NetBackup™ Web UI Kubernetes 管理者ガイド

リリース 10.3

VERITAS™

最終更新日: 2023-12-28

法的通知と登録商標

Veritas、Veritas ロゴ、NetBackup は、Veritas Technologies LLC または関連会社の米国およびその他の国における商標または登録商標です。その他の会社名、製品名は各社の登録商標または商標です。

この製品には、Veritas 社がサードパーティへの帰属を示す必要があるサードパーティ製ソフトウェア (「サードパーティ製プログラム」) が含まれる場合があります。サードパーティプログラムの一部は、オープンソースまたはフリーソフトウェアライセンスで提供されます。本ソフトウェアに含まれる本使用許諾契約は、オープンソースまたはフリーソフトウェアライセンスでお客様が有する権利または義務を変更しないものとします。このVeritas製品に付属するサードパーティの法的通知文書は次の場所で入手できます。

https://www.veritas.com/about/legal/license-agreements

本書に記載されている製品は、その使用、コピー、頒布、逆コンパイルおよびリバースエンジニアリングを制限するライセンスに基づいて頒布されます。Veritas Technologies LLC からの書面による許可なく本書を複製することはできません。

本書は、現状のままで提供されるものであり、その商品性、特定目的への適合性、または不侵害の暗黙的な保証を含む、明示的あるいは暗黙的な条件、表明、および保証はすべて免責されるものとします。ただし、これらの免責が法的に無効であるとされる場合を除きます。Veritas Technologies LLC およびその関連会社は、本書の提供、パフォーマンスまたは使用に関連する付随的または間接的損害に対して、一切責任を負わないものとします。本書に記載の情報は、予告なく変更される場合があります。

ライセンスソフトウェアおよび文書は、FAR 12.212 に定義される商用コンピュータソフトウェアと見なされ、Veritasがオンプレミスまたはホスト型サービスとして提供するかを問わず、必要に応じて FAR 52.227-19「商用コンピュータソフトウェア - 制限される権利 (Commercial Computer Software - Restricted Rights)」、DFARS 227.7202「商用コンピュータソフトウェアおよび商用コンピュータソフトウェア文書 (Commercial Computer Software and Commercial Computer Software Documentation)」、およびそれらの後継の規制に定める制限される権利の対象となります。米国政府によるライセンス対象ソフトウェアおよび資料の使用、修正、複製のリリース、実演、表示または開示は、本使用許諾契約の条項に従ってのみ行われるものとします。

Veritas Technologies LLC
2625 Augustine Drive
Santa Clara, CA 95054

http://www.veritas.com

テクニカルサポート

テクニカルサポートはグローバルにサポートセンターを管理しています。すべてのサポートサービスは、サポート契約と現在のエンタープライズテクニカルサポートポリシーに応じて提供されます。サポート内容およびテクニカルサポートの利用方法に関する情報については、次の Web サイトにアクセスしてください。

https://www.veritas.com/support

次の URL で Veritas Account の情報を管理できます。

https://my.veritas.com

現在のサポート契約についてご不明な点がある場合は、次に示すお住まいの地域のサポート契約管理チームに電子メールでお問い合わせください。

世界共通 (日本を除く)	CustomerCare@veritas.com
日本	CustomerCare_Japan@veritas.com

マニュアル

マニュアルの最新バージョンがあることを確認してください。各マニュアルには、2 ページ目に最終更新日が記載されています。最新のマニュアルは、Veritasの Web サイトで入手できます。

https://sort.veritas.com/documents

マニュアルに対するご意見

お客様のご意見は弊社の財産です。改善点のご指摘やマニュアルの誤謬脱漏などの報告をお願いします。その際には、マニュアルのタイトル、バージョン、章タイトル、セクションタイトルも合わせてご報告ください。ご意見は次のアドレスに送信してください。

NB.docs@veritas.com

次の Veritas コミュニティサイトでマニュアルの情報を参照したり、質問したりすることもできます。

http://www.veritas.com/community/

Veritas Services and Operations Readiness Tools (SORT)

Veritas SORT (Service and Operations Readiness Tools) は、特定の時間がかかる管理タスクを自動化および簡素化するための情報とツールを提供するWeb サイトです。製品によって異なりますが、SORT はインストールとアップグレードの準備、データセンターにおけるリスクの識別、および運用効率の向上を支援します。SORT がお客様の製品に提供できるサービスとツールについては、次のデータシートを参照してください。

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

目次

1

NetBackup for Kubernetes の概要

この章では以下の項目について説明しています。

概要

NetBackup Web UI は、名前空間の形式で、Kubernetes アプリケーションのバックアップとリストアの機能を提供します。Kubernetes クラスタ内の保護可能な資産は NetBackup 環境内で自動的に検出され、管理者は必要なスケジュール、バックアップ、保持の各設定を含む 1 つ以上の保護計画を選択できます。

NetBackup Web UI では、次の操作を実行できます。

- 保護のための Kubernetes クラスタの追加
- 検出された名前空間の表示
- 役割の権限の管理
- リソース制限を設定してインフラとネットワークの負荷を最適化
- Kubernetes 資産を保護するための保護とインテリジェントグループの管理
- 名前空間と永続ボリュームを同じ、または代替の Kubernetes クラスタにリストアします。
- バックアップおよびリストア操作の監視
- イメージの有効期限、イメージのインポートおよびイメージのコピー操作

Kubernetes 用の NetBackup サポート機能

表 1-1 NetBackup for Kubernetes

機能	説明
自動 NetBackup Kubernetes エージェントの構成	Kubernetes クラスタが追加され、ストレージクラスとボリュームスナップショットクラスなどの構成が追加され、自動配備がサポートされた状態でデータムーバーの構成を実行できます。
NetBackup RBAC (役割ベースのアクセス制御) との統合	NetBackup Web UI は RBAC の役割を提供し、どの NetBackup ユーザーが NetBackup の Kubernetes 操作を管理できるかを制御します。ユーザーは Kubernetes 操作を管理するために NetBackup 管理者である必要はありません。
ライセンス	容量ベースのライセンス
保護計画	次の利点があります。 ■ 単一の保護計画を使用して、複数の Kubernetes 名前空間を保護します。複数のクラスタに資産を分散できます。 ■ Kubernetes 資産を保護するために、Kubernetes コマンドを知る必要はありません。
Kubernetes 資産のインテリジェントな管理	NetBackup は自動的に、Kubernetes クラスタ内の名前空間、永続ボリューム、永続ボリューム要求などを検出します。また、手動検出を実行できます。資産が検出されると、Kubernetes 作業負荷管理者は、資産を保護するために 1 つ以上の保護計画を選択できます。
Kubernetes 固有のクレデンシャル	クラスタの認証と管理に使用する Kubernetes サービスアカウント。
検出 ■ 完全検出 ■ 増分検出	[今すぐ検出 (Discover now)]オプションを使用した検出は常に完全検出です。 新しいクラスタが NetBackup に追加されたときの検出は常に完全検出です。 Kubernetes クラスタが追加されると、自動検出サイクルがトリガされ、Kubernetes クラスタで利用可能なすべての資産が検出されます。その日最初の自動検出は完全検出で、以降の自動検出は増分検出です。
バックアップ機能 ■ スナップショットのみのバックアップ ■ スナップショットからのバックアップ	バックアップでは次の機能を利用できます。 ■ バックアップは、NetBackup サーバーによって中央サイトから完全に管理されます。管理者は、さまざまな Kubernetes クラスタで、名前空間の自動的な無人バックアップをスケジュールできます。 ■ NetBackup Web UI は、1 つのインターフェースからの名前空間のバックアップとリストアをサポートします。 ■ 完全バックアップのバックアップスケジュールの構成。 ■ 手動バックアップとスナップショットのみのバックアップ。 ■ バックアップのパフォーマンスを向上させるための各クラスタのリソースのスロットル。 ■ NetBackup はスナップショット方式を使用して Kubernetes 名前空間のバックアップを実行し、リカバリ時間目標を短縮できます。

機能	説明
リストア機能 ■ スナップショットからのリストア ■ バックアップコピーからのリストア	リストアでは次の機能を利用できます。 ■ Kubernetes 名前空間と永続ボリュームを異なる場所にリストアします。 ■ 並列リストアジョブを含むバックアップコピーからのリストアを使用して、異なる Kubernetes クラスタフレーバーにリストアします。
クライアント側のデータ重複排除のサポート	Kubernetes でクライアント側のデータ重複排除のサポート機能が有効になっています。 詳しくは、『NetBackup™ 重複排除ガイド』の「クライアント側の重複排除について」セクションを参照してください。
自動イメージレプリケーション (AIR)	1 つの NetBackup Kubernetes クラスタで生成されたバックアップを、1 つ以上のターゲット NetBackup ドメインのストレージにレプリケートできます。これは AIR とも呼ばれます。他の NetBackup ドメインのストレージにバックアップをレプリケートする機能。
ステートフルアプリケーションの保護	永続ボリュームを使用して状態を保持する Kubernetes アプリケーションを保護できます。次の機能をサポートする CSI (Container Storage Interface) プロバイダにおけるモードファイルシステムまたはブロック (あるいはその両方) の PVC (永続ボリューム要求) のバックアップとリストア: ■ PVC スナップショット機能 ■ NFS (Network File System) または他の非ブロックストレージに基づく PVC ボリュームプロビジョニング ■ ボリュームが混在する (VolumeMode: ファイルシステムとブロック) 名前空間のバックアップとリストアは、NetBackup 10.3 以降でサポートされるようになりました。
インポートと検証	インポートは 2 段階の操作です。第 1 段階では、指定したメディア上のバックアップに対するカタログエントリが再作成されます。第 2 段階のインポートが完了すると、それらのイメージによってバックアップされたファイルのカタログエントリが作成されます。 検証: NetBackup では、NetBackup カタログに記録されたものと内容を比較して、バックアップの内容を検証できます。
Redhat プラットフォーム用の FIPS (連邦情報処理標準) サポート	NetBackup 10.3 では、RedHat プラットフォームで、Kubernetes の作業負荷が FIPS 準拠の通信をサポートできます。

2

NetBackup Kubernetes Operator の配備と構成

この章では以下の項目について説明しています。

NetBackup Kubernetes Operator を配備するための前提条件

NetBackup Kubernetes Operator を配備する前に、Helm Chart をインストールし、永続ボリューム用の領域を用意する必要があります。

Helm の最新バージョンをインストールするには、次のコマンドを実行します。

1. `$ curl -fsSL -o get_helm.sh` https://raw.githubusercontent.com/helm

2. `$ chmod 700 get_helm.sh`
3. `$ ./get_helm.sh`

メモ: NetBackup を配備する各クラスタにオペレータを配備する必要があります。

新しい Helm Chart をインストールするには

1 名前空間内のすべての Helm Chart を一覧表示するには、次のコマンドを実行します。

```
- helm list -n <namespace>
```

2 古いプラグインをアンインストールするには、次のコマンドを実行します。

- helm uninstall `<plugin-name> -n <namespace>`

3 新しいプラグインをインストールするには、次のコマンドを実行します。

- helm install `<plugin-name> <chart-path> -n <namespace>`

Helm Chart とツリー構造のレイアウトを次に示します。

```
netbackupkops-helm-chart/
├── charts
├── Chart.yaml
├── templates
│   ├── deployment.yaml
│   └── _helpers.tpl
└── values.yaml
```

ディレクトリ構造:

```
tar --list -f netbackupkops-10.3.tar.gz
veritas_license.txt
netbackupkops.tar
netbackupkops-helm-chart/
netbackupkops-helm-chart/Chart.yaml
netbackupkops-helm-chart/values.yaml
netbackupkops-helm-chart/.helmignore
netbackupkops-helm-chart/templates/
netbackupkops-helm-chart/templates/deployment.yaml
netbackupkops-helm-chart/templates/_helpers.tpl
netbackupkops-helm-chart/charts/
```

NetBackup Kubernetes Operator でのサービスパッケージの配備

Helm Chart の構成

Helm Chart を使用して、NetBackup Kubernetes Operator を配備できます。

NetBackup Kubernetes Operator をアップグレードするには、Helm Chart をアップグレードする必要があります。

メモ: 新しいプラグインをインストールする前に、古いプラグインをアンインストールする必要があります。

NetBackup Kubernetes Operator を配備するには:

1. ベリタスのサポート Web サイト (https://www.veritas.com/content/support) から tar パッケージをダウンロードします。
2. ホームディレクトリにパッケージを抽出します。`netbackupkops-helm-chart` フォルダは、ホームディレクトリに存在する必要があります。
3. すべてのクラスタコンテキストを一覧表示するには、コマンド `kubectl config get-contexts` を実行します。
4. オペレータサービスを配備するクラスタに切り替えるには、次のコマンドを実行します。

   ```
   kubectl config use-context <cluster-context-name>
   ```

5. 現在のディレクトリをホームディレクトリに変更するには、コマンド `cd ~` を実行します。
6. NetBackup は、OCI 標準に準拠したコンテナイメージリポジトリをサポートしています。オペレータとデータムーバーイメージをプッシュする任意のツールを使用できます。

   プライベート Docker レジストリを使用している場合は、この手順の指示に従って、NetBackup 名前空間に Secret `nb-docker-cred` を作成します。それ以外の場合は、次の手順にスキップします。

   - プライベート Docker レジストリにログオンするには、コマンド `docker login -u <user name><repo-name>` を実行します。
     ログイン後、認証トークンを含む `config.json` ファイルが作成または更新されます。`config.json` ファイルを表示するには、コマンド `cat ~/.docker/config.json` を実行します。
     出力は次のようになります。

```
{
   "auths": {
       "https://index.docker.io/v1/": {
           "auth": "c3R...zE2"
       }
   }
}
```

- NetBackup 名前空間で netbackupkops-docker-cred という名前の Secret を作成するには、次のコマンドを実行します。

  ```
  kubectl create secret generic netbackupkops-docker-cred ¥
  --from-file=.dockerconfigjson=.docker/config.json ¥
  --type=kubernetes.io/dockerconfigjson -n netbackup
  ```

  Secret を作成する名前空間は任意に指定できます。
- NetBackup 名前空間で Secret netbackupkops-docker-cred が作成されたかどうかを確認するには、次のコマンドを実行します。

  ```
  kubectl get secrets -n netbackup
  ```
- Docker キャッシュにイメージをロードして Docker イメージリポジトリにイメージをプッシュするには、次のコマンドを実行します。
  - NetBackup Kubernetes Operator の tar ファイルをロードします。

    ```
    <docker load -i <nameof the tar file> ./>
    ```
  - 要件に従って、ロードされた Docker イメージにタグを付けます。

    ```
    docker tag <imagename:tagof the loadedimage>
    <repo-name/image-name:tag-name>
    ```
  - NetBackup Kubernetes Operator の配備時に Kubernetes がイメージをフェッチできるリポジトリから、イメージをプッシュします。

    ```
    docker push <repo-name/image-name:tag-name>
    ```

メモ: この例では、Docker が参照用に使用されています。同等の機能を提供する他の CLI ツールを使用できます。

---

7 テキストエディタで netbackupkops-helm-chart/values.yaml を編集します。

- マネージャセクションのイメージの値を、イメージ名とタグ repo-name/image-name:tag-name に置き換えます。
- レプリカの値を 0 に変更します。

メモ: NetBackup Kubernetes Operator を構成する手動の手順に従って、レプリカを 0 に設定します。

8 メタデータ永続ボリュームのサイズ調整が必要です。Kubernetes Operator のデフォルトの永続ボリュームサイズは 10Gi です。永続ボリュームサイズは構成可能です。

プラグインを配備する前に、ストレージの値を 10Gi からより大きい値に変更できます。これにより、nbukops ポッドには、そのポッドでマウントされた PVC のサイズが設定されます。

メタデータの永続ボリュームサイズは values.yaml で指定できます。

helm-chart の deployment.yaml の永続ボリューム要求は次のようになります。

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  labels:
    component: netbackup
  name: {{ .Release.Namespace }}-netbackupkops
  namespace: {{ .Release.Namespace }}
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 10Gi
```

- 新規インストール時に Helm Chart を構成する際、netbackupkops-helm-chart の deployment.yaml で PVC ストレージのサイズを変更できます。これにより、初期の PVC サイズが作成されます。
- インストール後、PVC サイズの更新 (ダイナミックボリューム拡張) は一部のストレージベンダーによってサポートされます。詳しくは、https://kubernetes.io/docs/concepts/storage/persistent-volumes を参照してください。

メモ: 永続ボリュームのデフォルトサイズは、データを失うことなく、より大きい値にサイズ変更できます。ボリュームの拡張をサポートするストレージプロバイダを追加することをお勧めします。

9 NetBackup Kubernetes Operator サービスを配備するには、次のコマンドを実行します。

