Red Hat Advanced Cluster Security for Kubernetes 4.4

Troubleshooting Collector

Troubleshooting Collector

Last Updated: 2024-06-27

Red Hat Advanced Cluster Security for Kubernetes 4.4 Troubleshooting Collector

Troubleshooting Collector

Legal Notice

Abstract

Use this guide to learn how to retrieve logs and debug issues with a failing Collector.

Table of Contents

CHAPTER 1. RETRIEVING AND ANALYZING THE COLLECTOR LOGS AND POD STATUS 3
1.1. RETRIEVING THE COLLECTOR LOGS 3
1.1.1. Retrieving the logs with the oc or kubectl command 3
1.1.2. Retrieving logs from a RHACS diagnostic bundle 4
1.2. ANALYZING THE COLLECTOR POD STATUS 4

CHAPTER 2. COMMONLY OCCURRING ERROR CONDITIONS 5
2.1. UNABLE TO CONNECT TO THE SENSOR 6
2.2. UNAVAILABILITY OF THE KERNEL DRIVER 6
2.3. FAILING TO LOAD THE KERNEL DRIVER 7

CHAPTER 1. RETRIEVING AND ANALYZING THE COLLECTOR LOGS AND POD STATUS

The first step in troubleshooting is to retrieve the logs and pods status. The logs allow you to identify the root cause of an error. In addition, examining the pod's most recent status can provide information about failure messages.

1.1. RETRIEVING THE COLLECTOR LOGS

First, you should examine the logs from failing Collectors. Depending on your environment and access rights, you can obtain these logs in two ways:

- Retrieving the logs with the **oc** or **kubectl** command
- Retrieving logs from a RHACS diagnostic bundle

1.1.1. Retrieving the logs with the oc or kubectl command

You can use either the **oc** or **kubectl** command to obtain logs from your running Collector pod. Optionally, you can even check the logs from a previous Collector pod if your current Collector pod is restarting.

Prerequisites

- Ensure that you have the authority to list the pods and logs:

```
$ oc auth can-i get pods && oc auth can-i get pods --subresource=logs
```
(1)

(1) If you use Kubernetes, enter **kubectl** instead of **oc**.

Procedure

1. List all the pods with label **app=collector**:

```
$ oc get pods -n stackrox -l app=collector
```
(1)

(1) If you use Kubernetes, enter **kubectl** instead of **oc**.

Example output

```
collector-vclg5    1/2     CrashLoopBackOff   2 (25s ago)   2m41s+
```

2. Get the logs for the Collector pod:

```
$ oc logs -n stackrox <collector_pod_name> collector
```
(1)

(1) If you use Kubernetes, enter **kubectl** instead of **oc**. For ***<collector_pod_name>***, specify the name of your Collector pod, for example, **collector-vclg5**.

3. (Optional) If the current Collector pod is restarting, you can check the logs for the previous Collector pod:

```
$ oc logs -n stackrox <collector_pod_name> collector --previous 1
```

1 If you use Kubernetes, enter **kubectl** instead of **oc**. For ***<collector_pod_name>***, specify the name of your Collector pod, for example, **collector-vclg5**.

1.1.2. Retrieving logs from a RHACS diagnostic bundle

You can also access Collector logs by downloading a diagnostic bundle from the Red Hat Advanced Cluster Security for Kubernetes (RHACS) user interface. Once you have downloaded the diagnostic bundle, you can inspect the logs for all the Collector pods. For more information, see Generating a diagnostic bundle.

1.2. ANALYZING THE COLLECTOR POD STATUS

Examining the pod's most recent status is another easy way to determine the cause of a Collector crash. Failure messages are recorded to the most recent status and are accessible using the **kubectl describe pod** or **oc describe pod** command.

Procedure

- Describe the Collector pod:

```
$ oc describe pod -n stackrox <collector_pod_name> 1
```

1 If you use Kubernetes, enter **kubectl** instead of **oc**. For ***<collector_pod_name>***, specify the name of your Collector pod, for example, **collector-vclg5**.

