



Red Hat Advanced Cluster Security for Kubernetes 4.0

Backup and restore

Backing up and restoring Red Hat Advanced Cluster Security for Kubernetes

Red Hat Advanced Cluster Security for Kubernetes 4.0 Backup and restore

Backing up and restoring Red Hat Advanced Cluster Security for Kubernetes

Legal Notice

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

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	3
1.1. BACKING UP CENTRAL DATABASE BY USING THE ROXCTL CLI	3
1.1.1. On-demand backups by using an API token	3
1.1.2. On-demand backups by using the administrator password	4
1.2. BACKING UP CENTRAL DEPLOYMENT	4
1.2.1. Backing up deployment using the RHACS Operator	4
1.2.2. Backing up deployment using Helm	5
CHAPTER 2. RESTORING FROM A BACKUP	6
2.1. RESTORING CENTRAL DATABASE BY USING THE ROXCTL CLI	6
2.1.1. Restoring by using an API token	6
2.1.2. Restoring by using the administrator password	6
2.1.3. Resuming the restore operation	7
2.2. RESTORING CENTRAL DEPLOYMENT USING THE ROXCTL CLI	8
2.2.1. Restore certificates using the roxctl CLI	8
2.2.2. Running the Central installation scripts	8
2.3. RESTORE CENTRAL DEPLOYMENT USING THE RHACS OPERATOR	9
2.4. RESTORE CENTRAL DEPLOYMENT USING HELM	10
2.5. RESTORING CENTRAL TO ANOTHER CLUSTER OR NAMESPACE	10

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

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

You can configure automatic backups for the Central database by integrating with [Amazon S3](#) or [Google Cloud Storage](#). You can perform on-demand backups of the Central database by using the **roxctl** CLI. You can also back up your Central deployment using RHACS Operator or Helm Chart installation methods.

Depending on your requirements, you can create two types of backups:

1. A backup of the Central database: It includes RHACS configurations, resources, events, and certificates. In an unforeseen incident, such as database failure or data corruption, you can use the backup to recover and restore the Central database to its earlier functional state. Doing this ensures the availability and integrity of essential data, allowing you to continue normal operations without significant disruptions or loss of critical information.
2. A backup of all custom deployment configurations: If you installed RHACS by using Helm charts or the RHACS Operator, you can back up settings, parameters, and customizations specific to your installation. When the RHACS installation gets accidentally deleted, or you need to migrate it to another cluster or namespace, having a backup of the deployment configurations enables a seamless recovery process. In addition, by restoring the custom settings from the backup, you can efficiently reinstate your Central installation's unique requirements and configurations, ensuring consistent and exact deployment of the system.

Because backup files include secrets and certificates, you must securely store the backup files.

1.1. BACKING UP CENTRAL DATABASE BY USING THE ROXCTL CLI

Backing up the Central database is critical to ensure data integrity and system reliability. Regular backups of the database, containing necessary configurations, resources, events, and certificates, protect against database failures, corruption, and accidental data loss.

You can use the **roxctl** CLI to take the backups by using the **backup** command. You require 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:

```
$ roxctl -e "$ROX_CENTRAL_ADDRESS" central backup 1
```

- 1 You can use the **--output** option to specify the backup file location.

By default, the **roxctl** CLI saves the backup file in the directory where you run the command.

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:

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

1.2. BACKING UP CENTRAL DEPLOYMENT

You can back up the deployment of a Central instance. This can be useful if you want to migrate central to another namespace or cluster by using the same configuration values.



NOTE

Red Hat does not support backing up deployment configurations by using the **roxctl** CLI. You can use the **oc** or **kubectrl** CLI to back up manifests related to your Central instance and restore the configuration.

1.2.1. Backing up deployment using the RHACS Operator

When you use the RHACS Operator to instal RHACS, OpenShift Container Platform stores all the custom configuration for your Central deployment within the Central custom resource. You can backup the Central custom resource, the **central-tls** secret, and the administrator password. The **central-tls** secret includes the certificates for authenticating with Secured clusters and signing API tokens.

Procedure

1. Run the following command to save the Central custom resource in a YAML file:

```
$ oc get central -n _<central-namespace>_ _<central-name>_ -o yaml > central-cr.yaml
```

2. Run the following command to save **central-tls** in a JSON file:

```
$ oc get secret -n _<central-namespace>_ central-tls -o json | jq  
'del(.metadata.ownerReferences)' > central-tls.json
```

3. Run the following command to the administrator password in a JSON file:

```
$ oc get secret -n _<central-namespace>_ central-htpasswd -o json | jq  
'del(.metadata.ownerReferences)' > central-htpasswd.json
```

1.2.2. Backing up deployment using Helm

When you use the Helm chart to install RHACS, you store all the custom configuration for your Central deployment within the custom values that you apply to the Helm chart.

You can back up the custom values and save it in a YAML file.

Procedure

- Run the following command to back up custom Helm chart values in a YAML file:

```
$ helm get values --all -n _<central-namespace>_ _<central-helm-release>_ -o yaml >  
central-values-backup.yaml
```

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).

Depending upon your requirements and the data you have backed up, you can restore from the following types of backups:

1. **Restore Central database from the Central database backup** Use this to recover from a database failure or data corruption event. It allows you to restore and recover the Central database to its earlier functional state.
2. **Restore Central from the Central deployment backup** Use this if you are migrating Central to another cluster or namespace. This option restores the configurations of your Central installation.