```
helm install <release name of the deployment>
./netbackupkops-helm-chart -n <namespace which runs NetBackup
operator service>
```

例: `helm install veritas-netbackupkops ./netbackupkops-helm-chart -n netbackup`

- 必要に応じて配備のリリース名を変更できます。
- NetBackup オペレータサービスと NetBackup を実行する名前空間を指定するには、-n オプションが必要です。

10 配備の状態を確認するには、次のコマンドを実行します。

```
helm list -n <namespace which runs NetBackup operator service >
```

例:

```
helm list -n netbackup
```

11 リリース履歴を確認するには、次のコマンドを実行します。

```
helm history veritas-netbackupkops -n
<namespace which runs NetBackup operator service>。
```

例:

```
helm history veritas-netbackupkops -n netbackup
```

Kubernetes Operator の配備のためのポート要件

次の表は、Kubernetes Operator を配備するためのポート要件を示しています。さまざまなホストの間にファイアウォールが存在する場合は、必要な通信ポートを開く必要があります。

表 2-1 NetBackup Kubernetes クラスタ環境で開く必要があるポート

ソース	ポート番号	宛先
プライマリサーバー	TCP ポート 443	Kubernetes クラスタ
メディアサーバー	TCP ポート 443 (NetBackup 10.0 で採用)。	Kubernetes クラスタ

メモ: Kubernetes API サーバーのポートが 443 からデフォルト以外のポート (通常は 6443 または 8443) に変更されていないことを Kubernetes の構成で確認します。

ソース	ポート番号	宛先
Kubernetes クラスタ	TCP ポート 443 (NetBackup バージョン 9.1 が該当、バージョン 10.0 以降は該当せず)。	プライマリサーバー

メモ: NetBackup Kubernetes Operator (KOps) と datamover ポッドの場合は追加要件があります (NetBackup 10.0 で採用)。

Kubernetes クラスタ	TCP ポート 1556 (アウトバウンド)	プライマリサーバー
Kubernetes クラスタ	TCP ポート 1556 (アウトバウンド)	メディアサーバー
Kubernetes クラスタ	耐性ネットワークを使用している場合は TCP ポート 13724 (双方向)。	プライマリサーバーとメディアサーバー

NetBackup Kubernetes Operator のアップグレード

Helm コマンドを使用して NetBackup Kubernetes Operator の配備をアップグレードできます。

NetBackup Kubernetes Operator をアップグレードするには:

1. 目的の、また最新の NetBackup Kubernetes Operator tar パッケージをダウンロードします。
2. NetBackup Kubernetes Operator をアップグレードするには、次のコマンドを実行します。
   - `helm upgrade <plugin-name> <chart-path> -n <namespace>`
     例: `helm upgrade veritas-netbackupkops ./netbackupkops-helm-chart -n netbackup`

NetBackup Kubernetes Operator の削除

クラスタから NetBackup Kubernetes Operator の配備を削除できます。

```
helm uninstall <plugin-name> -n <Netbackup Kubernetes Operator Namespace>
```

メモ: プラグインをアンインストールすると、スナップショットベースのバックアップに関するメタデータが含まれる、NetBackup Kubernetes Operator PVC も削除されます。

NetBackup Kubernetes Operator を削除すると、スナップショットメタデータもホストするメタデータボリュームが失われます。スナップショットがすでに実行されている場合、メタデータがないと、スナップショットコピーからのリストア操作は失敗します。

NetBackup 9.1 では、古いスナップショットを手動で削除してから、関連付けられた Velero スナップショットを削除する必要があります。

NetBackup 10.0 では、NetBackup 9.1 を使用して作成された Velero 管理スナップショットを期限切れにできません。バックアップイメージが NetBackup で期限切れになると、カタログは自動的にクリアされます。ただし、Kubernetes サーバー上のスナップショットは手動で削除する必要があります。

手動によるイメージの期限切れ操作について詳しくは、
https://www.veritas.com/content/support を参照してください。

メモ: スナップショットを期限切れにしない、または永続ボリュームスナップショットを削除しない場合、Kubernetes Operator をアンインストールすると、孤立したボリュームスナップショットが Kubernetes クラスタに残ります。

NetBackup Kubernetes データムーバーの構成

NetBackup Kubernetes 作業負荷のデータムーバーを構成する必要があります。データムーバーイメージの正しいバージョンをダウンロードします。

スナップショットからのバックアップとバックアップからのリストアをサポートするには、NetBackup Kubernetes Operator 名前空間を構成する必要があります。(バックアップコピー)。ご使用のリリースバージョンに対応した正しいバージョンのデータムーバーイメージ `veritasnetbackup-datamover-10.3.tar` をダウンロードセンターからダウンロードしてください。https://www.veritas.com/content/support を参照してください。

データムーバーを構成する方法

1. イメージレジストリにデータムーバーイメージをプッシュするには、次のコマンドを実行します。

   ```
   docker login -u <user name> <repo-name>
   ```

2. プロンプトでパスワードを入力します。ログイン済みの場合は、この手順をスキップします

3. `docker load -i <name of the datamover image file>` を実行します。

4. `docker tag <datamover image name:tag of the loaded datamover image> <repo-name/image-name:tag-name>` を実行します。

5 `docker push <repo-name/image-name:tag-name>`

メモ: この例では、Docker が参照用に使用されています。同等の機能を備えている CLI ツールを使用できます。

6 プライマリサーバー名の ConfigMap で、イメージ値が手順 4 でプッシュした *<repo-name/image-name:tag-name>* に設定されていることを確認します。

例:

```
apiVersion: v1
data:
  datamover.properties: image=<image-repo>/datamover:<datamover
tag>
  version: "1"
kind: ConfigMap
metadata:
  name: <Primary Server Name>
  namespace: <Netbackup Kubernetes Operator Namespace Name>
```

Configmap について詳しくは、『NetBackup Web UI Kubernetes 管理者ガイド』の「Kubernetes Operator でサポートされる構成パラメータ」セクションを参照してください。

Kubernetes 用の NetBackup 保護の自動構成

前提条件

Kubernetes 作業負荷で NetBackup を構成する前に、ポート 443、1556、および 13724 へのアクセス権を付与して NetBackup Server を実行する必要があります。

NetBackup の Kubernetes Operator とデータムーバーのイメージは、Kubernetes クラスタからアクセス可能なコンテナレジストリにアップロードする必要があります。

自動配備のために使用するシークレットを作成する必要があります。

NetBackup Web UI から新しい API キーを作成するには、次の操作を実行します。

1 [セキュリティ (Security)]、[アクセスキー (Access keys)]、[追加 (Add)]の順に移動します。ユーザー名を入力し、API キーの誤用を避けるために有効期間を 1 日に設定します。

2 Kubernetes クラスタで、次の内容を含む新しいシークレット nb-config-deploy-secret.yaml を作成します。

```
apiVersion: v1
kind: Secret
metadata:
  name: <kops-namespace>-nb-config-deploy-secret
  namespace: <kops-namespace>
type: Opaque
stringData:
  apikey: <Enter the value of API key from the earlier step>
```

3 シークレットを適用し、`kubectl apply -f nb-config-deploy-secret.yaml` コマンドを実行します

インストール前

1 `netbackupkops-helm-chart/values.yaml` の次のフィールドを編集します。

- containers.manager.image: NetBackup Kubernetes コントローライメージをプルするためのコンテナレジストリ URL
- imagePullSecrets: name: コンテナレジストリがイメージをプルするために認証を必要とする場合の、イメージプルシークレットの名前。
- nbprimaryserver: NetBackup プライマリサーバーの構成された名前。
- nbsha256fingerprint: NetBackup Web UI から sha256 の指紋をフェッチします。[セキュリティ (Security)]、[証明書 (Certificates)]、[認証局 (Certificate Authority)]の順に移動します。
- k8sCluster: Kubernetes クラスタ API サーバーの FQDN。
- k8sPort: Kubernetes API サーバーが一覧表示されるポート。

この情報は、Kubernetes クラスタの UI コンソールで利用可能です。

2 存在しない場合は、次のコマンドを実行して Kubernetes クラスタと Kubernetes ポートを取得します: `# kubectl cluster-info`。Kubernetes コントロールプレーンは https://<Kubernetes FQDN>:6443 で実行されます。

- datamoverimage: データムーバーイメージをプルするコンテナレジストリ URL。

- スナップショット操作とスナップショットからのバックアップ操作には、ストレージパラメータが必要です。ブロックまたはファイルシステムのストレージパラメータの少なくとも 1 つは必須です。

3 ストレージクラスを取得するには、次のコマンドを実行します。`# kubectl get storageclasses`

- storageclassblock: ブロックボリュームのプロビジョニングに使用されるストレージクラス。
- storageclassfilesystem: ファイルシステムボリュームのプロビジョニングに使用されるストレージクラス。

4 ボリュームスナップショットクラスを取得するには、次のコマンドを実行します。`# kubectl get volumesnapshotclasses`

- volumesnapshotclassblock: ブロックボリュームスナップショットを作成するためのボリュームスナップショットクラス。
- volumesnapshotclassfilesystem: ファイルシステムボリュームスナップショットを作成するためのボリュームスナップショットクラス。
- waitTimeBeforeCleanupMinutes: 成功した場合に構成の配備が削除されるまで待機する時間 (分単位)。最大値は 129600 (90 日) に設定できます。エラーが発生した場合、セキュリティトークンとクレデンシャルの NetBackup リソースは自動的に削除されますが、エラーをデバッグするために配備は実行されます。

インストール

helm をインストールするには、次のコマンドを実行します。`# helm install veritas-netbackupkops <path to netbackupkops-helm-chart> -n <kops namespace>`

デバッグ

Kubernetes Operator 名前空間から config-deploy ポッドを取得するには、次のコマンドを実行します。`# kubectl get pod -n <kops namespace> | grep "config-deploy"`

ログ

ポッド <namespace>-netbackup-config-deploy からログを確認するには、次のコマンドを実行します。`# kubectl logs <pod-name> -n <kops namespace>`

メモ: 詳しくは、『NetBackup Kubernetes クイックスタートガイド』リリース 10.3 を参照してください。

NetBackup スナップショット操作の設定を行う

永続ボリューム要求を保護するには、ストレージクラスとボリュームスナップショットクラスを構成する必要があります。NetBackup Kubernetes Operator をインストールした後、次の構成を行う必要があります。

1. CSI プラグインを指すストレージクラスを特定するか、ストレージクラスが存在しない場合は、CSI プラグインを指す新しいストレージクラスを定義します。
2. CSIプラグインを指すボリュームスナップショットクラスを特定するか、ボリュームスナップショットクラスが存在しない場合は、CSI プラグインを指す新しいボリュームスナップショットクラスを定義します。

   CSI ドライバの詳細で構成される *VolumeSnapshotClass* クラスを定義します。
3. Kubernetes クラスタで CSI ストレージクラスにラベルを付けます。
   - ストレージクラスのラベル *netbackup.veritas.com/default-csi-storage-class=true* は、ストレージクラスが raw ブロック (volumeMode=Block) に基づいてボリュームをプロビジョニングする場所のラベル付けに使用されます。
   - ストレージクラスのラベル *netbackup.veritas.com/default-csi-filesystem-storage-class=true* は、ストレージクラスがファイルシステム (volumeMode=FileSystem) に基づいてボリュームをプロビジョニングする場所のラベル付けに使用されます。

   メモ: 1 つのストレージクラスに両方のラベルを追加できます。ただしこれはストレージクラスが、raw ブロックを活用できるブロックボリュームと、ファイルシステムボリュームをサポートしている場合です。
4. NetBackup を使用する CSI ボリュームスナップショットクラスへのラベル付け
   - CSI ボリュームスナップショットクラスのラベル *netbackup.veritas.com/default-csi-storage-class=true* を、スナップショット操作に使用するすべての CSI ボリュームスナップショットクラスに追加する必要があります。

メモ: 永続ボリュームで構成される名前空間のスナップショットが失敗し、エラーメッセージ[Kubernetes 名前空間のスナップショットの作成に失敗しました。(Failed to create snapshot of the Kubernetes namespace.)]が表示されます。スナップショット操作は、有効な volumesnapshotclass ラベルが付いたドライバの有効なボリュームスナップショットクラスが見つからないなど、複数の原因によって失敗する場合があります。

Kubernetes Operator でサポートされる構成パラメータ

メモ: 構成値を取得するには、コマンド `kubectl get configmaps <namespace>-backup-operator-configuration -n <namespace> -o yaml > {local.file}` を実行します。

表 2-2 Kubernetes Operator でサポートされる、<namespace>-backup-operator-configuration の構成パラメータ

構成	説明	デフォルト値	指定可能な値
daemonsets	値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	true	true、false
deployments	値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	true	true、false
pods	値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	true	true、false

構成	説明	デフォルト値	指定可能な値
replicasets	値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	true	true、false
secrets	値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	true	true、false
services	値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	true	true、false

構成	説明	デフォルト値	指定可能な値
namespace	Kubernetes Operator は名前空間に配備されます。 値が false に設定されている場合、このリソースは検出され、バックアップされます。ただし、NetBackup Web UI で Kubernetes 名前空間をクリックした場合は、[リソース (Resource)]セクションにリソースは表示されません。	名前空間に指定された任意の名前。	NetBackup の名前空間。
cleanStaleCRDurationMinutes	古い CR をクリーニングするために CR ジョブが呼び出されてからの期間。古いカスタムリソースのクリーンアップジョブがトリガされてからの間隔。 長時間実行されるスナップショットからのバックアップとバックアップジョブからのリストアでは、cleanStaleCRDurationMinutes 値を増やす必要があります。	1,440 分	分単位の任意の値
ttlCRDurationMinutes	TTL CR の期間	30,240 分	30,240 分
fipsMode	NetBackup Kubernetes Operator とデータムーバーで FIPS_MODE を有効にする構成。	DISABLE	ENABLE、DISABLE
livesnessProbeInitialDelay	精査の初期遅延期間。	60 秒	60 分

構成	説明	デフォルト値	指定可能な値
livenessProbeTimeoutInSeconds	読み込まれたマシンでは、Liveness Probe の実行に 1 秒より長くかかる場合があります。ユーザーはこの値を大きくして、Liveness Probe のタイムアウトエラーを原因とする障害を回避できます。	1 秒	1 秒から 5 秒。
livenessProbePeriodInSeconds	精査の期間。	180 秒	秒単位の任意の値
checkNbcertdaemonStatusDurationMinutes	NB 証明書デーモンの状態の期間。	1,440 分	1,440 分
collectDataMoverLogs	datamover ログの収集ではメモリ使用率が高くなるため、ポッドのデバッグ、トラブルシューティング、再起動を行う場合のみログを有効にすることをお勧めします。 datamover のログを有効にする前に、NetBackup Kubernetes ポッド用のメモリ上限を 2 GB 以上に増やしてください。デバッグまたはトラブルシューティングの完了後は、以前の値またはデフォルト値にリセットできます。 **メモ:** 失敗したジョブの場合にのみ、datamover ログを収集するための詳細なサポートが提供されます。これにより、より詳細なレベルとして All/FailedOnly/Off が提供されます。	Failed	All、Failed、None

構成	説明	デフォルト値	指定可能な値
maxRetentionDataMoverLogsInHours	datamover ログの最大保持期間。	24 時間	72 時間
maxRetentionDataMoverInHours	指定した時間より古い datamover リソースがすべて削除されます。	24 時間	時間単位の任意の値
cleanStaleCertFilesDurationMinutes	古い証明書ファイルのクリーンアップジョブがトリガされてからの間隔。	60 分	1440 分
maxRetentionInDiscoveryCacheHours	検出キャッシュを保持する時間間隔を決定する時間。	24 時間	48 時間
pollingTimeoutInMinutes	期限切れになり失敗するまで再試行し続けるタイムアウトです。	15 分	分単位の任意の値
pollingFrequencyInSecs	ポーリング間隔。	5 秒	秒単位の任意の値
nbcertPrerequisteDirectoryAndFiles	NBCA の前提条件。	証明書名	証明書名

スナップショットからのバックアップ操作とバックアップからのリストア操作の前提条件

1. NetBackup での使用に有効なストレージクラスにラベルを付けます。ストレージクラスでサポートされているボリュームモード (ブロック/ファイルシステム) に基づいて、次のラベルを追加します。

   - ファイルシステムベースの永続ボリューム要求プロビジョニングストレージクラスの場合
     veritas.com/default-csi-filesystem-storage-class=true
   - ブロックベースの永続ボリューム要求プロビジョニングストレージクラスの場合
     veritas.com/default-csi-storage-class=true

   NetBackup のラベルが付いたストレージクラスが見つからない場合は、スナップショットからのバックアップとバックアップコピーからのリストアは失敗し、エラーメッセージ *[対象となるストレージクラスが見つかりません (No eligible storage class found)]* が表示されます。

   ストレージクラスにラベル付けするには、例に示されている次のコマンドを実行します。

例 1. コマンド `# kubectl get sc` を実行します。

名前	プロビジョナ
ocs-storagecluster-ceph-rbd (デフォルト)	openshift-storage.rbd.csi.ceph.com
ocs-storagecluster-ceph-rgw	openshift-storage.ceph.rook.io/bucket
ocs-storagecluster-ceph-rbd	openshift-storage.cephfs.csi.ceph.com
openshift-storage.noobaa.io	openshift-storage.noobaa.io/obc
thin	kubernetes.io/vsphere-volume

再生利用ポリシー	ボリュームバインドモード	ボリュームの拡張を許可する	経過時間
削除	即時	True	2 日 2 時間
削除	即時	False	2 日 2 時間
削除	即時	True	2 日 2 時間
削除	即時	False	2 日 2 時間
削除	即時	False	19 時間

例 2. コマンド `# kubectl get sc ocs-storagecluster-ceph-rbd --show-labels` を実行します。

名前	プロビジョナ	再生利用ポリシー
ocs-storagecluster-ceph-rbd (デフォルト)	openshift-storage.rbd.csi.ceph.com	削除

ボリュームバインドモード	ボリュームの拡張を許可する	経過時間	ラベル
即時	True	2 日 2 時間	netbackup.veritas.com/default-csi-storage-class=true

例 3. コマンド `oc label storageclass ocs-storagecluster-cephfs netbackup.veritas.com/default-csi-storage-class=true` を実行します。