Example output

```
# ...
    Last State:     Terminated
      Reason:       Error
      Message:      No suitable kernel object downloaded 1
      Exit Code:    1
      Started:      Fri, 21 Oct 2022 11:50:56 +0100
      Finished:     Fri, 21 Oct 2022 11:51:25 +0100
# ...
```

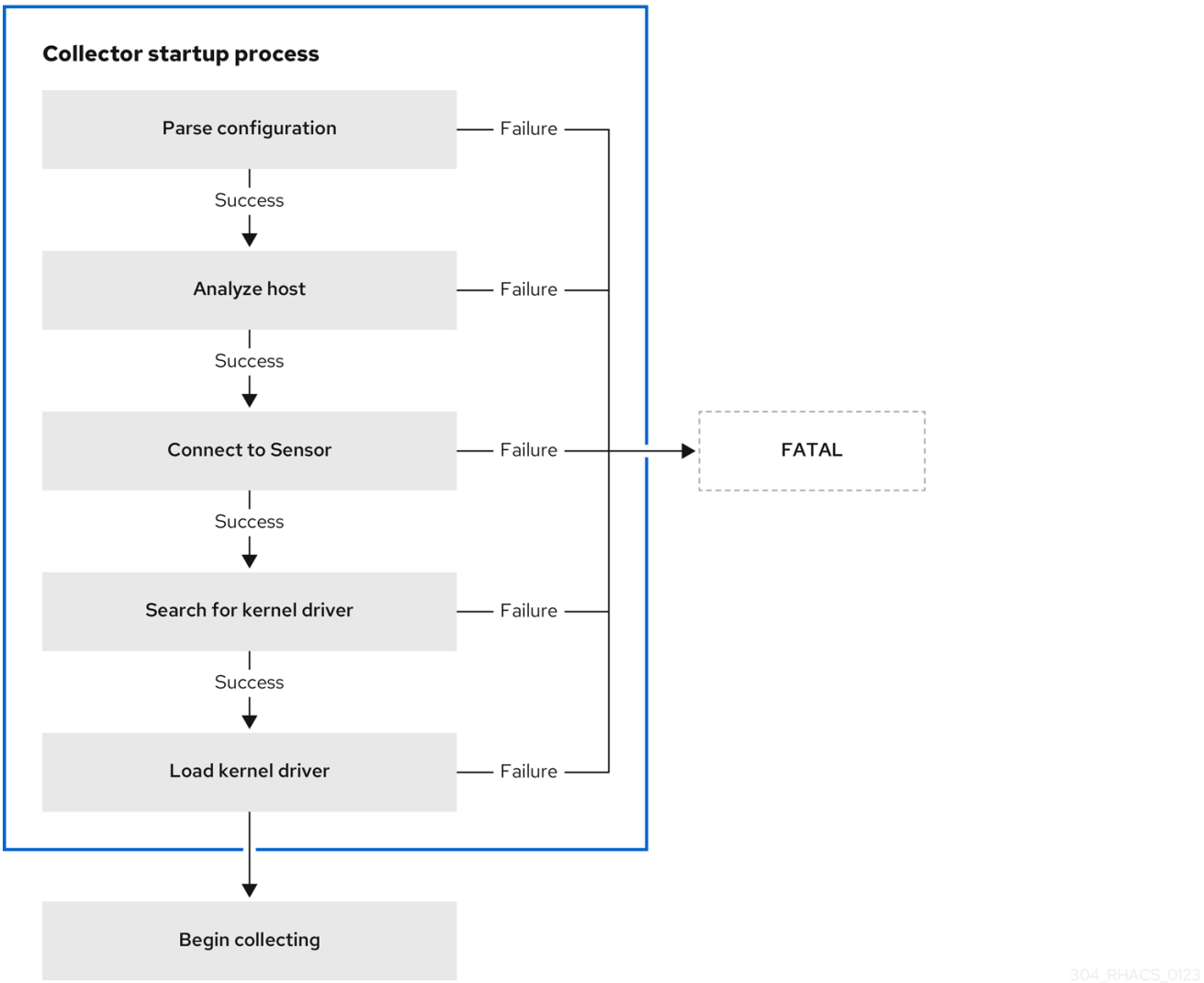
1 In this example, you can see that Collector has failed to download a kernel driver.