2.1. RESTORING CENTRAL DATABASE BY USING THE ROXCTL CLI

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

2.1.1. 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>
```

2.1.2. 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>
```

2.1.3. 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.

2.2. RESTORING CENTRAL DEPLOYMENT USING THE ROXCTL CLI

You can restore your Central deployment to its original configuration by using the backups you made.

You must first restore certificates by using the **roxctl** CLI, and then restore the Central deployment by running the Central installation scripts.

2.2.1. Restore certificates using the roxctl CLI

Use the **roxctl** CLI to generate Kubernetes manifests to install the RHACS Central component to your cluster. Doing this allows you to ensure that authentication certificates for Secured clusters and the API tokens remain valid for the restored version. If you backed up another instance of RHACS Central, you can use the certificate files from that backup.



NOTE

With the **roxctl** CLI, you can not restore the entire Central deployment. Instead, first you use the **roxctl** CLI to generate new manifests using the certificates in your central data backup. Afterwards, you use those manifests to install Central.

Prerequisites

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

Procedure

1. Run the interactive install command:

```
$ roxctl central generate interactive
```

2. For the following prompt, enter the path of the Red Hat Advanced Cluster Security for Kubernetes backup file:

```
Enter path to the backup bundle from which to restore keys and certificates (optional):  
_<backup-file-path>_
```

3. For other following prompts, press **Enter** to accept the default value or enter custom values as required.

On completion, the interactive install command creates a folder named **central-bundle**, which has the necessary YAML manifests and scripts to deploy Central.

2.2.2. Running the Central installation scripts

After you run the interactive installer, you can run the **setup.sh** script to install Central.

Procedure

1. Run the **setup.sh** script to configure image registry access:

```
$ ./central-bundle/central/scripts/setup.sh
```

2. Create the necessary resources:

```
$ oc create -R -f central-bundle/central
```

3. Check the deployment progress:

```
$ oc get pod -n stackrox -w
```

4. After Central is running, find the RHACS portal IP address and open it in your browser. Depending on the exposure method you selected when answering the prompts, use one of the following methods to get the IP address.

Exposure method	Command	Address	Example
Route	oc -n stackrox get route central	The address under the HOST/PORT column in the output	https://central-stackrox.example.route
Node Port	oc get node -owide && oc -n stackrox get svc central-loadbalancer	IP or hostname of any node, on the port shown for the service	https://198.51.100.0:31489
Load Balancer	oc -n stackrox get svc central-loadbalancer	EXTERNAL-IP or hostname shown for the service, on port 443	https://192.0.2.0
None	central-bundle/central/scripts/port-forward.sh 8443	https://localhost:8443	https://localhost:8443



NOTE

If you have selected autogenerated password during the interactive install, you can run the following command to see it for logging into Central:

```
$ cat central-bundle/password
```

2.3. RESTORE CENTRAL DEPLOYMENT USING THE RHACS OPERATOR

You can restore your Central deployment to its original configuration by using the RHACS Operator. To successfully restore, you need the backup of your Central custom resource, **central-tls**, and the administrator password.

Prerequisites

- You must have the **central-tls** backup file.

- You must have the Central custom resource backup file.
- You must have the administrator password backup file.

Procedure

1. Use the **central-tls** backup file to create resources:

```
$ oc apply -f central-tls.json
```

2. Use the **central-htpasswd** backup file to create secrets:

```
$ oc apply -f central-htpasswd.json
```

3. Use the **central-cr.yaml** file to create the Central deployment:

```
$ oc apply -f central-cr.yaml
```

2.4. RESTORE CENTRAL DEPLOYMENT USING HELM

You can restore your Central deployment to its original configuration by using Helm. To successfully restore, you need the backup of your Central custom resource, the **central-tls** secret, and the administrator password.

Prerequisites

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

Procedure

1. Generate **values-private.yaml** from the RHACS database backup file:

```
$ roxctl central generate k8s pvc --backup-bundle _<path-to-backup-file>_ --output-format "helm-values"
```

2. Run the **helm install** command and specify your backup files:

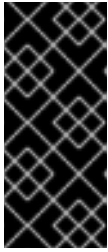
```
$ helm install -n stackrox --create-namespace stackrox-central-services rhacs/central-services -f central-values-backup.yaml -f central-bundle/values-private.yaml
```

2.5. RESTORING CENTRAL TO ANOTHER CLUSTER OR NAMESPACE

You can use the backups of the RHACS Central database and the deployment to restore Central to another cluster or namespace.

The following list provides a high-level overview of installation steps:

1. Depending upon your installation method, you must first restore Central deployment by following the instructions in the following topics:

**IMPORTANT**

- Make sure to use the backed-up Central certificates so that secured clusters and API tokens issued by the old Central instance remain valid.
 - If you are deploying to another namespace, you must change the namespace in backed-up resources or commands.
-
- [Restoring Central deployment using the **roxctl** CLI](#)
 - [Restore Central deployment using the RHACS Operator](#)
 - [Restore Central deployment using Helm](#)
2. Restore Central database by following the instruction in the [Restoring Central database by using the roxctl CLI](#) topic.
 3. If you have an external DNS entry pointing to your old RHACS Central instance, you must reconfigure it to point to the new RHACS Central instance that you create.