```
storageclass.storage.k8s.io/ocs-storagecluster-cephfs labeled
```

例 4. コマンド `kubectl get sc ocs-storagecluster-cephfs --show-labels` を実行します。

名前	プロビジョナ	再生利用ポリシー
ocs-storagecluster-cephfs	openshift-storage.cephfs.csi.ceph.com	削除

ボリュームバインドモード	ボリュームの拡張を許可する	経過時間	ラベル
即時	True	2 日 2 時間	netbackup.veritas.com/default-csi-storage-class=true

2. NetBackup を使用するために有効なボリュームスナップショットクラスにラベルを付け、ラベル *netbackup.veritas.com/default-csi-volume-snapshot-class=true* を追加します。NetBackup のラベルが付いた *VolumeSnapshotClass* クラスが見つからない場合は、メタデータイメージのスナップショットからのバックアップジョブとリストアジョブが失敗し、エラーメッセージ *[Kubernetes 名前空間のスナップショットの作成に失敗しました (Failed to create snapshot of the Kubernetes namespace)]* が表示されます。

   ボリュームスナップショットクラスにラベル付けするには、例に示されている次のコマンドを実行します。

   例 1. コマンド `# kubectl get volumesnapshotclass` を実行します。

名前	ドライバ
ocs-storagecluster-cephfsplugin-snapclass	openshift-storage.cephfs.csi.ceph.com
ocs-storagecluster-rbdplugin-snapclass	openshift-storage.rbd.csi.ceph.co

削除ポリシー	経過時間
削除	2 日 2 時間
削除	2 日 2 時間

   例 2. コマンド `# kubectl get volumesnapshotclass ocs-storagecluster-cephfsplugin-snapclass --show-labels` を実行します。

名前	ドライバ
ocs-storagecluster-cephfsplugin-snapclass	openshift-storage.cephfs.csi.ceph.com

削除ポリシー	経過時間
削除	2 日 2 時間

例 3. コマンド `# kubectl label volumesnapshotclass ocs-storagecluster-cephfsplugin-snapclass netbackup.veritas.com/default-csi-volume-snapshot-class=true` を実行します。

```
volumesnapshotclass.snapshot.storage.k8s.io/ocs-storagecluster-cephfsplugin-snapclass
 labeled
```

例 4. コマンド `# kubectl get volumesnapshotclass ocs-storagecluster-cephfsplugin-snapclass --show-labels` を実行します。

名前	ドライバ
ocs-storagecluster-cephfsplugin-snapclass	openshift-storage.cephfs.csi.ceph.com

削除ポリシー	経過時間	ラベル
削除	2 日 2 時間	netbackup.veritas.com/default-csi-volume-snapshot-class=true

3. スナップショットからのバックアップ操作とバックアップコピーからのリストア操作を実行する各プライマリサーバーは、プライマリサーバーの名前を使用して個別の *ConfigMap* を作成する必要があります。

   次の `configmap.yaml` の例では、

   - backupserver.sample.domain.com と mediaserver.sample.domain.com は、NetBackup プライマリサーバーとメディアサーバーのホスト名です。
   - IP: 10.20.12.13 と IP: 10.21.12.13 は、NetBackup プライマリサーバーとメディアサーバーの IP アドレスです。

```
apiVersion: v1
data:
  datamover.hostaliases: |
        10.20.12.13=backupserver.sample.domain.com
        10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
        image=reg.domain.com/datamover/image:latest
  version: "1"
kind: ConfigMap
metadata:
```

```
name: backupserver.sample.domain.com
namespace: kops-ns
```

- configmap.yaml ファイルの詳細をコピーします。
- テキストエディタを開き、yaml ファイルの詳細を貼り付けます。
- その後、それに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

4. datamover.properties: image=reg.domain.com/datamover/image:latest に適切な datamover イメージを指定します。
5. プライマリサーバーとプライマリサーバーに接続されているメディアサーバーで短縮名が使用されていて、datamover からのホストの解決が失敗している場合は、datamover.hostaliases を指定します。プライマリサーバーとメディアサーバーのすべてのホスト名と IP のマッピングを指定します。
6. プライベート Docker レジストリを使用するには、「NetBackup Kubernetes Operator でのサービスパッケージの配備」セクションのポイント 6 の説明に従って、シークレットを作成します。

   シークレットが作成されたら、configmap.yaml ファイルの作成中に次の属性を追加します。

   ```
   datamover.properties: |
   image=repo.azurecr.io/netbackup/datamover:10.0.0049
   imagePullSecret=secret_name
   ```

7. configmap.yaml ファイルを作成するには、コマンド kubectl create -f configmap.yaml を実行します。
8. Kubernetes Operator が短縮名に基づいてプライマリサーバーを解決できない場合
   - 証明書のフェッチ中に*[EXIT STATUS 8500: Web サービスとの接続が確立されませんでした (EXIT STATUS 8500: Connection with the web service was not established)]*というメッセージが表示された場合は、ホスト名の解決の状態を nbcert ログで確認します。
   - ホスト名の解決に失敗した場合は、次の手順を実行します。
     kops deployment.yaml を更新し、配備に *hostAliases* を追加します。
   - 次の hostAliases の例では、
     - backupserver.sample.domain.com と mediaserver.sample.domain.com は、NetBackup プライマリサーバーとメディアサーバーのホスト名です。
     - IP: 10.20.12.13 と IP: 10.21.12.13 は、NetBackup プライマリサーバーとメディアサーバーの IP アドレスです。

```
hostAliases:
- hostnames:
  - backupserver.sample.domain.com
  ip: 10.20.12.13
- hostnames:
  -  mediaserver.sample.domain.com
  ip: 10.21.12.13
```

hostAliases の例の詳細をコピーしてテキストエディタに貼り付け、配備で hostAliases に追加します。

メモ: hostAliases セクションは、デフォルトの `./netbackupkops-helm-chart/templates/deployment.yaml` ファイルの 2210 行に追加する必要があります。

---

hostAliases の例:

```
2104 hostAliases;
- ip:10.15.206.7
hostnames:
- lab02-linsvr-01.demo.sample.domain.com
- lab02-linsvr-01
- ip:10.15.206.8
hostnames:
- lab02-linsvr-02.demo.sample.domain.com
- lab02-linsvr-02
imagePullSecrets:
- name:  {{ .values.netbackupKops.imagePullSecrets.name}}
```

9. 指紋と認証トークンを使用して Secret を作成します。

   シークレットと backupservercert の作成について詳しくは、『NetBackup™ Web UI Kubernetes 管理者ガイド』の NetBackup Kubernetes Operator での証明書の配備に関するセクションを参照してください。

10. 証明書をフェッチするための backupservercert 要求を作成します。

    詳しくは、『NetBackup™ Web UI Kubernetes 管理者ガイド』の「NetBackup Kubernetes Operator での証明書の配備」を参照してください。

    詳しくは『NetBackup™ セキュリティおよび暗号化ガイド』を参照してください。

メモ: これは、スナップショットからのバックアップとバックアップコピーからリストアを正常に実行するために必須の手順です。

Kubernetes でサポートされる DTE クライアント設定

DTE_CLIENT_MODE オプションは、バックアップサーバー固有の configmap によって datamover に設定される移動中のデータの暗号化 (DTE) モードを指定します。バックアップイメージの移動中のデータの暗号化は、グローバル DTE モードとクライアント DTE モードに基づいて実行されます。

バックアップサーバー固有の configmap を更新し、DTE_CLIENT_MODE キーを追加します。このキーは次の値を取ることができます。

- AUTOMATIC
- ON
- OFF

DTE_CLIENT_MODE について詳しくは、『Veritas NetBackup 管理者ガイド Vol. 1』の「クライアント用 DTE_CLIENT_MODE」のセクションを参照してください。

次に、DTE_CLIENT_MODE 設定を追加した configmap を示します。

```
apiVersion: v1
data:
  datamover.hostaliases: |
        10.20.12.13=backupserver.sample.domain.com
        10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
        image=reg.domain.com/datamover/image:latest
        DTE_CLIENT_MODE=ON
  version: "1"
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: kops-ns
```

datamover プロパティのカスタマイズ

キーと値のペアをバックアップサーバー固有の configmap に渡すことによって、datamover プロパティをカスタマイズできます。

表 2-3 datamover のプロパティ

キー名	指定可能な値
VXMS_VERBOSE	範囲: [0,99]
VERBOSE	範囲: [0,5]

キー名	指定可能な値
DTE_CLIENT_MODE	■ AUTOMATIC ■ ON ■ OFF

configmap を更新するには、次のようにキーと値のペアを追加します。

```
apiVersion: v1
data:
  datamover.properties: |
        image=reg.domain.com/datamover/image:latest
        VERBOSE=5
        DTE_CLIENT_MODE=OFF
        VXMS_VERBOSE=5
  version: "1"
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: kops-ns
```

短縮名の付いた NetBackup サーバーのトラブルシューティング

1 NetBackup Kubernetes Operator が短縮名を基にバックアップサーバーまたはメディアサーバーを解決できない場合は、次の手順を実行します。

- 証明書のフェッチ中に *[EXIT STATUS 8500: Web サービスとの接続が確立されませんでした (EXIT STATUS 8500: Connection with the web service was not established)]* というメッセージが表示された場合。次に、ホスト名解決が成功したかどうかを nbcert ログから確認します。失敗した場合は、次の手順を実行します。
- Kubernetes Operator の `deployment.yaml` を更新し、その配備に `hostAliases` を追加します。
- 次の `hostAliases` の例では、
  - backupserver.sample.domain.com と mediaserver.sample.domain.com は、NetBackup プライマリサーバーとメディアサーバーのホスト名です。
  - IP: 10.20.12.13 と IP: 10.21.12.13 は、NetBackup プライマリサーバーとメディアサーバーの IP アドレスです。

```
hostAliases:
- hostnames:
  - backupserver.sample.domain.com
  ip: 10.20.12.13
- hostnames:
  -  mediaserver.sample.domain.com
  ip: 10.21.12.13
```

hostAliases の例の詳細をコピーしてテキストエディタに貼り付け、配備で hostAliases に追加します。

2 データムーバーがバックアップサーバーまたはメディアサーバーの短縮名を解決できない場合。この問題を解決するには、次の手順を実行します。

- バックアップサーバー名を使用して configmap を更新します。
- *datamover.hostaliases* フィールドを追加し、ホスト名に IP アドレスをマップします。
- 次の `configmap.yaml` の例では、
  - backupserver.sample.domain.com と mediaserver.sample.domain.com は、NetBackup プライマリサーバーとメディアサーバーのホスト名です。
  - IP: 10.20.12.13 と IP: 10.21.12.13 は、NetBackup プライマリサーバーとメディアサーバーの IP アドレスです。

```
apiVersion: v1

data:
  datamover.hostaliases: |
        10.20.12.13=backupserver.sample.domain.com
        10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
        image=reg.domain.com/datamover/image:latest
  version: "1"
kind: configmap
metadata:
  name: backupserver.sample.domain.com
  namespace: kops-ns
```

- `configmap.yaml` ファイルの詳細をコピーします。
- テキストエディタを開き、yaml ファイルの詳細を貼り付けます。
- その後、それに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

- configmap.yaml ファイルを作成するには、コマンド kubectl create -f configmap.yaml を実行します。
- すでに作成されている configmap.yaml を更新する場合は、コマンドを実行して configmap を更新します。kubectl apply -f configmap.yaml

datamover ポッドのスケジュールメカニズムのサポート

バックアップサーバーの ConfigMap で次のフィールドを指定して、ノード上の datamover ポッドをスケジュールします。

1. nodeSelector: nodeSelector は、ポッドを特定のラベルを持つノードに制約する簡単な方法です。

   例:

```
apiVersion: v1

kind: ConfigMap

metadata:

  name: backupserver.sample.domain.com

  namespace: netbackup

data:

  datamover.hostaliases: |

    10.20.12.13=backupserver.sample.domain.com

    10.21.12.13=mediaserver.sample.domain.com

  datamover.properties: |

    image=reg.domain.com/datamover/image:latest

  datamover.nodeSelector: |

    kubernetes.io/hostname: test1-l94jm-worker-k49vj

    topology.rook.io/rack: rack1
```

```
  version: "1"
```

2. nodeName: nodeName は、親和性または nodeSelector よりも直接的にノードを選択する形式です。ポッドがバックアップ用にスケジュールされているノードを指定でき、デフォルトのスケジュールメカニズムを上書きできます。

   例:

```
apiVersion: v1

kind: ConfigMap

metadata:

  name: backupserver.sample.domain.com

  namespace: netbackup

data:

  datamover.hostaliases: |

    10.20.12.13=backupserver.sample.domain.com

    10.21.12.13=mediaserver.sample.domain.com

  datamover.properties: |

    image=reg.domain.com/datamover/image:latest

  datamover.nodeName : test1-l94jm-worker-hbblk

  version: "1"
```

3. Taint と Toleration: Toleration を使用すると、類似する Taint を使用してポッドをスケジュールすることができます。Taint と Toleration を連携して使用することで、ポッドを適切なノードに確実にスケジュールできます。1 つ以上の Taint がノードに適用される場合を考えます。そのノードは、Taint を容認 (tolerate) しないポッドを受け入れることはできません。

   例:

```
apiVersion: v1
```

```
kind: ConfigMap

metadata:

  name: backupserver.sample.domain.com

  namespace: netbackup

data:

  datamover.hostaliases: |

    10.20.12.13=backupserver.sample.domain.com

    10.21.12.13=mediaserver.sample.domain.com

  datamover.properties: |

    image=reg.domain.com/datamover/image:latest

  datamover.tolerations: |

    - key: "dedicated"

      operator: "Equal"

      value: "experimental"

      effect: "NoSchedule"

  version: "1"
```

4. 親和性と反親和性: ノード親和性は nodeSelector フィールドのように機能しますが、より表現力が高く、柔軟なルールを指定できます。ポッド間の親和性/反親和性を使用すると、他のポッドのラベルに対してポッドを制約できます。

   例:

   - ノード親和性:

     ```
     apiVersion: v1

     kind: ConfigMap
     ```

```
metadata:
  name: backupserver.sample.domain.com
  namespace: netbackup
data:
  datamover.hostaliases: |
    10.20.12.13=backupserver.sample.domain.com
    10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
    image=reg.domain.com/datamover/image:latest
  datamover.affinity: |
    nodeAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
        nodeSelectorTerms:
        - matchExpressions:
          - key: kubernetes.io/hostname
            operator: In
            values:
            - test1-l94jm-worker-hbblk
      preferredDuringSchedulingIgnoredDuringExecution:
      - weight: 1
        preference:
          matchExpressions:
```

