



CXone Mpower EXPERT

如何收集SECD跟踪

https://kb-cn-stage.netapp.com/on-prem/ontap/da/NAS/NAS-KBs/How_to_collect_a_SECD_trace

Updated: Wed, 22 Apr 2026 07:30:18 GMT

适用场景

- ONTAP 9

问题描述

- 在对与身份验证相关的问题描述进行故障排除时、可能需要提高安全守护进程(SECD)的详细程度。
- 下面概述了与diag级别命令关联的选项及其建议用法。
 - 诊断级别命令只能在NetApp支持的指导下使用
 - 默认情况下、SECD仅记录错误和失败的RPC命令

- 完成故障排除后、清除跟踪。
- 这将通过运行活动跟踪来确保系统不会产生额外开销。
- 要清除、请对所有节点执行以下操作：

```
::*> diag secd trace clear -node <node> -vserverids yes
```

'NetApp provides no representations or warranties regarding the accuracy or reliability or serviceability of any information or recommendations provided in this publication or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS and the use of this information or the implementation of any recommendations or techniques herein is a customers responsibility and depends on the customers ability to evaluate and integrate them into the customers operational environment. This document and the information

追加信息

- SECD是一个diag命令、因此这些示例不会记录在任何面向公众的信息中。
 - 以下是运行跟踪的命令示例
 - 请完整阅读本文档、了解所有不同的选项：

```
test1::*> diag secd trace set -node test1-01 -vserverids main -trace-all yes
```

SECD跟踪设置基础知识：

- secd 使用量限制为diag级别
- set diag 必须在运行命令之前调用。
 - 命令示例：

```
test1::*> diag secd trace ?
clear                *Clear Trace Options
set                  *Set Trace Options
show                 *Show Trace
```

- 使用show 选项确认当前跟踪设置。
 - 如果跟踪设置与参数匹配、请使用clear 选项删除所有当前设置。

```
test1::*> diag secd trace show -node test1-01
Trace Spec
-----
Trace spec has not been set.
```

- 以下是可在设置迹线时使用的参数。
 - 注意：每当运行set 命令时，都需要该节点：

```
test1::*> diag secd trace set ?
[-node] <nodename>                *Node
[[-trace-all] {yes|no}]           *Trace All RPCs
[ -vserverids <vserver>, ... ]    *Id List
[ -ipaddrs <text>, ... ]           *Address List
[ -domain-names <text>, ... ]     *Domain List
[ -dc-names <text>, ... ]          *DC List
[ -rpc-names <rpc_name>, ... ]     *RPC List
[ -module-names <module_name>, ... ] *Module List
[ -trace-connectivity {yes|no} ]   *Trace Connection Errors
```

- 以下示例将在节点test1-01, 上设置一个跟踪、其中包括一个名为"main"的Storage Virtual Machine (SVM)。

```
test1::*> diag secd trace set -node test1-01 -vserverids main -trace-all yes
Trace spec is set successfully for trace-all Vservers.
```

- 检查跟踪参数：

```
test1::*> diag secd trace show -node test1-01
Trace Spec
-----
TraceAll:                Tracing all RPCs
Vserver IDs:             main
```

- 注意：

- 与"set"关联的命令是累积的、与"clear"关联的命令是减去的。
- 因此、在上述示例的基础上、通过运行以下命令将捕获限制为特定域：

```
test1::diag secd trace set -node test1-01 -domain-names test
Trace spec is set successfully for domains.
```

```
test1::diag secd trace show -node test1-01
Trace Spec
-----
TraceAll:                Tracing all RPCs
Domain names:           test
Vserver IDs:             main
```

- 注意：

- 添加域将构建到包含SVM ID的现有跟踪参数中、添加要筛选的IP地址也是如此、如下所示：
- 将跟踪限制为一组特定的IP地址：

```
test1:: secd trace set -node test1-01 -ipaddrs
10.10.10.10,10.10.10.11,10.10.10.12,10.10.10.13
Trace spec is set successfully for ipaddrs.
```

