

排除CNDP上的Kubernetes Pod崩溃故障

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[问题](#)

[分析](#)

[行动计划](#)

简介

本文档介绍如何对云本地部署平台(CNDP)上的Pod崩溃进行故障排除。

先决条件

要求

本文档没有任何特定的要求。

使用的组件

本文档不限于特定的软件和硬件版本。

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。如果您的网络处于活动状态,请确保您了解所有命令的潜在影响。

背景信息

在此设置中,云本地部署平台(CNDP)托管会话管理功能(SMF)。

问题

您会看到有关Pod崩溃的通用执行环境(CEE)的警报。

Command:

```
cee# show alerts active summary summary
```

Example:

```
[smf-rcdn/cee-rcdn] cee# show alerts active summary summary  
NAME UID SUMMARY
```

```
k8s-pod-crashing-loop bd4394046466 Pod smf-rcdn/smf-service-n0-6 (smf-service) is...
k8s-pod-crashing-loop 0ac1019911e3 Pod smf-rcdn/smf-service-n0-14 (smf-service) i...
k8s-pod-crashing-loop eeff8fa16660 Pod smf-rcdn/smf-service-n0-9 (smf-service) is...
k8s-pod-crashing-loop 470ff66822dc Pod smf-rcdn/smf-service-n0-5 (smf-service) is...
k8s-pod-crashing-loop cc8950f07ace Pod smf-rcdn/smf-service-n0-15 (smf-service) i...
k8s-pod-crashing-loop 05a7d1e291a6 Pod smf-rcdn/smf-service-n0-3 (smf-service) is...
```

分析

连接到主节点并显示所有崩溃的Kubernetes Pod。Grep for CrashLoopBackOff. 从相同输出中，我们可以看到此Pod重新启动的次数。

Command:

```
master$ kubectl get pods -n
```

```
cloud-user@smf-rcdn-master-1:~$ kubectl get pods -n smf-rcdn |grep -v Running
```

NAME	READY	STATUS	RESTARTS
smf-service-n0-10 6d7h	1/2	CrashLoopBackOff	1224
smf-service-n0-11 6d7h	1/2	CrashLoopBackOff	1242
smf-service-n0-15 6d7h	1/2	CrashLoopBackOff	1244
smf-service-n0-2 6d7h	1/2	CrashLoopBackOff	1241
smf-service-n0-3 6d7h	1/2	CrashLoopBackOff	1251
smf-service-n0-5 6d7h	1/2	CrashLoopBackOff	1231
smf-service-n0-7 6d7h	1/2	CrashLoopBackOff	1249

描述崩溃的Pod。这样您就可以了解有关Pod崩溃原因的更多详细信息。在Events (事件) 下观察日志。

Command:

```
master$ kubectl describe pod -n
```

```
:
```

```
cloud-user@smf-rcdn-master-1:~$ kubectl describe pod -n smf-rcdn smf-service-n0-11 |grep -i start
Start Time: Tue, 09 Aug 2022 03:13:54 +0000 Started: Tue, 09 Aug 2022 03:13:56 +0000 Restart
Count: 0 Started: Mon, 15 Aug 2022 11:33:10 +0000 Started: Mon, 15 Aug 2022 11:26:55 +0000
Restart Count: 1263 Started: Tue, 09 Aug 2022 03:13:58 +0000 Restart Count: 0 Events: Type
Reason Age From Message ---- - - - - - - - - - - - - - - - - - - - - - - Warning BackOff 65s (x15210 over 3d6h)
kubelet Back-off restarting failed container
```

例如，您有Pod smf-service-n1-0 发生故障时，您需要连接到 smf-rcdn-service-ims2 收集核心文件。

```
ubuntu@smf-rcdn-master1:~$ kubectl get pods -n smf-ims -o wide | grep smf-service-n1-0
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE
NOMINDATEDN NODE	READINESS	GATES				
smf-service-n1-0	2/2	Running	10	9h	10.20.9.142	smf-rcdn-service-ims2

连接到节点是崩溃并收集二进制文件的主机Pod。思科分析需要此文件。

Command:

```
master1:~$ kubectl cp
```

Example:

```
ubuntu@smf-rcdn-master1:~$ kubectl cp smf-ims/smf-service-n1-0:/opt/workspace/smf-service
/tmp/smf-service
```

连接到节点是崩溃的主机Pod，并转至文件夹/var/lib/systemd/coredump/并显示内容。如果已生成，您可以在此文件夹中看到它们。

Example:

```
ubuntu@smf-rcdn-master1:~$ ssh smf-rcdn-service-ims2
ubuntu@smf-rcdn-service-ims2:~$ cd /var/lib/systemd/coredump/
ubuntu@smf-rcdn-service-ims2:/var/lib/systemd/coredump$ ls -ltr
total 982340
-rw-r----- 1 root root 52968460 Sep 21 16:40 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.1232.1599842408000000.lz4
-rw-r----- 1 root root 61609776 Sep 21 16:41 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.3468.1599842463000000.lz4
-rw-r----- 1 root root 74233259 Sep 21 16:46 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.28259.1599842775000000.lz4
-rw-r----- 1 root root 58241763 Sep 21 16:52 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.17155.1599843174000000.lz4
-rw-r----- 1 root root 43732684 Sep 21 16:56 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.3076.1599843385000000.lz4
-rw-r----- 1 root root 52377930 Sep 21 17:06 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.8024.1599844002000000.lz4
-rw-r----- 1 root root 63990106 Sep 21 17:07 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.26962.1599844074000000.lz4
-rw-r----- 1 root root 98058261 Sep 21 17:15 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.13026.1599844546000000.lz4
-rw-r----- 1 root root 59586871 Sep 21 17:24 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.21720.1599845052000000.lz4
-rw-r----- 1 root root 71187759 Sep 21 17:50 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.19705.1599846648000000.lz4
-rw-r----- 1 root root 96949278 Sep 21 17:57 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.11744.1599847049000000.lz4
-rw-r----- 1 root root 6052439 Sep 21 17:57 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.23846.1599847052000000.lz4
-rw-r----- 1 root root 70642243 Sep 21 17:58 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.18327.1599847110000000.lz4
-rw-r----- 1 root root 66052273 Sep 21 18:10 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.1504.1599847843000000.lz4
-rw-r----- 1 root root 65132876 Sep 21 18:10 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.12528.1599847855000000.lz4
-rw-r----- 1 root root 65000665 Sep 21 18:32 core.smf-
service.0.a829fbabe2e649a7ab02150838fe47ae.9462.1599849167000000.lz4
```

```
ubuntu@smf-rcdn-master1:~$:/var/lib/systemd/coredump$
```

压缩文件夹中的所有文件。

```
ubuntu@smf-rcdn-service-ims2:~$ sudo tar czvfsmf-rcdn-service-ims2.tar.gz *.lz4
```

从主SFTP到核心所在的节点，并将它们下载到主/tmp文件夹，然后将其拉到PC。

```
ubuntu@smf-rcdn-master1:~$ sftp smf-rcdn-service-ims2
```

命令会在上次Pod重新启动之前打印日志，并捕获崩溃的签名。

Command:

```
master:~$ kubectl logs -n
```

Example:

```
ubuntu@smf-rcdn-master1:~$ kubectl logs -n smf-ims -p smf-service-n1-0 -c smf-service
/usr/local/go/src/runtime/asm_amd64.s:1357 (0x462d01) panic: runtime error: invalid memory
address or nil pointer dereference [signal SIGSEGV: segmentation violation code=0x1 addr=0x50
pc=0x13d92f6] goroutine 839296 [running]: panic(0x196c320, 0x3441300)
/usr/local/go/src/runtime/panic.go:722 +0x2c2 fp=0xc000a9d050 sp=0xc000a9cfc0 pc=0x432d82
runtime.panicmem(...) /usr/local/go/src/runtime/panic.go:199 runtime.sigpanic()
/usr/local/go/src/runtime/signal_unix.go:394 +0x3ec fp=0xc000a9d080 sp=0xc000a9d050 pc=0x4487cc
smf-service/userplane.(*UpfServData).
```

```
ProcessSessionModificationResponse(0xc0059fe660, 0xc005b98f00, 0xc00aa6e3c0,
0x2001181ae72b892, 0xc00ea43570, 0x3, 0x4,
```

```
0xc005cd0820, 0xc005b11410, 0xc005b10d20, ...) /opt/workspace/smf-service/src/smf-
service/userplane/upfSessionModification.go:743 +0x526 fp=0xc000a9d408 sp=0xc000a9d080
pc=0x13d92f6 smf-
service/procedures/4g/pdn5g4gHo.(*Pdn5g4gHoProcedure).awtUpfModifyProcN4ModifyResp(0xc005a17440,
0xc0099e36c0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0) /opt/workspace/smf-service/src/smf-
service/procedures/4g/pdn5g4gHo/mbrUtils.go:485 +0x24d fp=0xc000a9d630 sp=0xc000a9d408
pc=0x1562d0d smf-
service/procedures/4g/pdn5g4gHo.(*Pdn5g4gHoProcedure).handleUpfModifyEvents(0xc005a17440,
0xc0099e36c0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0) /opt/workspace/smf-service/src/smf-
service/procedures/4g/pdn5g4gHo/stateHandler.go:196 +0x4a1 fp=0xc000a9d768 sp=0xc000a9d630
pc=0x1570d31 smf-service/procedures/4g/pdn5g4gHo.(*Pdn5g4gHoProcedure).HandleEvent(0xc005a17440,
0xc0099e36c0, 0x6, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, ...) /opt/workspace/smf-service/src/smf-
service/procedures/4g/pdn5g4gHo/procedure.go:364 +0x707 fp=0xc000a9d8d0 sp=0xc000a9d768
pc=0x1567887 smf-service/vendor/wwin-github.cisco.com/mobile-cnat-smf/smf-common.git/src/smf-
common/callflow.(*BaseProcedure).Handle(0xc00568b4a0, 0xc0099e36c0, 0x0,
0x0, 0x0, 0x0, 0x0, 0x0, 0x0) /opt/workspace/smf-service/src/smf-service/vendor/wwin-
github.cisco.com/mobile-cnat-smf/smf-common.git/src/smf-common/callflow/BaseProcedure.go:54
+0xdb
fp=0xc000a9d978 sp=0xc000a9d8d0 pc=0xf5996b smf-service/vendor/wwin-github.cisco.com/mobile-
cnat-smf/smf-common.git/src/smf-common/callflow.(*SessionState).ProcessContinue(0xc00b79b6d0,
0xc0099e36c0,
0xc00568b4a0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, ...) /opt/workspace/smf-service/src/smf-
service/vendor/wwin-github.cisco.com/mobile-cnat-smf/smf-common.git/src/smf-
common/callflow/SessionState.go:169 +0x1f2
```