```
          - key: beta.kubernetes.io/arch
            operator: In
            values:
            - amd64
  version: "1"
```

- ポッド親和性

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: netbackup
data:
  datamover.hostaliases: |
    10.20.12.13=backupserver.sample.domain.com
    10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
    image=reg.domain.com/datamover/image:latest
  datamover.affinity: |
    podAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
      - labelSelector:
```

```
            matchExpressions:
            - key: component
              operator: In
              values:
              - netbackup
          topologyKey: kubernetes.io/hostname
    version: "1"
```

5. topologySpreadConstraints:トポロジー分散制約は、地域、ゾーン、ノード、その他のユーザー定義トポロジードメインなどの障害ドメイン間でクラスタ全体に分散するポッドの動作を制御するために使用されます。

   例:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: backupserver.sample.domain.com
  namespace: netbackup
data:
  datamover. hostaliases: |
    10.20.12.13=backupserver.sample.domain.com
    10.21.12.13=mediaserver.sample.domain.com
  datamover.properties: |
    image=reg.domain.com/datamover/image:latest
  datamover.topologySpreadConstraints : |
```

```
  - maxSkew: 1

    topologyKey: kubernetes.io/hostname

    whenUnsatisfiable: DoNotSchedule

version: "1"
```

- ラベル: ラベルは、ポッドなどのオブジェクトに関連付けられているキーと値のペアです。ラベルは、オブジェクトの重要でユーザーに関連する属性を識別することを意図しています。ラベルを使用することで、オブジェクトのサブセットを整理して選択できます。ラベルは作成時にオブジェクトにアタッチでき、その後いつでも追加および変更できます。
  例:

```
apiVersion: v1

kind: ConfigMap

metadata:

  name: backupserver.sample.domain.com

  namespace: netbackup

data:

  datamover.hostaliases: |

    10.20.12.13=backupserver.sample.domain.com

    10.21.12.13=mediaserver.sample.domain.com

  datamover.properties: |

    image=reg.domain.com/datamover/image:latest

  datamover.labels: |

    env: test

    pod: datamover

  version: "1"
```

- 注釈: ユーザーはラベルまたは注釈を使用して、Kubernetes オブジェクトにメタデータをアタッチできます。注釈を使用してオブジェクトを識別および選択することはできません。
  例:

```
apiVersion: v1

kind: ConfigMap

metadata:

  name: backupserver.sample.domain.com

  namespace: netbackup

data:

  datamover.hostaliases: |

    10.20.12.13=backupserver.sample.domain.com

    10.21.12.13=mediaserver.sample.domain.com

  datamover.properties: |

    image=reg.domain.com/datamover/image:latest

  datamover.annotations: |

    buildinfo: |-

      [{

            "name": "test",

            "build": "1"

      }]

    imageregistry: "https://reg.domain.com/"

  version: "1"
```

3

NetBackup Kubernetes Operator での証明書の配備

この章では以下の項目について説明しています。

Kubernetes Operator での証明書の配備

datamover と NetBackup メディアサーバー間で安全に通信するために証明書を配備する必要があります。

メモ: スナップショットからのバックアップ操作とバックアップからのリストア操作を実行する前に、証明書を配備する必要があります。

BackupServerCert を作成する前に、クラスタの追加と検出が正常に行われる必要があります。これは、状態を成功として設定するために、NetBackup がいくつかの `clusterInfo` を渡すことに依存しているためです。

---

datamover 通信でサポートされる証明書

datamover は、NetBackup 環境内のデータ移動を容易にし、TLS (Transport Layer Security) を介してメディアサーバーと通信します。詳しくは、『NetBackup™ セキュリティおよび暗号化ガイド』の「NetBackup での安全な通信について」セクションを参照してく

ださい。datamover が通信するためには、ホスト ID ベースの証明書、または NetBackup プライマリサーバーによって発行され、ECA が署名した証明書が必要です。NBCA (NetBackup 認証局) または ECA (外部認証局) モードで証明書配備操作を実行できるように、新しいカスタムリソース定義 BackupServerCert が導入されました。

カスタムリソースの指定は次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupservercert-sample-nbca
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: primary.server.sample.com
  certificateOperation: Create | Update | Remove
  certificateType: NBCA | ECA
  nbcaAttributes:
    nbcaCreateOptions:
      secretName: "Secret name consists of token and fingerprint"
    nbcaUpdateOptions:
      secretName: "Secret name consists of token and fingerprint"
      force: true | false
    nbcaRemoveOptions:
      hostID: "hostId of the nbca certificate. You can view on Netbackup UI"
  ecaAttributes:
    ecaCreateOptions:
      ecaSecretName: "Secret name consists of cert, key, passphrase, cacert"
      copyCertsFromSecret: true | false
      isKeyEncrypted: true | false
    ecaUpdateOptions:
      ecaCrlCheck: DISABLE | LEAF | CHAIN
      ecaCrlRefreshHours: [0,4380]
```

ホスト ID ベースの証明書操作の実行

プライマリサーバーが NBCA モードで構成されていることを確認します。NBCA モードがオンかどうかを確認するには、コマンド `/usr/openv/netbackup/bin/nbcertcmd -getSecConfig -caUsage` を実行します。

出力は次のようになります。

```
NBCA: ON
ECA: OFF
```

ホスト ID ベースの証明書の指定は次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupservercert-sample
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: primaryserver.sample.domain.com
  certificateOperation: Create | Update | Remove
  certificateType: NBCA
  nbcaAttributes:
    nbcaCreateOptions:
      secretName: "Secret name consists of token and fingerprint"
    nbcaUpdateOptions:
      secretName: "Secret name consists of token and fingerprint"
      force: true
    nbcaRemoveOptions:
      hostID: "hostId of the nbca certificate. You can view on
Netbackup UI"
```

表 3-1 ホスト ID ベースの証明書操作

操作形式	オプションとコメント
作成 (Create)	secretName: トークンと指紋を含む Secret の名前。
削除 (Remove)	hostID: NBCA 証明書のホスト ID。
更新 (Update)	secretName: トークンと指紋を含む Secret の名前。

Kubernetes Operator 用ホスト ID ベースの証明書の作成

次の手順を使用して、Kubernetes Operator 用にホスト ID ベースの証明書を作成できます。

Kubernetes Operator 用ホスト ID ベースの証明書を作成するには

1. バックアップサーバーで、次のコマンドを実行し、SHA-256 指紋を取得します。

   ```
   /usr/openv/netbackup/bin/nbcertcmd -listCACertDetails
   ```

2. 認証トークンを作成するには、『NetBackup™ セキュリティおよび暗号化ガイド』の「認証トークンの作成」を参照してください。

3. 再発行トークンを作成するには、必要に応じて、『NetBackup™ セキュリティおよび暗号化ガイド』の「再発行トークンの作成」を参照してください。

4. トークンと指紋を使用して Secret を作成します。

5. セキュリティレベルに関係なく必須のため、トークンを指定します。

   Token-fingerprint-secret.yaml は次のようになります。

   ```
   apiVersion: v1
   kind: Secret
   metadata:
     name: secret-name
     namespace: kops-ns
   type: Opaque
   stringData:
     token: "Authorization token | Reissue token"
     fingerprint: "SHA256 Fingerprint"
   ```

   - Token-fingerprint-secret.yaml ファイルのテキストをコピーします。
   - テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
   - その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

6. Token-fingerprint-secret.yaml ファイルを作成するには、コマンド kubectl create -f Token-fingerprint-secret.yaml を実行します。

7. nbcaCreateOptions を使用して

   backupservercert オブジェクトを作成し、Secret 名を指定します。

   nbca-create-backupservercert.yaml は次のようになります。

   ```
   apiVersion: netbackup.veritas.com/v1
   kind: BackupServerCert
   metadata:
     name: backupserver-nbca-create
     namespace: kops-ns
   spec:
     clusterName: cluster.sample.com:port
   ```

```
backupServer: backupserver.sample.domain.com
certificateOperation: Create
certificateType: NBCA
nbcaAttributes:
  nbcaCreateOptions:
    secretName: nbcaSecretName with token and fingerprint
```

- nbca-create-backupservercert.yaml ファイルのテキストをコピーします。
- テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
- その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

8 nbca-create-backupservercert.yaml ファイルを作成するには、コマンド kubectl create -f nbca-create-backupservercert.yaml を実行します。

9 証明書が作成されたら、カスタムリソースの状態を確認します。カスタムリソースの状態が正常の場合は、スナップショットからのバックアップジョブを実行できます。

メモ: スナップショットからのバックアップ操作またはバックアップコピーからのリストア操作を開始する前に、BackupServerCert カスタムリソースの状態が正常であることを確認する必要があります。

メモ: ホスト ID ベースの証明書を更新するには、NetBackup のホスト ID 証明書で、24 時間後に更新が予定されているかどうかを確認します。証明書は、有効期限の 180 日 (6 カ月) 前に自動的に更新されます。

メモ: NetBackup プライマリサーバーのクロックと NetBackup Kubernetes Operator のクロックが同期しているかどうかを確認します。CheckClockSkew のエラーについて詳しくは、『NetBackup™ セキュリティおよび暗号化ガイド』の「証明書の有効期間に対するクロックスキューの意味」セクションを参照してください。

Kubernetes Operator からのプライマリサーバー証明書の削除

プライマリサーバーがバックアップおよびリストア操作の実行に使用されていない場合は、そのサーバーから証明書を削除できます。

Kubernetes Operator からプライマリサーバー証明書を削除するには

1. NetBackup Web UI にログオンし、削除する証明書のホスト ID を取得します。

   証明書のホスト ID を取得するには、『NetBackup™ セキュリティおよび暗号化ガイド』の「ホスト ID ベースの証明書の詳細の表示」セクションを参照してください。

2. 操作形式を削除に設定して backupservercert を作成します。

   `nbca-remove-backupservercert.yaml` ファイルは次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupserver-nbca-domain.com
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: backupserver.sample.domain.com
  certificateOperation: Remove
  certificateType: NBCA
  nbcaAttributes:
    nbcaRemoveOptions:
      hostID: nbcahostID
```

   - `nbca-remove-backupservercert.yaml` ファイルのテキストをコピーします。
   - テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
   - その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

3. `nbca-remove-backupservercert.yaml` ファイルを作成するには、コマンド `kubectl create -f nbca-remove-backupservercert.yaml` を実行します。

4. 証明書を無効にするには、『NetBackup™ セキュリティおよび暗号化ガイド』の「ホスト ID ベースの証明書の無効化」セクションを参照してください。

メモ: `nbca-remove-backupservercert.yaml` が適用されると、証明書は Kubernetes Operator のローカル証明書ストアから削除されます。ただし、まだ NetBackup データベースには存在し、有効なままです。したがって、証明書を無効にする必要があります。

プライマリサーバー証明書の更新

証明書が読み取り可能で、Kubernetes Operator に存在することを前提に、証明書を更新するシナリオを次に示します。

NetBackup Kubernetes Operator に存在する証明書が無効化されている場合は、更新操作を行って証明書を再発行できます。この問題を解決するには、サーバー証明書を更新するか、サーバー証明書を削除して新しい証明書を作成します。

メモ: 証明書の更新操作が失敗した場合は、最初に証明書を削除してから新しい証明書を作成する必要があります。

Kubernetes Operator でプライマリサーバー証明書を更新するには

1 更新操作を行って backupservercert オブジェクトを作成します。

`nbca-update-backupservercert.yaml` ファイルは次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupserver-nbca-update
  namespace:kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: backupserver.sample.domain.com
  certificateOperation: Update
  certificateType: NBCA
  nbcaAttributes:
    nbcaUpdateOptions:
    secretName: "Name of secret containing
token and fingerprint"
    force: true
```

- `nbca-update-backupservercert.yaml` ファイルのテキストをコピーします。
- テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
- その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

2 `nbca-update-backupservercert.yaml` ファイルを作成するには、コマンド `kubectl create -f nbca-update-backupservercert.yaml` を実行します。

3 backupservercert オブジェクトが作成されたら、カスタムリソースの状態を確認します。

ECA 証明書操作の実行

外部認証局 (ECA) の作成、更新、削除の各操作を実行する前に、ECA モードでバックアップサーバーを構成する必要があります。

ECA モードがオンかどうかを確認するには、コマンド `/usr/openv/netbackup/bin/nbcertcmd -getSecConfig -caUsage` を実行します。

出力は次のようになります。

```
NBCA: ON
ECA: ON
```

ECA モードでバックアップサーバーを構成するには、『NetBackup™ セキュリティおよび暗号化ガイド』の「NetBackup での外部 CA のサポートについて」を参照してください。

ECA 証明書の指定は次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupservercert-sample-eca
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: primaryserver.sample.domain.com
  certificateOperation: Create | Update | Remove
  certificateType: ECA
  ecaAttributes:
    ecaCreateOptions:
      ecaSecretName: "Secret name consists of cert, key, passphrase,
 cacert"
      copyCertsFromSecret: true | false
      isKeyEncrypted: true | false
    ecaUpdateOptions:
      ecaCrlCheck: DISABLE | LEAF | CHAIN
      ecaCrlRefreshHours: range[0,4380]
```

表 3-2 ECA 証明書操作

操作形式	オプションとコメント
作成 (Create)	■ secretName: 証明書、キー、パスフレーズ、cacert を含む Secret の名前。 ■ copyCertsFromSecret: 指定可能な値は true および false です。このオプションは、外部 CA がすべてのプライマリサーバーで共通しているため、追加されます。すべてのプライマリサーバーの Kubernetes Operator に同じ証明書を登録できます。したがって、証明書とキーを毎回コピーする必要はありません。証明書とキーのコピーは、このオプションを使用して制御できます。 証明書とキーの問題が原因で ECAHealthCheck が失敗した場合は、証明書を再度コピーする必要があります。 ■ isKeyEncrypted: 秘密鍵が暗号化されている場合はこのフィールドを true に設定し、それ以外の場合は false に設定します。
削除 (Remove)	なし
更新 (Update)	■ ecaCrlCheck: 外部証明書の失効の確認レベルを指定できます。 指定可能な値は LEAF、CHAIN、DISABLE です。 ■ ecaCrlRefreshHours: 証明書失効リストをダウンロードする間隔 (時間単位) を指定します。 指定可能な値の範囲は 0 から 4380 までです。

ECA が署名した証明書の作成

NetBackup は、Kubernetes Operator に登録された ECA の複数のプライマリサーバーでの使用をサポートしています。外部 CA がプライマリサーバーで共通している場合、通信中に証明書失効リストを動的にフェッチするには、証明書失効リスト配布ポイントを使用する必要があります。

ECA が署名した証明書を作成するには

1. 証明書失効リスト配布ポイントを使用して、証明書失効リストをフェッチします。
2. ECA が署名した証明書チェーン、秘密鍵、(必要な場合は) パスフレーズをホームディレクトリに準備しておきます。
3. 手順 2 で説明した各ファイルでサポートされているさまざまな形式 (DER、PEM など) を識別します。詳しくは、『NetBackup™ セキュリティおよび暗号化ガイド』の「外部 CA が署名した証明書の構成オプション」セクションを参照してください。
4. 手順 3 で説明したファイルを使用して Secret を作成します。
   - 秘密鍵が暗号化されていない場合に Secret を作成するには、コマンド `kubectl create secret generic <Name of secret>` を実行します。

```
--from-file=cert_chain=<File path to ECA signed certificate
chain> --from-file=key=<File path to private key>
--from-file=cacert=<File path to External CA certificate> -n
<Namespace where kops is deployed>
```

- 秘密鍵が暗号化されている場合に Secret を作成するには、コマンド `kubectl create secret generic <Name of secret>` を実行します。

```
--from-file=cert_chain=<File path to ECA signed certificate
chain> --from-file=key=<File path to private key>
--from-file=cacert=<File path to External CA certificate>
--from-file=passphrase=<File path to passphrase
of encrypted private key> -n <Namespace where kops is deployed>
```

ディレクトリ構造は次のようになります。

```
├── cert_chain.pem
├── private
|  |___key.pem
|  |___passphrase.txt
|___trusted
     |__cacerts.pem
```

cert_chain.pem は ECA が署名した証明書チェーンです。

private/key.pem は秘密鍵です。

private/passphrase.txt は秘密鍵のパスフレーズです。

trusted/cacerts.pem は外部 CA 証明書です。

- 秘密鍵が暗号化されていない場合に名前 eca-secret の Secret を作成するには、次のコマンドを実行します。

```
kubectl create secret generic
eca-secret--from-file=cert_chain=cert_chain.pem
--from-file=key=private/key.pem
--from-file=cacert=trusted/cacerts.pem -n kops-ns
```