- 现在、跟踪筛选器包括SVM ID、域和特定IP地址列表：

```
test1::diag secd*> trace show -node
test1-01
Trace Spec
-----
TraceAll:                Tracing all RPCs
IP Addrs:                10.10.10.10 10.10.10.11 10.10.10.12 10.10.10.13
Domain names:           test
Vserver IDs:             main
```

- 如上所述、clear 选项为减值。
 - 如以下帮助上下文中所述、与每个参数关联的输入为"是"或"否"。

```
test1::diag secd*> trace clear ?
[-node] <nodename> *Node
[[-trace-all] {yes|no}] *Clear Trace All
[ -vserverids {yes|no} ] *Clear VserverIds
[ -ipaddrs {yes|no} ] *Clear Ipaddrs
[ -domain-names {yes|no} ] *Clear Domains
[ -dc-names {yes|no} ] *Clear DC
[ -rpc-names {yes|no} ] *Clear RPC
[ -module-names {yes|no} ] *Clear Module
[ -trace-connectivity {yes|no} ] *Clear Trace Connection Errors
```

- 因此、假设以下命令将整个条目清除为虚假假设(如下面的show 输出中所示)、请确保清除所有条目。
- 注意：如果参数清除不当、发生原因可能会在将来的日志收集工作中出现问题：

```
test1::diag secd trace clear -node test1-01 -trace-all yes

test1::diag secd trace show -node test1-01
Trace Spec
-----
IP Addr:          10.10.10.10 10.10.10.11 10.10.10.12 10.10.10.13
Domain names:    test
Vserver IDs:     main
```

- 以下语法将完全删除跟踪条目：

```
test1::diag secd trace clear -node test1-01 -ipaddrs yes -domain-names yes
-vserverids yes

test1::diag secd trace show -node test1-01
Trace Spec
-----
Trace spec has not been set.
```

操作步骤-实际跟踪收集：

- 使用上述信息、可以成功设置跟踪参数。
- 如果您不确定要打开的特定模块、最好使用trace-all yes 设置来确保捕获尽可能多的数据。
- 请执行以下步骤：

1. 在复制问题描述之前、请运行show 命令验证参数并将其包含在案例数据中。

注：

- 1) 确保仅在复制期间启用跟踪。**复制完成后、立即禁用跟踪。**
- 2) 必须记录复制过程中涉及的所有相关信息。
时间/日期、用户名和信息、IP地址及其所属人员(客户端、LIF、DC)、文件名、卷、和共享信息将极大地有助于日志分析。

2. 有关跟踪收集顺序、请参见以下示例：

```
test1::> set d
Warning: These diagnostic commands are for use by NetApp personnel only.
Do you want to continue? {y|n}: y

test1::*> diag secd trace show -node test1-01
Trace Spec
-----
Trace spec has not been set.

test1::*> diag secd trace set -node test1-01 -vserverids main -trace-all
yes
Trace spec set successfully for trace-all, Vservers.

test1::*> diag secd trace show -node test1-01
Trace Spec
-----
TraceAll:                Tracing all RPCs
Vserver IDs:              main
```

3. 完成活动后、彻底清除跟踪的所有方面。
运行show 命令以验证是否尚未设置跟踪。

```
test1::*> diag secd trace set -node test1-01 -vserverids main -trace-all no
Trace spec set successfully for Vservers.
Trace spec cleared successfully for trace-all.

test1::*> diag secd trace show -node test1-01
Trace Spec
-----
Vserver IDs:              main

test1::*> diag secd trace clear -node test1-01 -vserverids yes

test1::*> diag secd trace show -node test1-01
Trace Spec
-----
```

Trace spec has not been set.

4. 从两个节点生成“**Type All**”(键入所有**AutoSupport**)(这将上传SECD跟踪数据): `::> autosupport invoke`
`* -type all`

注:

用户也可以上传 `secd` 日志 (/etc/log/mlog)。

- 有关详细信息、请参见文章: [如何将文件上传到NetApp](#)