Red Hat Advanced Cluster Security for Kubernetes 3.70

Backup and restore

Backing up and restoring Red Hat Advanced Cluster Security for Kubernetes

Last Updated: 2022-12-06

Red Hat Advanced Cluster Security for Kubernetes 3.70 Backup and restore

Backing up and restoring Red Hat Advanced Cluster Security for Kubernetes

Legal Notice

Abstract

Describes how to back up the system and restore from a backup.

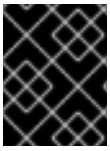
Table of Contents

CHAPTER 1. BACKING UP RED HAT ADVANCED CLUSTER SECURITY FOR KUBERNETES

You can perform data backups for Red Hat Advanced Cluster Security for Kubernetes. You can use these backups for data restoration in the case of an infrastructure disaster, or corrupt data.

You can configure automatic or on-demand backups by integrating with Amazon S3 or Google Cloud Storage. Or you can perform on-demand backups by using the **roxctl** command-line interface (CLI).

The backup includes the entire Red Hat Advanced Cluster Security for Kubernetes database, which includes all configurations, resources, events, and certificates. Make sure that backups are stored securely.



IMPORTANT

If you are using Red Hat Advanced Cluster Security for Kubernetes 3.0.53 or older, the backup does not include certificates.

1.1. ON-DEMAND BACKUPS BY USING THE ROXCTL CLI

You can use the **roxctl** CLI to take the backups by using the **backup** command. You require either an API token or your administrator password to run this command.

1.1.1. On-demand backups by using an API token

You can back up the entire database of Red Hat Advanced Cluster Security for Kubernetes by using an API token.

Prerequisites

- You must have an API token with the **Admin** role.
- You must have installed the **roxctl** CLI.

Procedure

1. Set the **ROX_API_TOKEN** and the **ROX_CENTRAL_ADDRESS** environment variables:

```
$ export ROX_API_TOKEN=<api_token>
```

```
$ export ROX_CENTRAL_ADDRESS=<address>:<port_number>
```

2. Run the **backup** command:
   - For Red Hat Advanced Cluster Security for Kubernetes 3.0.55 or later:

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central backup
```

   - For Red Hat Advanced Cluster Security for Kubernetes 3.0.54 or older:

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central db backup
```

By default, the **roxctl** CLI saves the backup file in the directory in which you run the command. You can use the **--output** option to specify the backup file location.

Additional resources

- System roles

1.1.2. On-demand backups by using the administrator password

You can back up the entire database of Red Hat Advanced Cluster Security for Kubernetes by using your administrator password.

Prerequisites

- You must have the administrator password.
- You must have installed the **roxctl** CLI.

Procedure

1. Set the **ROX_CENTRAL_ADDRESS** environment variable:

```
$ export ROX_CENTRAL_ADDRESS=<address>:<port_number>
```

2. Run the **backup** command:
   - For Red Hat Advanced Cluster Security for Kubernetes 3.0.55 or later:

```
$ roxctl -p <admin_password> -e "$ROX_CENTRAL_ADDRESS" central backup
```

   - For Red Hat Advanced Cluster Security for Kubernetes 3.0.54 or older:

```
$ roxctl -p <admin_password> -e "$ROX_CENTRAL_ADDRESS" central db backup
```

By default, the **roxctl** CLI saves the backup file in the directory in which you run the command. You can use the **--output** option to specify the backup file location.

CHAPTER 2. RESTORING FROM A BACKUP

You can restore Red Hat Advanced Cluster Security for Kubernetes from an existing backup by using the **roxctl** command-line interface (CLI).

You can use the **roxctl** CLI to restore Red Hat Advanced Cluster Security for Kubernetes by using the **restore** command. You require either an API token or your administrator password to run this command.

2.1. RESTORING BY USING AN API TOKEN

You can restore the entire database of Red Hat Advanced Cluster Security for Kubernetes by using an API token.

Prerequisites

- You must have a Red Hat Advanced Cluster Security for Kubernetes backup file.
- You must have an API token with the administrator role.
- You must have installed the **roxctl** CLI.

Procedure

1. Set the **ROX_API_TOKEN** and the **ROX_CENTRAL_ADDRESS** environment variables:

```
$ export ROX_API_TOKEN=<api_token>
```

```
$ export ROX_CENTRAL_ADDRESS=<address>:<port_number>
```

2. Run the **restore** command:

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central db restore <backup_file>
```

2.2. RESTORING BY USING THE ADMINISTRATOR PASSWORD

You can restore the entire database of Red Hat Advanced Cluster Security for Kubernetes by using your administrator password.

Prerequisites

- You must have a Red Hat Advanced Cluster Security for Kubernetes backup file.
- You must have the administrator password.
- You must have installed the **roxctl** CLI.

Procedure

1. Set the **ROX_CENTRAL_ADDRESS** environment variable:

```
$ export ROX_CENTRAL_ADDRESS=<address>:<port_number>
```

2. Run the **restore** command:

```
$ roxctl -p <admin_password> -e "$ROX_CENTRAL_ADDRESS" central db restore
<backup_file>
```

2.3. RESUMING THE RESTORE OPERATION

During a restore operation, if your connection is interrupted or you need to go offline, you can resume the restore operation.

- If you do not have access to the machine running the resume operation, use the **roxctl central db restore status** command to check the status of an ongoing restore operation.
- In case of connection interruptions, the **roxctl** CLI automatically tries to restore a task when the connection becomes available. The automatic connection retries depend on the duration specified by the **timeout** option.
- Use the **--timeout** option to specify the time in seconds, minutes, or hours, after which the **roxctl** CLI stops trying to resume a restore operation. If not specified, the default timeout is 10 minutes (**10m**).
- If a restore operation is stuck or if you want to cancel it, use the **roxctl central db restore cancel** command to cancel an ongoing restore operation.
- If a restore operation is stuck, or you have canceled it, or it timed out, you can resume the previous restore by re-running the original command.

NOTE

- During interruptions, Red Hat Advanced Cluster Security for Kubernetes caches an ongoing restore operation for 24 hours. You can resume this operation by re-running the original restore command.
- The **--timeout** option only governs client-side connection retries and does not affect the 24 hours server-side restore cache.
- You cannot resume restore operations across restarts of the Central pod.
- If a restore operation is interrupted, you must restart it within 24 hours and before Central restarts, otherwise Red Hat Advanced Cluster Security for Kubernetes cancels the restore operation.