- 秘密鍵が暗号化されている場合に名前 eca-secret の Secret を作成するには、次のコマンドを実行します。

```
kubectl create secret generic eca-secret
--from-file=cert_chain=cert_chain.pem
--from-file=key=private/key.pem
--from-file=cacert=trusted/cacerts.pem
--from- file=passphrase=private/passphrase.txt
-n kops-ns
```

5 Secret が作成されたら、backupservercert オブジェクトのカスタムリソースを作成します。

eca-create-backupservercert.yaml ファイルは次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupservercert-eca-create
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: backupserver.sample.domain.com
  certificateOperation: Create
  certificateType: ECA
  ecaAttributes:
    ecaCreateOptions:
      ecaSecretName: eca-secret
      copyCertsFromSecret: true
      isKeyEncrypted: false
```

- eca-create-backupservercert.yaml ファイルのテキストをコピーします。
- テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
- その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

6 証明書とキーを Kubernetes Operator にコピーするには、次の操作のいずれかを実行します。

- copyCertsFromSecret を true に設定します。
- Kubernetes Operator に存在する証明書とキーのコピーを回避するには、copyCertsFromSecret を false に設定します。

メモ: ECA はすべてのプライマリサーバーで共通しているため、Kubernetes Operator では、必要に応じてすべてのプライマリサーバーに登録できる 1 組の証明書とキーが必要です。以前にコピーした証明書とキーに問題がないかぎり、証明書とキーを毎回コピーする必要はありません。

メモ: 証明書やキーに関連した理由 (破損、期限切れ、または ECA の変更) が原因で ecaHealthCheck が失敗した場合は、エラーの原因を特定し、フラグを使用して有効な証明書のコピーを実行します。

---

7 秘密鍵が暗号化されている場合は isKeyEncrypted フラグを true に設定し、暗号化されていない場合は false に設定します。秘密鍵が暗号化されている場合は、パスフレーズが Secret で指定されていることを確認します。

8 手順 5 で backupservercert yaml を作成した Secret 名を使用して ecaSecretName を設定します。

9 eca-create-backupservercert.yaml ファイルを作成するには、コマンド kubectl create -f eca-create-backupservercert.yaml を実行します。

10 backupservercert カスタムリソースが作成されたら、カスタムリソースの状態を確認します。

11 NetBackup Web UI で外部証明書の詳細を表示するには、『NetBackup™ Web UI 管理者ガイド』の「ドメイン内の NetBackup ホストの外部証明書情報の表示」セクションを参照してください。

ECA が署名した証明書の削除

ECA が署名した証明書をプライマリサーバーから削除できます。

ECA が署名した証明書を削除するには

1 操作を削除、証明書の種類を ECA に設定して backupservercert を作成します。

eca-remove-backupservercert.yaml ファイルは次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupservercert-eca-remove
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: backupserver.sample.domain.com
  certificateOperation: Remove
  certificateType: ECA
```

- eca-remove-backupservercert.yaml ファイルのテキストをコピーします。
- テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
- その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

2 eca-remove-backupservercert.yaml ファイルを作成するには、コマンド kubectl create -f eca-remove-backupservercert.yaml を実行します。

3 オブジェクトが作成されたら、カスタムリソースの状態を確認する必要があります。失敗した場合は、必要な措置を講じることができます。

上記の手順を実行すると、指定したプライマリサーバーに関する外部証明書の詳細がローカル証明書ストアから削除されます。証明書は、システムからも NetBackup データベースからも削除されません。

ECA を無効にする場合は、『NetBackup™ セキュリティおよび暗号化ガイド』の「NetBackup ドメインでの外部 CA の無効化」セクションを参照してください。

バックアップサーバーの Kubernetes Operator に ECA を登録したものの、その後 NBCA のみをサポートするバックアップサーバーを再インストールした場合は、Kubernetes Operator から ECA の登録を削除する必要があります。これは、*nbcertcmd* の実行時に、バックアップサーバーの CA のサポートとの通信が比較されることがあり、不一致の場合にエラーが発生するためです。

ECA が署名した証明書の更新

ECA で設定可能な特定のオプションがあります。更新操作でこれらのオプションを設定できます。

ECA が署名した証明書を更新するには

1 操作形式を更新に設定して `backupservercert` オブジェクトを作成します。

`eca-update-backupservercert.yaml` ファイルは次のようになります。

```
apiVersion: netbackup.veritas.com/v1
kind: BackupServerCert
metadata:
  name: backupservercert-eca-update
  namespace: kops-ns
spec:
  clusterName: cluster.sample.com:port
  backupServer: backupserver.sample.domain.com
  certificateOperation: Update
  certificateType: ECA
  ecaAttributes:
    ecaUpdateOptions:
      ecaCrlCheck: DISABLE | LEAF | CHAIN
      ecaCrlRefreshHours: [0,4380]
```

- `eca-update-backupservercert.yaml` ファイルのテキストをコピーします。
- テキストエディタを開き、yaml ファイルのテキストを貼り付けます。
- その後、そのテキストに yaml ファイル拡張子を付けて、Kubernetes クラスタにアクセスできるホームディレクトリに保存します。

2 `eca-update-backupservercert.yaml` ファイルを作成するには、コマンド `kubectl create -f eca-update-backupservercert.yaml` を実行します。

3 ECA_CRL_CHECK オプションを使用すると、ホストの外部証明書の失効の確認レベルを指定できます。外部証明書の失効の確認を無効にすることもできます。確認に基づいて、ホストとの通信時に、証明書失効リスト (CRL) に対して証明書の失効状態が検証されます。詳しくは、『NetBackup™ セキュリティおよび暗号化ガイド』の「NetBackup サーバーとクライアントの ECA_CRL_CHECK」セクションを参照してください。

4 ECA_CRL_REFRESH_HOURS オプションは、ピアホスト証明書の証明書失効リスト配布ポイント (CDP) で指定した URL から CRL をダウンロードする間隔 (時間単位) を指定します。詳しくは、『NetBackup™ セキュリティおよび暗号化ガイド』の「NetBackup サーバーとクライアントの ECA_CRL_REFRESH_HOURS」セクションを参照してください。

証明書の種類の識別

NetBackup は、Kubernetes Operator に登録されている証明書の種類を識別するのに役立ちます。

証明書の種類を識別するには

1 Kubernetes Operator ポッドを一覧表示するには、コマンド `kubectl get pods -n <namespace of Kubernetes operator>` を実行します。

2 管理者権限を使用して Kubernetes Operator にログオンし、次のコマンドを実行します。

```
kubectl exec pod/nbu-controller-manager-7c99fb8474-hzrsl -n
<namespace of Kubernetes operator> -c netbackupkops -it -- bash
```

3 Kubernetes に NBCA 証明書を使用しているバックアップサーバーを一覧表示するには、次のコマンドを実行します。

```
/nbcertcmdtool/nbcertcmdtool -atLibPath/nbcertcmdtool/

-standalone -installDir "/usr/openv" -listCertDetails -NBCA
```

出力は次のようになります。

```
Master Server : masterserver.sample.domain.com
Host ID : b06738f0-a8c1-47bf-8d95-3b9a41b7bb0a
Issued By : /CN=broker/OU=NBCANBKOps
Serial Number : 0x508cdf4500000008
Expiry Date : Dec 22 05:46:32 2022 GMT
SHA-1 Fingerprint : 4D:7A:D9:B9:61:4E:93:29:B8:93:0B:E0:
07:0A:28:16:46:F6:39:C6
SHA-256 Fingerprint : C2:FA:AC:B5:21:6B:63:49:30:AC:4D:5E:
61:09:9A:8C:C6:40:4A:44:B6:39:7E:2B:B3:36:DE:D8:F5:D1:3D:EF
Key Strength : 2048
Subject Key Identifier : AC:C4:EF:40:7D:8D:45:B4:F1:89:DA:FB:
E7:FD:0F:FD:EC:61:12:C6
Authority Key Identifier : 01:08:CA:40:15:81:75:7B:37:9F:51:78:

B2:6A:89:A1:44:2D:82:2B
```

4 Kubernetes に ECA 証明書を使用しているバックアップサーバーを一覧表示するには、次のコマンドを実行します。

```
/nbcertcmdtool/nbcertcmdtool -atLibPath/nbcertcmdtool/

-standalone -installDir"/usr/openv" -listCertDetails -ECA
```

出力は次のようになります。

```
Subject Name : CN=ECA-KOPS,O=Veritas,OU=ECANBKOps
 Issued By : CN=ICA-2,O=Veritas,OU=ECANBKOps
 Serial Number : 0x56cf16040258d3654339b7f39817de89240d58
 Expiry Date : Dec 16 05:48:16 2022 GMT
 SHA-1 Fingerprint : 70:DE:46:72:57:56:4E:47:DB:82:8B:8D:A3:
 4B:BB:F9:8D:2C:B7:8E
 SHA-256 Fingerprint : E0:69:5F:79:A6:60:DB:7B:69:76:D3:A8:
 E6:E1:F2:0D:8C:6C:E6:4E:C4:5D:A4:77:17:5A:C2:42:89:74:15:7D
 Key Strength : 2048
 Subject Key Identifier : F0:E7:1F:8C:50:FD:4D:25:40:69:77:6C:
 2A:35:72:B6:1D:8E:E5:17
 Authority Key Identifier : D7:53:57:C7:A6:72:E3:CB:73:BD:48:51:

 2F:CB:98:A3:0B:8B:BA:5C
 Master Server : masterserver.sample.domain.com
 Host ID : b85ba9bf-02a8-439e-b787-ed52589c37d1
```

4

Kubernetes 資産の管理

この章では以下の項目について説明しています。

Kubernetes クラスタの追加

NetBackup に Kubernetes クラスタを追加する前に、クラスタに Kubernetes Operator をインストールして構成する必要があります。そうしないと、クラスタの検証が失敗し、さらにクラスタの追加操作が失敗します。

Kubernetes Operator を構成すると、NetBackup に Kubernetes クラスタを追加し、クラスタ内のすべての資産を自動的に検出できます。

クラスタを追加するには

1. 左側の[作業負荷 (Workloads)]で、[Kubernetes]をクリックします。
2. [Kubernetes クラスタ (Kubernetes clusters)]タブをクリックし、[追加 (Add)]をクリックします。
3. [Kubernetes クラスタの追加 (Add Kubernetes cluster)]ページで、次を入力します。
   - [クラスタ名 (Cluster name)]: クラスタの名前を入力します。この名前は DNS の解決可能な値または IP アドレスである必要があります。例: cluster.sample.domain.com。
   - [ポート (Port)]: Kubernetes API サーバーのポート番号を入力します。
   - [コントローラの名前空間 (Controller namespace)]: Kubernetes クラスタ内で NetBackup Kubernetes Operator が配備されている名前空間を入力します。例: kops-ns。

4 [次へ (Next)]をクリックします。[クレデンシャルの管理 (Manage credentials)]ページで、クラスタにクレデンシャルを追加できます。

- 既存のクレデンシャルを使用するには、[既存のクレデンシャルから選択してください (Select from existing credentials)]を選択し、[次へ (Next)]をクリックします次のページで、必要なクレデンシャルを選択し、[次へ (Next)]をクリックします。
- 新しいクレデンシャルを作成するには、[クレデンシャルの追加 (Add credential)]をクリックし、[次へ (Next)]をクリックします。[クレデンシャルの管理 (Manage credentials)]ページで、次を入力します。
  - [クレデンシャル名 (Credential name)]: クレデンシャルの名前を入力します。
  - [タグ (Tag)]: クレデンシャルに関連付けるタグを入力します。
  - [説明 (Description)]: クレデンシャルの説明を入力します。
  - NetBackup に Kubernetes クラスタを追加するには、認証局 (CA) 証明書とトークンが必要です。Kubernetes クラスタの認証と認可には、CA 証明書とバックアップサービスアカウントのトークンが必要です。CA 証明書とトークンを取得するには、Kubernetes クラスタでコマンド `kubectl get secret <[namespace-name]-backup-server-secret> -n <namespace name> -o yaml.` を実行します。
    - [トークン (Token)]: 認証トークンの値を Base64 エンコード形式で入力します。
    - [CA 証明書 (CA certificate)]: CA 証明書ファイルの内容を指定します。

5 [次へ (Next)]をクリックします。

クレデンシャルが検証され、検証に成功すると、クラスタが追加されます。クラスタが追加されると、自動検出が実行され、クラスタ内の利用可能な資産が検出されます。

メモ: NetBackup Kubernetes バージョン 10.1 で、クラスタの編集操作が失敗し、エラーメッセージが表示されます。この問題を解決するには、最初にクラスタを削除し、クラスタを再び追加することをお勧めします。

設定を行う

Kubernetes の設定では、Kubernetes の配備のさまざまな側面を構成できます。

Kuberentes リソース形式のリソース制限の変更

リソース制限の設定について

この設定によって、Kubernetesクラスタで同時に実行できるバックアップの数を制御できます。Kubernetes では、スナップショットジョブを実行する場合のデフォルト値は 1、スナップショットからのバックアップジョブを実行する場合のデフォルト値は 4 とそれぞれ異なります。

例:

20 の資産を保護し、制限を 5 に設定している場合に、スナップショットのみのバックアップジョブを実行すると、5 つの資産のみ同時にバックアップを実行でき、残りの 15 の資産はキューに入ります。最初の 5 つの資産のうち 1 つのバックアップが完了すると、キューの資産にバックアップの順番が回ります。

スナップショットジョブを実行する場合、リソース制限のデフォルト値は 1 です。これは、クラスタごとに 1 つのバックアップジョブのみが進行中になり、残りの資産はキューに投入された状態になることを示します。

システムとネットワークリソースの使用を最適化するため、この設定をお勧めします。この設定は、選択しているプライマリサーバーのすべての Kubernetes バックアップに適用されます。

リソース制限を設定するには

1. 左側で[作業負荷 (Workloads)]、[Kubernetes]の順にクリックします。
2. 右上で[Kubernetes 設定 (Kubernetes settings)]、[リソース制限 (Resource limits)]の順にクリックします。
3. リソース制限を設定するには、次のいずれかを実行します。
   - [Kubernetes クラスタあたりのバックアップジョブ (Backup jobs per Kubernetes cluster)]の横にある[編集 (Edit)]をクリックします。デフォルトでは、制限は 1 です。
     デフォルトでは、クラスタあたりのバックアップジョブのリソース制限は 1 です。
   - [Kubernetes クラスタあたりのスナップショットからのバックアップジョブ (Backup from Snapshot Jobs per Kubernetes Cluster)]の横にある[編集 (Edit)]をクリックします。
     デフォルトでは、クラスタあたりのスナップショットからのバックアップジョブのリソース制限は 4 です。
4. [Kubernetes クラスタの編集 (Edit Kubernetes cluster)]ダイアログで、次の操作を行います。
   - [グローバル (Global)]フィールドに値を入力し、すべてのクラスタのグローバル制限を設定します。この制限は、クラスタで同時に実行されるバックアップジョブとスナップショットからのバックアップジョブの数を示します。

- そのクラスタのグローバル制限を上書きする個別の制限をクラスタに追加できます。クラスタに個々の制限を設定するには、[追加 (Add)]をクリックします。
- リストから利用可能なクラスタを選択し、選択したクラスタの制限値を入力できます。配備されている利用可能な各クラスタに制限を追加できます。
- [保存 (Save)]をクリックして、変更を保存します。

メモ: NetBackup 10.0 リリースでは、datamover ポッドは Kubernetes のリソース制限の設定を超過します。

p.106 の「datamover ポッドが Kubernetes のリソース制限を超過 」を参照してください。

自動検出の間隔の構成

自動検出により、クラスタ内で NetBackup によって保護される資産数が記録されます。この設定を使用すると、NetBackup が自動検出を実行して、クラスタ内の新しい資産を特定する間隔を設定できます。クラスタから排除または削除された資産の数を収集します。

指定できる値は、5 分から 1 年の間です。デフォルト値は 30 分です。

自動検出の間隔を設定するには

1. 左側で[作業負荷 (Workloads)]、[Kubernetes]の順にクリックします。
2. 右上で[Kubernetes 設定 (Kubernetes settings)]、[自動検出 (Autodiscovery)]の順にクリックします。
3. [間隔 (Frequency)]の近くにある[編集 (Edit)]をクリックします。
4. NetBackup が自動検出を実行した後の時間数を入力します。[保存 (Save)]をクリックします。

