradeTime=1348139852000, jtapiClientVersion=8.6(2.10000)-2 ]
  cT=969, uT=969, rT=528, serVer=3, cvdVer=3, points=0
```

Om die reden, door het algoritme, wordt het meesterschap overhandigd aan Node 2 (punt 1 in het algoritme). Dit verklaart waarom UCCX Node 2 na de convergentie de Master werd.

U moet echter wel controleren waarom knooppunt 1 een gedeeltelijke service had. Er was een gedeeltelijke service door het subsysteem telefonie:

```
name=Unified CM Telephony Subsystem, Feature Service, isActivationSupported=false, node=001,
state=PARTIAL SERVICE
```