CHAPTER 2. COMMONLY OCCURRING ERROR CONDITIONS

Most errors occur during Collector startup when Collector configures itself and finds or downloads a kernel driver for the system.

The following diagram describes the main parts of Collector startup process:

Figure 2.1. Collector pod startup process



If any part of the startup procedure fails, the logs display a diagnostic summary detailing which steps succeeded or failed .

The following log file example shows a successful startup:

```
[INFO  2022/11/28 13:21:55] == Collector Startup Diagnostics: ==
[INFO  2022/11/28 13:21:55]  Connected to Sensor?       true
[INFO  2022/11/28 13:21:55]  Kernel driver available?   true
[INFO  2022/11/28 13:21:55]  Driver loaded into kernel? true
[INFO  2022/11/28 13:21:55] ====================================
```

The log output confirms that Collector connected to Sensor and located and loaded the kernel driver. You can use this log to check for the successful startup of Collector.

2.1. UNABLE TO CONNECT TO THE SENSOR

When starting, first check if you can connect to Sensor. Sensor is responsible for downloading kernel drivers and CIDR blocks for processing network events, making it an essential part of the startup process. The following logs indicate you are unable to connect to the Sensor:

```
Collector Version: 3.15.0
OS: Ubuntu 20.04.4 LTS
Kernel Version: 5.4.0-126-generic
Starting StackRox Collector...
[INFO    2023/05/13 12:20:43] Hostname: 'hostname'
[...]
[INFO    2023/05/13 12:20:43] Sensor configured at address: sensor.stackrox.svc:9998
[INFO    2023/05/13 12:20:43] Attempting to connect to Sensor
[INFO    2023/05/13 12:21:13]
[INFO    2023/05/13 12:21:13] == Collector Startup Diagnostics: ==
[INFO    2023/05/13 12:21:13]  Connected to Sensor?       false
[INFO    2023/05/13 12:21:13]  Kernel driver candidates:
[INFO    2023/05/13 12:21:13] ====================================
[INFO    2023/05/13 12:21:13]
[FATAL   2023/05/13 12:21:13] Unable to connect to Sensor.
```

This error could mean that Sensor has not started correctly or that Collector configuration is incorrect. To fix this issue, you must verify Collector configuration to ensure that Sensor address is correct and that the Sensor pod is running correctly.

View the Collector logs to specifically check the configured Sensor address. Alternatively, you can run the following command:

```
$ kubectl -n stackrox get pod <collector_pod_name> -o jsonpath='{.spec.containers[0].env[?
(@.name=="GRPC_SERVER")].value}' 1
```