完全検出と増分検出の実行

Kubernetes クラスタが追加されると、自動検出サイクルがトリガされ、Kubernetes クラスタで利用可能なすべての資産が検出されます。その日最初の自動検出は完全検出で、以降の自動検出は増分検出です。

検出を実行するには

1. 左側で[作業負荷 (Workloads)]、[Kubernetes]の順にクリックします。
2. Kubernetes クラスタのリストで、クラスタ名を見つけます。次に、[処理 (Actions)]、[今すぐ検出 (Discover now)]の順にクリックします。

この場合、増分検出では前回の検出実行以降にクラスタで変更された NetBackup 資産のみをフェッチします。したがって、最初の検出は完全検出で、それ以降のすべての検出は増分検出になります。

権限の構成

管理権限を使用して、ユーザーロールに異なるアクセス権を割り当てることができます。詳しくは、『NetBackup Web UI 管理者ガイド』の「役割ベースのアクセス制御の管理」の章を参照してください。

資産への保護の追加

[名前空間 (Namespaces)]タブ ([作業負荷 (Workloads)]、[Kubernetes]) を使用して、Kubernetes クラスタ内の資産の監視、保護状態の確認、保護されていない資産への保護の追加を簡単に行えます。また、[今すぐバックアップ (Backup now)]機能を使用して資産のクイックバックアップを作成できます。この機能は、スケジュール設定されたバックアップに影響を与えることなく、選択した資産のワンタイムバックアップを作成します。

[名前空間 (Namespaces)]タブに、NetBackup によって保護できる検出済みおよびインポート済みの Kubernetes 資産がすべて表示されます。このタブには、次の情報が表示されます。

- [名前空間 (Namespaces)]: 資産の表示名。
- [クラスタ (Cluster)]: 資産が属するクラスタ。
- [保護計画名 (Protected by)]: 資産に適用された保護計画の名前。
- [最後に成功したバックアップ (Last successful backup)]: 資産のバックアップが最後に成功した日時。

[名前空間 (Namespaces)]タブで次の操作を実行できます。

保護されていない資産に保護を追加するには

1. 左側で[作業負荷 (Workloads)]、[Kubernetes]の順にクリックします。
2. 資産の行でオプションを選択します。右上の[保護の追加 (Add protection)]をクリックします。または、資産の行の[処理 (Actions)]メニューをクリックして、[保護の追加 (Add protection)]をクリックします。
3. リストから保護計画を選択し、[次へ (Next)]をクリックします。次のページで、[保護 (Protect)]をクリックします。

資産をすばやくバックアップするには

1. 資産の行でオプションを選択し、右上の[今すぐバックアップ (Backup now)]をクリックします。または、資産の行の[処理 (Actions)]メニューをクリックして、[今すぐバックアップ (Backup now)]をクリックします。
2. 次のページで、

- すでに保護されている資産をバックアップする場合は、資産がすでにサブスクライブされている計画のリストから保護計画を選択し、[バックアップの開始 (Start backup)]をクリックします。
- 保護されていない資産をバックアップする場合は、その資産で利用可能な計画から保護計画を選択し、[バックアップの開始 (Start backup)]をクリックします。

5

Kubernetes インテリジェントグループの管理

この章では以下の項目について説明しています。

インテリジェントグループについて

問い合わせと呼ばれるフィルタのセットに基づいて、インテリジェント資産グループを定義して、資産のダイナミックグループを作成および保護できます。NetBackup は、問い合わせに基づいて Kubernetes 名前空間を選択し、それらをグループに追加します。インテリジェントグループでは、資産の環境内の変更が自動的に反映されるため、環境内で資産を追加または削除しても、グループ内の資産のリストを手動で修正する必要がないことに注意してください。

インテリジェントグループに保護計画を適用すると、問い合わせ条件を満たすすべての資産が自動的に保護されます。

メモ: インテリジェントグループの作成、更新、削除は、管理が必要な資産に対する必要な RBAC 権限が役割に付与されている場合にのみ実行できます。NetBackup のセキュリティ管理者は、資産タイプ (クラスタ、名前空間、VMGroup) に対するアクセス権を付与できます。『NetBackup Web UI 管理者ガイド』を参照してください。

インテリジェントグループの作成

インテリジェントグループを作成するには

1. 左側の[作業負荷 (Workloads)]で、[Kubernetes]をクリックします。
2. [インテリジェントグループ (Intelligent groups)]タブ、[+ 追加 (+ Add)]の順にクリックします。
3. グループの名前と説明を入力します。
4. [クラスタ (Cluster)]セクションで、[クラスタの追加 (Add clusters)]をクリックします。
5. [クラスタの追加 (Add clusters)]ウィンドウで、一覧から 1 つ以上のクラスタを選択し、[選択 (Select)]をクリックします。選択したクラスタがインテリジェントグループに追加されます。

   メモ: インテリジェントグループは、複数のクラスタをまたいで作成できます。グループにクラスタを追加するために必要な権限を持っていることを確認します。グループを表示して管理するには、選択したクラスタとグループに対する表示と管理の権限がグループ管理者に付与されている必要があります。

6. [資産の選択 (Select assets)]セクションで、次のいずれかを実行します。
   - [すべての資産を含める (Include all assets)]を選択します。
     このオプションでは、デフォルトの問い合わせを使用して、保護計画の実行時にすべての資産をバックアップ対象として選択します。
   - 特定の条件を満たす資産のみを選択するには、独自の問い合わせを作成するために[条件の追加 (Add condition)]をクリックします。
   - 資産のラベル条件を追加するには、[ラベルの条件の追加 (Add label condition)]をクリックして追加します。

7 条件を追加するには、ドロップダウンを使用してキーワードと演算子を選択し、値を入力します。

問い合わせの効果を変更するには、[+ 条件 (+ Condition)]をクリックし、[AND]または[OR]をクリックして、キーワード、演算子、条件の値を選択します。

メモ: ラベル条件を追加するには、[ラベル条件の追加 (Add label condition)]をクリックしてラベルキーと値を入力します。

メモ: 条件にラベル値を含めず、ラベルキーのみを含めることもできます。値は、ラベル条件に追加するオプションのパラメータであるためです。

メモ: サブクエリーを追加するには、[サブクエリーの追加 (Add sub-query)]をクリックします。複数のレベルのサブクエリーを追加できます。

---

8 問い合わせをテストするには、[プレビュー (Preview)]をクリックします。

問い合わせベースの選択処理は動的です。Kubernetes クラスタの変更は、保護計画の実行時に問い合わせが選択する資産に影響する可能性があります。その結果、保護計画が後で実行された時に問い合わせが選択する資産が、プレビューに現在表示されているものと同一でなくなる可能性があります。

メモ: [インテリジェントグループ (Intelligent groups)]で問い合わせを使用する場合、問い合わせ条件に英語以外の文字が含まれていると、NetBackup Web UI に、問い合わせに一致する正確な資産のリストが表示されないことがあります。

任意の属性に `not equals` フィルタ条件を使用すると、属性に値が存在しない (null) 資産を含む資産が戻されます。

メモ: [プレビュー (Preview)]をクリックするかグループを保存した場合、グループの資産を選択するときに、問い合わせオプションでは大文字と小文字が区別されます。

---

9 グループを保護計画に追加せずに保存するには、[追加 (Add)]をクリックします。

10 グループを保護計画に追加して保存するには、[追加と保護 (Add and protect)]をクリックします。

11 保護計画にグループをサブスクライブするには、[保護の追加 (Add protection)]をクリックします。

グループを選択して保護計画を適用し、[保護する (Protect)]をクリックします。

選択した資産グループが保護計画に正常にサブスクライブされます。

資産にラベル条件を追加する際の制限事項

条件とラベルの組み合わせがある場合は、最初に名前空間条件を定義してから、ラベル条件を定義する必要があります。

メモ: 条件については、名前空間の値のみが許可されます。

インテリジェントグループの削除

インテリジェントグループを削除するには

1. 左側の[作業負荷 (Workloads)]で、[Kubernetes]をクリックします。
2. [インテリジェントグループ (Intelligent groups)]タブでグループを見つけます。
3. グループが保護されていない場合は、グループを選択して[削除 (Delete)]をクリックします。
4. グループが保護されている場合は、グループを選択し、[保護の削除 (Remove protection)]をクリックしてすべての保護計画を削除します。
5. 次に、[インテリジェントグループ (Intelligent groups)]タブでこのグループを選択し、[削除 (Delete)]をクリックします。

インテリジェントグループの編集

インテリジェントグループの名前と説明の詳細を編集できます。スケジュールバックアップの時間帯や他のオプションなど、保護計画の特定の設定を編集できます。

インテリジェントグループを編集するには

1. 左側の[作業負荷 (Workloads)]で、[Kubernetes]をクリックします。
2. [インテリジェントグループ (Intelligent groups)]タブで、保護を編集するグループをクリックします。
3. 次のいずれかを実行します。
   - [名前と説明を編集する (Edit name and description)]をクリックして、選択したグループの名前と説明を編集した後、[保存 (Save)]をクリックします。
   - [資産 (Assets)]タブで、[編集 (Edit)]をクリックしてクラスタを追加または削除します。選択した資産の問い合わせ条件を更新し、[保存 (Save)]をクリックします。
     グループのクラスタリストを編集したり、グループのクラスタを追加または削除したりできます。選択した資産グループの問い合わせ条件を変更することもできます。
   - [権限 (permissions)]タブで、[追加 (Add)]をクリックして利用可能な役割の権限を更新し、[保存 (Save)]をクリックします。

6

Kubernetes 資産の保護

この章では以下の項目について説明しています。

インテリジェントグループの保護

Kubernetes 作業負荷用に Kubernetes 固有の保護計画を作成できます。その後、保護計画にインテリジェントグループをサブスクライブできます。

次の手順を使用して、インテリジェントグループを保護計画にサブスクライブします。

メモ: 自分に割り当てられている RBAC の役割によって、管理するインテリジェントグループと、使用する保護計画にアクセスできるようにする必要があります。

インテリジェントグループを保護するには

1. 左側で[Kubernetes]をクリックします。
2. [インテリジェントグループ (Intelligent groups)]タブで、グループにチェックマークを付けて[保護の追加 (Add protection)]をクリックします。
3. 保護計画を選択し、[次へ (Next)]をクリックします。
4. グループを選択し、[保護する (Protect)]をクリックして保護計画にサブスクライブします。

即時保護のための[今すぐバックアップ (Backup now)]オプション

スケジュール設定された保護計画とは別に、[今すぐバックアップ (Backup now)]オプションを使用してグループをすぐにバックアップし、計画外の状況に対して保護することもできます。

インテリジェントグループからの保護の削除

インテリジェントグループのサブスクライブを、保護計画から解除できます。インテリジェントグループのサブスクライブが保護計画から解除されると、それ以降バックアップは実行されません。

インテリジェントグループから保護を削除するには

1. 左側の[作業負荷 (Workloads)]で、[Kubernetes]をクリックします。
2. [インテリジェントグループ (Intelligent groups)]タブで、保護を削除するグループをクリックします。
3. [保護の削除 (Remove protection)]、[はい (Yes)]の順にクリックします。

バックアップスケジュールの構成

Kubernetes 作業負荷の保護計画を作成する際、[バックアップスケジュールの追加 (Add backup schedule)]ダイアログの[属性 (Attributes)]タブでバックアップスケジュールを追加できます。

保護計画の作成方法について詳しくは、『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションを参照してください。

Kubernetes バックアップジョブのバックアップスケジュールを追加するには

1. 左側で[保護 (Protection)]、[保護計画 (Protection plans)]、[追加 (Add)]の順にクリックします。
2. [基本プロパティ (Basic properties)]で、[名前 (Name)]と[説明 (Description)]を入力し、[作業負荷 (Workload)]ドロップダウンリストから[Kubernetes]を選択します。
3. [次へ (Next)]をクリックします。[スケジュール (Schedules)]で、[スケジュールの追加 (Add schedule)]をクリックします。

   [バックアップスケジュールの追加 (Add backup schedule)]タブで、バックアップとスナップショットを保持するためのオプションを構成できます。
4. [反復 (Recurrence)]ドロップダウンから、バックアップの頻度を指定します。
5. [スナップショットとバックアップコピー (Snapshot and backup copy)]オプションで、以下のいずれかを行います。

   - 保護計画のスナップショットからのバックアップを構成するには、[スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択します。[バックアップの保持期間 (Keep backup for)]ドロップダウンを使用して、スナップショットからのバックアップの保持期間を指定します。

     メモ: Kubernetes 作業負荷でサポートされるのは、完全バックアップのスケジュールのみです。バックアップ期間は、時間、日、週、月、年単位で設定できます。

     デフォルトでは、4 週間がバックアップの保持期間です。

     メモ: バックアップコピーのレプリケーションと複製のオプションを有効にするには、[スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択する必要があります。

   - [スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択しなかった場合、デフォルトでは、バックアップジョブを実行するために[スナップショットのみのストレージ (Snapshot only storage)]バックアップが構成されます。
   - バックアップのレプリカコピーを作成するには、[スナップショットからバックアップのレプリカコピー (自動イメージレプリケーション) を作成 (Create a replica copy (Auto Image Replication) of the backup from snapshot)]オプションを選択します。
   - バックアップの複製コピーを作成するには、[スナップショットからバックアップの複製コピーを作成 (Create a duplicate copy of the backup from snapshot)]オプションを選択します。

6 『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションにある説明に従って、[開始時間帯 (Start window)]タブでスケジュールの作成を続行します。

7 『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションにある説明に従って、スナップショットからのバックアップ用に[ストレージオプション (Storage options)]の設定を続行します。

バックアップオプションの構成

保護計画のバックアップオプションを構成できます。

保護計画の作成方法について詳しくは、『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションを参照してください。

保護計画の構成時にバックアップオプションを構成するには

1 [バックアップオプション (Backup options)]ページの[リソースの種類の選択 (Resource kind selection)]セクションで、次を実行します。

- デフォルトでは[すべてのリソースの種類をバックアップに含めます。(Include all resource kinds in the backup)]オプションが選択されており、バックアップジョブのすべてのリソースの種類が含まれています。
- [次のリソースの種類をバックアップから除外します。(Exclude the following resource kinds from the backup)]オプションを選択すると、リソースの種類がバックアップジョブから除外されます。[選択 (Select)]をクリックして、静的リストからリソースの種類を選択します。選択したリソースの種類がテキストフィールドに表示されるか、カスタムリソース定義 (CRD) を正しい形式 (type.group) で手動で入力できます。選択したリソースの種類を除外リストから削除できます。
  カスタムリソースの種類の定義が静的リストにない場合は、カスタムリソース定義 (CRD) を手動で入力できます。たとえば、demo.nbu.com のように入力します。

メモ: リソースの種類の除外リストは、リソースをマッピングするという点で、バックアップ用に選択したラベルより優先されます。

---

2 [ラベルの選択 (Label selection)]セクションで[追加 (Add)]をクリックして、バックアップ用に関連付けられたリソースをマッピングするラベルを追加し、ラベルの接頭辞とキーを入力し、演算子を選択します。含まれているラベルに関連付けられているすべてのリソースが、バックアップジョブに対してマッピングされます。

ラベルに追加できる 4 つの演算子を次に示します。

- 値と等しいラベルキーを入力します。
- すでに存在するラベルキーを、値なしで入力します。
- 一連の値に含まれているラベルキーを入力します。

- 一連の値に含まれていないラベルキーを入力します。

演算子の一連の値に含まれている、または含まれていない複数の値をカンマ区切りで追加できます。

メモ: 条件が正常に適用されるようにするには、選択したラベルがバックアップ時に存在する必要があります。

メモ: ラベルの選択では、複数のラベル条件間で矛盾しないリソースの種類の選択のみを除外する必要があります。

[確認 (Review)]ページには、リソースの種類の除外リストと、リストに含めるために選択したラベル、および選択したストレージユニットが表示されます。

メモ: Kubernetes 作業負荷用に作成された保護計画は編集または削除できます。

Kubernetes 作業負荷用に作成された保護計画はカスタマイズできません。

バックアップの構成

NetBackup では、スナップショットのみとスナップショットからのバックアップという 2 種類のバックアップジョブを Kubernetes 作業負荷で実行できます。Kubernetes Operator のバックアップジョブを構成する手順に従ってください。