```
fp=0xc000a9da20 sp=0xc000a9d978 pc=0xf5d552 smf-  
service/processor.(*SmfAppMessageProcessor).ProcessContinue(0x3a31da0, 0xc005b98f00, 0x1d34988,  
0x35, 0x9, 0x1d34988, 0x35) /opt/workspace/smf-service/src/smf-  
service/processor/grpc_message_processor.go:430 +0x4ab fp=0xc000a9dc20 sp=0xc000a9da20  
pc=0x174fc0b smf-service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-  
infra.git/src/app-infra/infra.(*masterBlueprint).processTransaction  
(0xc0003141e0, 0xc005b98f00, 0xc000a9dd98) /opt/workspace/smf-service/src/smf-  
service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-infra.git/src/app-  
infra/infra/MasterBlueprint.go:301  
+0x1a7 fp=0xc000a9dce8 sp=0xc000a9dc20 pc=0xd39ca7 smf-service/vendor/wwwin-  
github.cisco.com/mobile-cnat-golang-lib/app-infra.git/src/app-infra/infra.(*masterBlueprint).  
processTransactionWithCR(0xc0003141e0, 0xc005b98f00, 0x1cfeb00) /opt/workspace/smf-  
service/src/smf-service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-  
infra.git/src/app-infra/infra/MasterBlueprint.go:234  
+0x394 fp=0xc000a9de78 sp=0xc000a9dce8 pc=0xd396e4 smf-service/vendor/wwwin-  
github.cisco.com/mobile-cnat-golang-lib/app-infra.git/src/app-infra/infra.(*masterBlueprint).  
processSessionTransaction(0xc0003141e0, 0xc005b98f00, 0x1, 0x0) /opt/workspace/smf-  
service/src/smf-service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-  
infra.git/src/app-infra/infra/MasterBlueprint.go:177  
+0x124 fp=0xc000a9ded0 sp=0xc000a9de78 pc=0xd39104 smf-service/vendor/wwwin-  
github.cisco.com/mobile-cnat-golang-lib/app-infra.git/src/app-infra/infra.(*masterBlueprint).  
processEvent(0xc0003141e0, 0xc005b98f00, 0x1d02487) /opt/workspace/smf-service/src/smf-  
service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-infra.git/src/app-  
infra/infra/MasterBlueprint.go:138 +0x5fc  
fp=0xc000a9df88 sp=0xc000a9ded0 pc=0xd3869c smf-service/vendor/wwwin-github.cisco.com/mobile-  
cnat-golang-lib/app-infra.git/src/app-infra/infra.(*ApplicationContext).NewTransaction.func2  
(0xc0006af400, 0xc005b98f00) /opt/workspace/smf-service/src/smf-service/vendor/wwwin-  
github.cisco.com/mobile-cnat-golang-lib/app-infra.git/src/app-  
infra/infra/ApplicationContext.go:1268  
+0x7c fp=0xc000a9dfd0 sp=0xc000a9df88 pc=0xd9b69c runtime.goexit()  
/usr/local/go/src/runtime/asm_amd64.s:1357 +0x1 fp=0xc000a9dfd8 sp=0xc000a9dfd0 pc=0x462d01  
created by smf-service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-  
infra.git/src/app-infra/infra.(*ApplicationContext).NewTransaction /opt/workspace/smf-  
service/src/smf-service/vendor/wwwin-github.cisco.com/mobile-cnat-golang-lib/app-  
infra.git/src/app-infra/infra/ApplicationContext.go:1266 +0x62c goroutine 1 [sleep]:  
runtime.gopark(0x1dbaa10, 0x34ef580, 0xc001f01313, 0x2) /usr/local/go/src/runtime/proc.go:304  
+0xe0 fp=0xc000a3bca8 sp=0xc000a3bc88 pc=0x434ea0 runtime.goparkunlock(...)
```

在出现Pod崩溃之前和之后连接到CEE并收集tac-debug。

tac-debug-pkg create from yyyy-mm-dd_hh:mm:ss to yyyy-mm-dd_hh:mm:ss tac-debug-pkg create from yyyy-mm-dd_hh:mm:ss to yyyy-mm-dd_hh:mm:ss

行动计划

[打开思科TAC](#)的服务请求以查找此崩溃的根本原因。

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）。