1 For ***<collector_pod_name>***, specify the name of your Collector pod, for example, **collector-vclg5**.

2.2. UNAVAILABILITY OF THE KERNEL DRIVER

Collector determines if it has a kernel driver for the node's kernel version. Collector first searches the local storage for a driver with the correct version and type, and then attempts to download the driver from Sensor. The following logs indicate that neither a local kernel driver nor a driver from Sensor is present:

```
Collector Version: 3.15.0
OS: Alpine Linux v3.16
Kernel Version: 5.15.82-0-virt
Starting StackRox Collector...
[INFO    2023/05/30 12:00:33] Hostname: 'alpine'
[INFO    2023/05/30 12:00:33] User configured collection-method=ebpf
[INFO    2023/05/30 12:00:33] Afterglow is enabled
[INFO    2023/05/30 12:00:33] Sensor configured at address: sensor.stackrox.svc:443
[INFO    2023/05/30 12:00:33] Attempting to connect to Sensor
[INFO    2023/05/30 12:00:33] Successfully connected to Sensor.
[INFO    2023/05/30 12:00:33] Module version: 2.5.0-rc1
[INFO    2023/05/30 12:00:33] Config: collection_method:0, useChiselCache:1, scrape_interval:30,
```

```
turn_off_scrape:0, hostname:alpine, processesListeningOnPorts:1, logLevel:INFO
[INFO    2023/05/30 12:00:33] Attempting to find eBPF probe - Candidate versions:
[INFO    2023/05/30 12:00:33] collector-ebpf-5.15.82-0-virt.o
[INFO    2023/05/30 12:00:33] Attempting to download collector-ebpf-5.15.82-0-virt.o
[INFO    2023/05/30 12:00:33] Attempting to download kernel object from
https://sensor.stackrox.svc:443/kernel-objects/2.5.0/collector-ebpf-5.15.82-0-virt.o.gz (1)
[INFO    2023/05/30 12:00:33] HTTP Request failed with error code 404 (2)
[WARNING 2023/05/30 12:02:03] Attempted to download collector-ebpf-5.15.82-0-virt.o.gz 90 time(s)
[WARNING 2023/05/30 12:02:03] Failed to download from collector-ebpf-5.15.82-0-virt.o.gz
[WARNING 2023/05/30 12:02:03] Unable to download kernel object collector-ebpf-5.15.82-0-virt.o to
/module/collector-ebpf.o.gz
[WARNING 2023/05/30 12:02:03] No suitable kernel object downloaded for collector-ebpf-5.15.82-0-
virt.o
[ERROR   2023/05/30 12:02:03] Failed to initialize collector kernel components.
[INFO    2023/05/30 12:02:03]
[INFO    2023/05/30 12:02:03] == Collector Startup Diagnostics: ==
[INFO    2023/05/30 12:02:03]  Connected to Sensor?       true
[INFO    2023/05/30 12:02:03]  Kernel driver candidates:
[INFO    2023/05/30 12:02:03]    collector-ebpf-5.15.82-0-virt.o (unavailable)
[INFO    2023/05/30 12:02:03] ====================================
[INFO    2023/05/30 12:02:03]
[FATAL   2023/05/30 12:02:03] Failed to initialize collector kernel components. (3)
```

1. The logs display attempts to locate the module first, followed by any efforts to download the driver from Sensor.
2. The 404 errors indicate that the node's kernel does not have a kernel driver.
3. As a result of missing a driver, Collector enters the **CrashLoopBackOff** state.

The Kernel versions file contains a list of all supported kernel versions.

2.3. FAILING TO LOAD THE KERNEL DRIVER

Before Collector starts, it loads the kernel driver. However, in rare cases, you might encounter issues where Collector cannot load the kernel driver, resulting in various error messages or exceptions. In such cases, you must check the logs to identify the problems with failure in loading the kernel driver.

Consider the following Collector log:

```
[INFO    2023/05/13 14:25:13] Hostname: 'hostname'
[...]
[INFO    2023/05/13 14:25:13] Successfully downloaded and decompressed /module/collector.o
[INFO    2023/05/13 14:25:13]
[INFO    2023/05/13 14:25:13] This product uses ebpf subcomponents licensed under the GNU
[INFO    2023/05/13 14:25:13] GENERAL PURPOSE LICENSE Version 2 outlined in the /kernel-
modules/LICENSE file.
[INFO    2023/05/13 14:25:13] Source code for the ebpf subcomponents is available at
[INFO    2023/05/13 14:25:13] https://github.com/stackrox/falcosecurity-libs/
[INFO    2023/05/13 14:25:13]
-- BEGIN PROG LOAD LOG --
[...]
-- END PROG LOAD LOG --
[WARNING 2023/05/13 14:25:13] libscap: bpf_load_program()
```

```
event=tracepoint/syscalls/sys_enter_chdir: Operation not permitted
[ERROR   2023/05/13 14:25:13] Failed to setup collector-ebpf-6.2.0-20-generic.o
[ERROR   2023/05/13 14:25:13] Failed to initialize collector kernel components.
[INFO    2023/05/13 14:25:13]
[INFO    2023/05/13 14:25:13] == Collector Startup Diagnostics: ==
[INFO    2023/05/13 14:25:13]  Connected to Sensor?       true
[INFO    2023/05/13 14:25:13]  Kernel driver candidates:
[INFO    2023/05/13 14:25:13]    collector-ebpf-6.2.0-20-generic.o (available)
[INFO    2023/05/13 14:25:13] ====================================
[INFO    2023/05/13 14:25:13]
[FATAL   2023/05/13 14:25:13] Failed to initialize collector kernel components.
```

If you encounter this kind of error, it is unlikely that you can fix it yourself. So instead, report it to Red Hat Advanced Cluster Security for Kubernetes (RHACS) support team or create a GitHub issue.