Kubernetes 作業負荷でバックアップを実行するには

1. 左側で[保護 (Protection)]、[保護計画 (Protection plans)]、[追加 (Add)]の順にクリックします。
2. [基本プロパティ (Basic properties)]で、[名前 (Name)]と[説明 (Description)]を入力し、[作業負荷 (Workload)]ドロップダウンリストから[Kubernetes]を選択します。
3. [次へ (Next)]をクリックします。[スケジュール (Schedules)]で、[スケジュールの追加 (Add schedule)]をクリックします。

   [バックアップスケジュールの追加 (Add backup schedule)]タブで、バックアップとスナップショットを保持するためのオプションを構成できます。
4. [反復 (Recurrence)]ドロップダウンから、バックアップの頻度を指定します。
5. [スナップショットとバックアップコピー (Snapshot and backup copy)]オプションで、以下のいずれかを行います。

   - 保護計画のスナップショットからのバックアップを構成するには、[スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択し

ます。[バックアップの保持期間 (Keep backup for)]ドロップダウンを使用して、スナップショットからのバックアップの保持期間を指定します。

メモ: Kubernetes 作業負荷でサポートされるのは、完全バックアップのスケジュールのみです。バックアップ期間は、時間、日、週、月、年単位で設定できます。デフォルトでは、4 週間がバックアップの保持期間です。

メモ: バックアップコピーのレプリケーションと複製のオプションを有効にするには、[スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択する必要があります。

- [スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択しなかった場合、デフォルトでは、バックアップジョブを実行するために[スナップショットのみのストレージ (Snapshot only storage)]バックアップが構成されます。
- バックアップのレプリカコピーを作成するには、[スナップショットからバックアップのレプリカコピー (自動イメージレプリケーション) を作成 (Create a replica copy (Auto Image Replication) of the backup from snapshot)]オプションを選択します。
- バックアップの複製コピーを作成するには、[スナップショットからバックアップの複製コピーを作成 (Create a duplicate copy of the backup from snapshot)]オプションを選択します。

6 『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションにある説明に従って、[開始時間帯 (Start window)]タブでスケジュールの作成を続行します。

7 『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションにある説明に従って、スナップショットからのバックアップ用に[ストレージオプション (Storage options)]の設定を続行します。

- [スナップショットからのバックアップ (Backup from Snapshot)]オプションにストレージを選択する場合、選択したストレージユニットには NetBackup バージョン 10.0 以降のメディアサーバーが必要です。
- ストレージを管理するメディアサーバーには、選択した Kubernetes クラスタへのアクセス権が必要です。
- メディアサーバーは API サーバーに接続できる必要があります。メディアサーバーからのアウトバウンド接続のために、API サーバーに対応するポートを開く必要があります。datamover ポッドはメディアサーバーに接続できる必要があります。

自動イメージレプリケーション (AIR) と複製の構成

1 つの NetBackup ドメインで生成されたバックアップを、1 つ以上のターゲット NetBackup ドメインのストレージにレプリケートできます。この処理は自動イメージレプリケーション (AIR) と呼ばれます。

NetBackup Kubernetes は、ある NetBackup ドメインのメディアサーバー重複排除プール (MSDP) から、別のドメインのメディアサーバー重複排除プール (MSDP) への自動イメージレプリケーションをサポートします。NetBackup は、AIR 操作を管理するソースドメインとターゲットドメインでストレージライフサイクルポリシー (SLP) を使用します。

自動イメージレプリケーションの構成について詳しくは、『NetBackup 管理者ガイド Vol. 1』の「NetBackup のレプリケーションについて」の章を参照してください。

メモ: Kubernetes AIR 構成には、バージョン 10.0.1 以降の NetBackup プライマリサーバーとメディアサーバーが必要です。

Kubernetes バックアップの自動イメージレプリケーション (AIR) と複製を構成するには

1. 2 台の NetBackup プライマリサーバー間で、自動イメージレプリケーションを構成します。
   - ドメイン間の操作のために、2 台のプライマリサーバー間の信頼関係を確立します。
     - ソースプライマリサーバーにログオンし、左側で[ホスト (Hosts)]、[ホストプロパティ (Host properties)]の順に選択して、ソースプライマリサーバーとターゲットプライマリサーバー間の接続を構築します。
     - ソースプライマリサーバーを選択します。必要に応じて、[接続 (Connect)]をクリックします。次に、[プライマリサーバーの編集 (Edit primary server)]をクリックします。
     - [サーバー (Servers)]をクリックします。[信頼できるプライマリサーバー (Trusted primary servers)]タブで、[追加 (Add)]をクリックしてソースサーバーを追加します。
       - [認証局の検証 (Validate Certificate Authority)]をクリックし、[次へ (Next)]をクリックして認証局の検証に進みます。
       - 信頼できるプライマリサーバーを作成するには、次のオプションから選択します。
         - [信頼できるプライマリサーバーの認証トークンの指定 (Specify authentication token of the trusted primary server)]を選択して既存のトークンを追加するか、ソースプライマリサーバーに新しいトークンを作成します。

- [信頼できるプライマリサーバーのクレデンシャルの指定 (Specify credentials of the trusted primary server)]を選択して、ソースプライマリサーバーのユーザークレデンシャルを追加します。
- [信頼を作成 (Create trust)]をクリックします。

ホストプロパティのデータベースが正常に更新されます。

- [保存 (Save)]をクリックします。

2 ソースプライマリサーバーでメディアサーバー重複排除プール (MSDP) ストレージを構成し、MSDP ディスクプールにレプリケーションターゲットを追加します。

- 左側で[ストレージ (Storage)]、[ディスクストレージ (Disk storage)]の順に選択します。
- MSDP ストレージとディスクプールを追加します。
- [ディスクプール (Disk pools)]タブ、[追加 (Add)]の順に選択します。
  - 信頼できるプライマリサーバーとターゲットストレージサーバーを選択します。
  - [ユーザー名 (Username)]フィールドと[パスワード (Password)]フィールドに、レプリケーションターゲットサーバーのユーザークレデンシャルを追加します。
- [追加 (Add)]をクリックします。

3 ターゲットプライマリサーバーでインポート操作を使用して SLP を作成します。

- 左側で[ストレージ (Storage)]、[ストレージライフサイクルポリシー (Storage lifecycle policy)]の順に選択します。次に[追加 (Add)]をクリックします。
- [ストレージライフサイクルポリシー名 (Storage lifecycle policy name)]フィールドにポリシー名を入力し、[追加 (Add)]をクリックします。
- [操作 (Operation)]リストから、[インポート (Import)]を選択します。
- [宛先ストレージ (Destination storage)]リストで、MSDP ストレージユニットを選択します。
- [作成 (Create)]をクリックします。

4 [スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを指定して Kubernetes 保護計画を作成し、レプリケーションコピーオプションを有効にします。

左側で[保護 (Protection)]、[保護計画 (Protection plans)]の順に選択します。[スケジュール (Schedules)]で、[スケジュールの追加 (Add schedule)]をクリックします。

5 [スナップショットとバックアップコピーのオプション (Snapshot and backup copy options)]セクションで[スナップショットからバックアップを作成 (Create backup from snapshot)]オプションを選択して、レプリケーションおよび複製コピーのオプションを有効にします。

6 [スナップショットからバックアップのレプリカコピー (自動イメージレプリケーション) を作成 (Create a replica copy (Auto Image Replication) of the backup from snapshot)]オプションを選択し、レプリカコピーを保持する期間を設定します。

メモ: 自動イメージレプリケーションは、信頼できる NetBackup プライマリサーバーでのみ作成できます。

---

7 [スナップショットからバックアップの複製コピーを作成 (Create a duplicate copy of the backup from snapshot)]オプションを選択し、複製コピーを保持する期間を設定します。

8 [追加 (Add)]をクリックします。

9 『NetBackup Web UI 管理者ガイド』の「保護計画の管理」セクションにある説明に従って、[開始時間帯 (Start window)]タブでスケジュールの作成を続行します。

10 [次へ (Next)]をクリックします。

11 [ストレージオプション (Storage options)]タブで、スナップショットからのバックアップ、レプリケート、または複製を行うストレージユニットを選択します。

メモ: スナップショットと複製からのバックアップでは、単純なストレージユニットを追加できます。ただし、レプリケーションの場合は、インポートストレージライフサイクルポリシー (SLP) を使用して、信頼できるストレージユニットを追加する必要があります。

---

12 選択したバックアップオプションの右側にある[編集 (Edit)]をクリックして、バックアップ用に選択したストレージユニットを変更します。

- レプリカコピーオプションの場合は、レプリケーションコピー用のプライマリサーバーを選択します。次に[次へ (Next)]をクリックします。
- 信頼できるサーバーで定義されているインポートストレージライフサイクルポリシーを選択し、[選択したレプリケーションターゲットを使用 (Use selected replication target)]をクリックします。

13 ウィザードの手順を続行します。

ストレージユニットの構成

保護計画では、すべての形式のストレージユニットをバックアップ用に構成できます。

メモ: バックアップジョブでは、ストレージライフサイクルポリシー (SLP) でサポートされるすべてのストレージ形式がサポートされます。

ストレージユニットをバックアップ用に構成するには

1. 左側で[ストレージ (Storage)]、[ストレージユニット (Storage units)]の順に選択します。
2. [ストレージユニット (Storage units)]タブをクリックし、[追加 (Add)]をクリックして、ストレージユニットの構成を追加します。
3. リストからストレージの形式を選択します。
4. [カテゴリ (Category)]を選択し、[開始 (Start)]をクリックします。
5. [名前 (Name)]フィールドにストレージユニット名を入力します。
6. [最大並列実行ジョブ数 (Maximum concurrent jobs)]フィールドで、バックアップジョブの最大数を選択します。
7. [最大フラグメントサイズ (Maximum fragment size)]フィールドで、ストレージユニットのフラグメントサイズの最大数を選択し、[次へ (Next)]をクリックします。
8. [ディスクプール (Disk pool)]で、ストレージユニットで使用するディスクプールを選択し、[次へ (Next)]をクリックします。
9. [オンデマンドのみ (On demand only)]オプションはストレージユニットがオンデマンドで排他的に利用可能かどうかを指定します。このストレージユニットを使用するためにポリシーまたはスケジュールを明示的に構成する必要があります。
10. [メディアサーバー (Media servers)]タブで、使用するメディアサーバーを選択し、[次へ (Next)]をクリックします。NetBackup がメディアサーバーを自動で選択するか、ラジオボタンを使用してメディアサーバーを手動で選択できます。
    - すべてのメディアサーバーが NetBackup バージョン 10.0 以降である必要があります
    - ストレージを管理するすべてのメディアサーバーには、選択した Kubernetes クラスタへのアクセス権が必要です。
    - メディアサーバーは API サーバーに接続できる必要があります。メディアサーバーからのアウトバウンド接続のために、API サーバーに対応するポートを開く必要があります。datamover ポッドはメディアサーバーに接続できる必要があります。

11 ストレージユニットの設定を確認し、[保存 (Save)]をクリックします。

12 スケジュールバックアップまたは今すぐバックアップのジョブの詳細を確認するには、[アクティビティモニター (Activity monitor)]タブで[ジョブ ID (Job ID)]をクリックして、バックアップジョブの詳細を表示します。ファイルモードの場合、[ジョブの詳細 (Job Details)]セクションですべてのイメージのバックアップ済みファイルの合計数を確認できます。

ボリュームモードのサポート

NetBackup Kubernetes は、次の機能を使用してサポートします。

- 次の機能をサポートする CSI (Container Storage Interface) プロバイダにおけるモードファイルシステムまたはブロック (あるいはその両方) の PVC (永続ボリューム要求) のバックアップとリストア:
  - PVC スナップショット機能。
  - NFS (Network File System) または他の非ブロックストレージに基づく PVC ボリュームプロビジョニング。
  - ブロックストレージに基づく PVC ボリュームプロビジョニング。

メモ: ボリュームが混在する (VolumeMode: ファイルシステムとブロック) 名前空間のバックアップとリストアは、NetBackup 10.3 以降でサポートされるようになりました。

アプリケーションの一貫したバックアップの構成

データベースなどのアプリケーションを実行しているいくつかのポッドには、アプリケーションの一貫したバックアップを取得するために追加の手順が必要です。

アプリケーションの一貫したバックアップでは、アプリケーションメタデータ、メモリ内の状態、永続ストレージに存在する永続データを把握するためのメカニズムが必要です。リストア中に正常な状態を実現するために、これらのすべての Kubernetes リソースにわたって一貫したアプリケーションバックアップを取得することは、リカバリ処理の合理化に役立ちます。クラッシュ整合バックアップのみが必要な場合、これらの手順は必要ありません。

アプリケーションには、アプリケーションの整合性スナップショットを作成するために入出力 (I/O) 操作を一時停止する手順があり、これはベンダーによって文書化されています。この手順はアプリケーションによって異なるため、それぞれの実行方法の性質が重要になります。これらの手順の内容はお客様の責任です。

NetBackup を使用して Kubernetes 作業負荷を保護する場合、アプリケーションの整合性スナップショットを作成するには、バックアップフックを利用するアプリケーションポッドの注釈を適用します。Kubernetes の注釈は、任意の Kubernetes リソースに適用できる単なるメタデータです。Kubernetes 内のフックはユーザー定義の処理で、任意のコマン

ドまたは複数のコマンドを指定できます。Kubernetes インフラ内で、静止状態を必要とするアプリケーションポッドにこれらの注釈とフックを適用します。

バックアップフックは、前処理 (スナップショットの前) と後処理 (スナップショットの後) の両方に使用されます。データ保護のコンテキストでは、通常、netbackup-pre-backup フックが静止プロシージャまたはコマンドを呼び出し、netbackup-post-backup フックが静止解除のプロシージャまたはコマンドを呼び出すことを意味します。フックの各セットで、コマンドとその適用先のコンテナを指定します。コマンドはコンテナのシェル内では実行されないことに注意してください。したがって、指定の例では、ディレクトリを含む完全なコマンド文字列が使用されます。

アプリケーションの一貫したバックアップを必要とするアプリケーションを識別し、Kubernetes データ保護の構成の一部として一連のバックアップフックを含む注釈を適用します。

ポッドに注釈を追加するには、Kubernetes UI (ユーザーインターフェース) を使用します。または、特定のポッドまたはラベルに対して、Kubernetes クラスタコンソールで `kubectl` の注釈機能を使用します。注釈を適用する方法はディストリビューションによって異なる場合があるため、次の例では、ほとんどのディストリビューションでの広範な可用性に基づいて、`kubectl` コマンドに重点を置いて説明します。

さらに、配備リソースやレプリカセットリソースなどの基本 Kubernetes オブジェクトに注釈を追加して、新しく配備されたポッドに注釈を確実に含めることができます。Kubernetes 管理者は注釈を動的に更新できます。

ラベルは、ポッドやサービスなどの Kubernetes オブジェクトに関連付けられるキーと値のペアです。ラベルは、ユーザーにとって重要で関連性のあるオブジェクトの属性として使用されます。ラベルは作成時にオブジェクトにアタッチでき、その後いつでも追加および変更できます。Kubernetes では、これらのラベルを使用して、選択したサブセットに対してオブジェクトを問い合わせ、一括操作を実行するための統合サポートが提供されます。各オブジェクトには、キーと値のラベルのセットを定義できます。各キーは特定のオブジェクトに対して一意である必要があります。

ラベルメタデータのフォーマットと構文の例を次に示します。

```
"metadata": {"labels": {"key1":"value1","key2":"value2"}}
```

ポッド名を具体的に指定するか、ポッドの目的のグループに該当するラベルを指定します。複数の注釈引数を使用する場合は、次に示す JSON アレイのように正しい JSON 形式を指定します: '["item1","item2","itemn"]'`# kubectl annotate pod [ {pod_name} | -l {label=value}] -n {the-pods-namespace_name} [annotation syntax - see following]`

目的の結果を得るために一部のアプリケーションで複数のコマンドが必要な場合は、複数のコマンドを結合するために、このメソッドを `&&` と組み合わせることができます。指定したコマンドはベリタスによって提供されないため、ユーザーは手動でアプリケーションポッドをカスタマイズする必要があります。{values} を環境で使用されている実際の名前に置き換えます。

メモ: すべての kubectl コマンドは 1 行で定義する必要があります。次の例をコピーまたは貼り付けるときは注意してください。

NetBackup 10.2 にアップグレードした後、これらの新しい netbackup-pre と netbackup-post のバックアップフックに対する注釈を更新します。これには、「netbackup」の接頭辞が含まれています。

```
netbackup-pre.hook.back.velero.io/command
netbackup-pre.hook.backup.velero.io/container
netbackup-post.hook.back.velero.io/command
netbackup-post.hook.backup.velero.io/container
```

ポッド名を使用した MongoDB の例

MongoDB 4.2.23 データベースをロックおよびロック解除するコマンドを次に示します。

```
# mongo --eval "db. fsyncLock ()"

# mongo --eval "db.fsyncUnlock()"
```

これは、MongoDB のバックアップ前と後の両方のフックを設定する次の単一のコマンドに変換されます。特殊文字と、JSON 形式の一部として使用される角カッコ ([])、一重引用符、二重引用符、およびカンマ (,) をエスケープするための特殊な構文に注意してください。

```
# kubectl annotate pod {mongodb-pod-name} -n {mongodb namespace}
netbackup-pre.hook.back.velero.io/command='["/bin/bash", "-c", "mongo
--eval ¥"db.fsyncLock()¥""]'
netbackup-pre.hook.backup.velero.io/container={mongodb-pod-name}
netbackup-post.hook.backup.velero.io/command='["/bin/bash","-c","mongo
--eval ¥"db.fsyncUnlock()¥""]'
netbackup-post.hook.backup.velero.io/container={mongodb-pod-name}
```

ラベルを使用した MySQL の例

次に、MySQL データベースを静止および静止解除するコマンドを示します。

```
# mysql -uroot -ppassword -e "flush tables with read lock"

# mysql -uroot -ppassword -e "unlock tables"
```

これは、MySQL のバックアップ前と後の両方のフックを設定する次の単一のコマンドに変換されます。この例では、ポッド名の代わりにラベルを使用しており、このラベルは一度に複数のポッドに注釈を付けることができます。特殊文字と、JSON 形式の一部として使用される角カッコ ([])、一重引用符、二重引用符、およびカンマ (,) をエスケープするための特殊な構文に注意してください。

```
# kubectl annotate pod -l label=value -n {mysql namespace}
netbackup-pre.hook.backup.velero.io/command='["/bin/bash", "-c",
"mysql -uroot -ppassword -e ¥"flush tables with read lock¥""]'
netbackup-pre.hook.backup.velero.io/container={mysql container name}
netbackup-post.hook.backup.velero.io/command='["/bin/bash", "-c",
"mysql -uroot -ppassword -e ¥"unlock tables¥""]'
netbackup-post.hook.backup.velero.io/container={mysql container name}
```

ラベルを使用した Postgres の例

次に、PostgreSQL データベースを静止および静止解除するコマンドを示します。

```
# Psql -U postgres -c "SELECT pg_start_backup('tagvalue');"
```

```
# psql -U postgres -c ¥"SELECT pg_stop_backup();"
```

これは、Postgres のバックアップ前と後の両方のフックを設定する次の単一のコマンドに変換されます。この例では、ポッド名の代わりにラベルを使用しており、ラベルは一度に複数の一致するポッドに注釈を付けることができます。任意の Kubernetes オブジェクトにラベルを適用できます。この場合は、特定のコンテナを変更し、特定のポッドのみを選択するための別の方法としてラベルを使用しています。特殊文字と、JSON 形式の一部として使用される角カッコ ([])、一重引用符、二重引用符、およびカンマ (,) をエスケープするための特殊な構文に注意してください。

```
# kubectl annotate pod -l app=app-postgresql -n {postgres namespace}
netbackup-pre.hook.backup.velero.io/command='["/bin/bash", "-c",
"psql -U postgres -c ¥"SELECT
pg_start_backup(quote_literal($EPOCHSECONDS));¥""]'
netbackup-pre.hook.backup.velero.io/container={postgres container
name} netbackup-post.hook.backup.velero.io/command='["/bin/bash",
"-c", "psql -U postgres -c ¥"SELECT pg_stop_backup();¥""]'
netbackup-post.hook.backup.velero.io/container={postgres container
name}
```

コンテナフックなしの NGINX アプリケーションの例

次に、NGINX アプリケーションを静止および静止解除するコマンドを示します。

```
# /sbin/fsfreeze --freeze /var/log/nginx
```

```
# /sbin/fsfreeze --unfreeze /var/log/nginx
```

これは、NGINX のバックアップ前と後の両方のフックを設定する次の単一のコマンドに変換されます。この例では、コンテナフックを省略します。これにより、デフォルトでポッド名と一致する最初のコンテナが変更されます。特殊文字と、JSON 形式の一部として使用される角カッコ ([])、一重引用符、二重引用符、およびカンマ (,) をエスケープするための特殊な構文に注意してください。

```
# kubectl annotate pod {nginx-pod-name} -n {nginx namespace}
netbackup-pre.hook.backup.velero.io/command='["/sbin/fsfreeze",
"--freeze", "/var/log/nginx"]'
netbackup-post.hook.backup.velero.io/command='["/sbin/fsfreeze",
"--unfreeze", "/var/log/nginx"]'
```

Cassandra の例

次に、Cassandra データベースを静止および静止解除するコマンドを示します。

```
# nodetool flush
```

```
# nodetool verify
```

これは、Cassandra のバックアップ前と後の両方のフックを設定する次の単一のコマンドに変換されます。特殊文字と、JSON 形式の一部として使用される角カッコ (**[]**)、一重引用符 (‘’)、二重引用符 (“”)、およびカンマ (**,**) をエスケープするための特殊な構文に注意してください。

```
# kubectl annotate pod {cassandra-pod} -n {Cassandra namespace}
netbackup-pre.hook.backup.velero.io/command='["/bin/bash", "-c",
"nodetool flush"]'
netbackup-pre.hook.backup.velero.io/container={cassandra-pod}
netbackup-post.hook.backup.velero.io/command='["/bin/bash", "-c",
"nodetool verify"]'
netbackup-post.hook.backup.velero.io/container={cassandra-pod}
```

メモ: ここに示す例では、初期ガイドのみが提供されています。各作業負荷の特定の要件には、バックアップ、作業負荷、Kubernetes 管理者間のコラボレーションが含まれている必要があります。

現時点では、Kubernetes はエラー時のフックをサポートしていません。ユーザー指定のコマンドが失敗した場合、バックアップスナップショットは続行されません。

終了状態を返すコマンドのデフォルトのタイムアウト値は 30 秒です。ただし、この値は、ポッドへの注釈として次のフックで変更できます。

```
netbackup-pre.hook.backup.velero.io/timeout=#in-seconds#
```

```
netbackup-post.hook.backup.velero.io/timeout=#in-seconds#
```

7

イメージグループの管理

この章では以下の項目について説明しています。

イメージグループについて

各 Kubernetes リカバリポイントに対して、イメージグループが作成されます。イメージグループには、名前空間内の対象永続ボリューム要求の数に応じて、複数のイメージが含まれる場合があります。

メタデータに対して個別のイメージが作成され、永続ボリューム要求ごとに 1 つのイメージが作成されます。

リカバリポイントの詳細 API は、イメージグループのすべてのバックアップ ID、リソース名、コピー完了状態についての詳細を取得するために使用します。

Kubernetes 作業負荷に対するスナップショットからのバックアップ操作をサポートするために、単一の名前空間に対して複数のバックアップイメージが作成され、スナップショットからのバックアップが実行されます。

Kubernetes のバックアップ操作では、すべての永続ボリュームに対して個別のバックアップイメージが作成されます。特定の操作 (リストア、削除、インポートなど) を正常に実行するには、作成されたすべてのイメージをグループ化する必要があります。

イメージの期限切れ

期限切れのイメージが占有するストレージ領域を再利用するには、それらのイメージを削除する必要があります。

イメージの有効期限に関連した重要なポイントを次に示します。

複数のイメージで構成されるリカバリポイントの場合:

- イメージグループ内の 1 つのイメージを期限切れにしても、残りのイメージの有効期限が自動的に切れることはありません。イメージグループ内のすべてのイメージを明示的に期限切れにする必要があります。
- いくつかのイメージを期限切れにした場合、リカバリポイントは不完全になります。リカバリポイントが不完全な場合、リストア操作はサポートされません。
- いずれかのイメージの有効期限を変更した場合は、残りのイメージの有効期限を変更する必要があります。そうしないと、リカバリポイントに対応するイメージの有効期限にずれが生じ、ある時点でリカバリポイントが不完全になります。

イメージのインポート

Kubernetes のリカバリポイントは、複数のイメージで構成されている場合があります。リストア操作を実行するには、リカバリポイントに対応するすべてのイメージをインポートする必要があります。そうしないと、リカバリポイントは不完全とマークされ、リストアは実行されません。

詳しくは、『NetBackup™ 管理者ガイド Vol. 1』の「バックアップイメージのインポートについて」セクションを参照してください。

イメージのコピー

イメージコピーは、次の 2 種類のバックアップ操作で作成できます。

1. スナップショットはデフォルトのコピーで、コピー番号 1 としてマークされます。
2. スナップショットからのバックアップはコピー番号 2 としてマークされます。

任意の今すぐバックアップ操作またはスケジュールバックアップがトリガされるたびに、スナップショットが作成されます。ただし、[スナップショットからのバックアップ (Backup from snapshot)]は、保護計画の作成時に[スナップショットからのバックアップ (Backup from snapshot)]オプションが選択されているかどうかによって左右されるため、任意です。

イメージグループは、メタデータの資産のイメージや永続ボリューム要求 (PVC) で構成されます。各コピーには、名前空間用に 1 つのイメージ、名前空間に存在する各 PVC 用に 1 つのイメージが含まれます。

リカバリポイントの詳細 API を使用して、イメージのコピー完了状態を識別します。この API では、それぞれのコピーに存在するすべてのバックアップ ID とリソース名も詳細に示されます。不完全なイメージコピーから資産をリストアしようとするとエラーが発生するため、イメージコピーが完全な状態か不完全な状態かはリストア機能に役立ちます。

不完全なイメージコピー

不完全なイメージの条件を次に示します。

1. スナップショットジョブまたはスナップショットからのバックアップジョブが進行中の場合、対応するコピーが不完全なコピーとして表示されます。
2. PVC のバックアップ処理が失敗すると、コピーは不完全としてマークされます。

3. コピーの子イメージ (複数の子を含む) が期限切れになると、コピーは不完全としてマークされます。

8

NetBackup でのランチャ管理クラスタの保護

この章では以下の項目について説明しています。

自動構成を使用した NetBackup へのランチャ管理 RKE クラスタの追加

次の手順に従って、自動構成を使用して NetBackup にランチャ管理 RKE クラスタを追加します。

自動構成を使用して NetBackup にランチャ管理 RKE クラスタを追加するには

メモ: グローバルランチャ管理サーバー証明書を抽出します。この CA 証明書は、管理サーバーの作成時に、ランチャによってデフォルトで生成される証明書、または別の/外部の CA (認証機関) を使用して構成される証明書である場合があります。

---

1 CA 証明書を抽出します。ランチャ管理サーバーの UI に移動して、左側のパネルの[グローバル設定 (Global Settings)]の[CA 証明書 (CA Certificate)]の下で、[CA 証明書の表示 (Show CA Certificate)]ボタンをクリックします。一時ファイルに CA 証明書の完全な値を抽出します。

メモ: 開始行と終了行を含む完全な値を抽出してください。

---

2 CA 証明書の値が、Kubernetes Operator の Helm インストールの前に作成されるシークレットに追加されます。

3 トークンを抽出するには、次のようにします。ランチャ管理サーバーの UI で左側のパネルを開き、[Explore Cluster]セクションで保護するクラスタに移動し、右上隅にある[Download KubeConfig]アイコンをクリックします。

4 アイコンを使用してクラスタの KubeConfig をダウンロードします。トークンフィールドがファイル内に存在します。

5 ダウンロードした Kubeconfig ファイルから、二重引用符を除いてトークンの値を抽出します。

6 この構成プロセスでは、シークレットの命名パターン (`<kops-namespace>-nb-config-deploy-secret`) が使用されます。

シークレットには、手順 1 と手順 3 で抽出した値があります。

7 次の形式で yaml ファイル `nb-config-deploy-secret. yaml` を作成し、すべてのフィールドに値を入力します。

```
apiVersion: v1
kind: secret
metadata:
   name: <kops-namespce>-nb-config-deploy-secret
   namespace:  <kops-namespace>

type: Opaque
stringData:
#All the 3 fields are mandatory here to add a Rancher managed
RKF2 cluster in NetBackup
     apikey:
A_YoUkgYQwPLUkmyj9Q6A1-6RX8RNY-PtYX0SukbqCwIK-osPz8qVm9zCL9phje

     k8stoken:
kubeconfig-user-mvvgcm8sq8:nrscvn8hj46t24r2tjrxd2kn8tzo2bg4kj8waxpw36k8ktrchp826

     k8scacert: |
-------BEGIN CERTIFICATE-----
MIIDDDCCAfSgAwIBAgIBATANBgkqhkiG9w0BAQwIgYDVQQDDBtpbmdy
ZXNzLW9wZXJhdG9yQDE2ODc1MzY4NjgWHhcNMjMwNjIzMTYxNDI3WhcNMjUwNjIy
XtXqbaBGrXIuCCo90mxv4g==
-------END CERTIFICATE------
```

8 コマンド `kubectl apply -f nb-config-deploy-secret. yaml` を実行します。

9 Helm Chart の values.yaml ファイルの残りの入力については、『Kubernetes クイックスタートガイド』の自動構成に関するセクションを参照し、設定の完了に必要なすべての値を入力します。

10 必要なすべての簡易インストールの入力が values.yaml ファイルに追加されたら、NetBackup Kubernetes Operator グラフで Helm のインストールコマンドを実行します。自動化された構成ポッド <kops-namespace>-netbackup-config-deploy が起動するはずです。

11 <kops-namespace>-netbackup-config-deploy ログを確認して、更新されたシークレット値が config-deploy ポッドによって選択されたかどうかを確認します。

12 config-deploy ポッドがそのタスクを実行すると、クラスタが NetBackup に正常に追加され、検出要求が進行中になるか、正常に完了します。NetBackup Web UI から別のクレデンシャルの検証と手動検出を実行して、プロセスが正常に動作していることを確認します。

NetBackup でのランチャ管理 RKE クラスタの手動での追加

次の手順に従って、NetBackup でランチャ管理 RKE クラスタを手動で追加します。

NetBackup 用の Kubernetes クレデンシャルの作成

NetBackup Web UI で[クレデンシャルの管理 (Credential Management)]、[指定したクレデンシャル (Named Credentials)]、[追加 (Add)]、[クレデンシャルを追加 (Add credentials)]の順に移動し、クレデンシャルストアを NetBackup として選択し、[カテゴリ (Category)]フィールドで Kubernetes を選択し、以前に Global Rancher Management プラットフォーム UI から抽出したトークンと CA 証明書を入力し、このクレデンシャルを保存します。

NetBackup でランチャ管理 RKE クラスタを手動で追加するには

1 外部 CA 証明書: 外部アクセス用の証明書の構成に使用されている CA (認証機関) が異なる場合、NetBackup がクラスタと正常に通信するために外部 CA 証明書が必要です。

- ランチャ管理サーバーの UI に移動して、左側のパネルの[Global Settings]の[cacerts]の下で、[showcacerts]ボタンをクリックします。
  一時ファイルにこの CA 証明書の完全な値を抽出します。
- たとえば、<cacert-value-file>

2 サービスアカウントの CA 証明書:

メモ: クラスタ内で利用可能なサービスアカウントの CA 証明書と比較して、Kubernetes API サーバーの外部アクセス用に構成されている CA (認証機関) が異なるため、次の手順を実行する必要があります。そのため、これらの 2 つの CA 証明書を組み合わせる必要があります。

サービスアカウントの CA 証明書を取得するには、Linux クラスタホストで次のコマンドを実行します。

- 次のコマンドを使用して、Kubernetes Operator の名前空間で利用可能なサービスアカウントシークレット名を取得します。

  ```
  kubectl describe serviceaccount <kopsnamespace>-backup-server
  -n <kopsnamespace> | grep Tokens | cut -d ":" -f 2
  ```

- 次のコマンドを使用して、このサービスアカウントのシークレットから base 64 デコードされた形式で CA 証明書を取得します。

  ```
  kubectl get secret <output-from-previous-command> -n
  <kopsnamespace> -o jsonpath='{. data.ca¥.crt}' | base64 -d
  ```

  このコマンドの出力全体を、手順 1 で作成した一時ファイルに追加する必要があります。

3 手順 2 の後に生成された出力を <cacert-value-file> ファイルの最後に追加します。必要な外部 CA 証明書と内部 CA 証明書の値が抽出され、ファイル <cacert-value-file> で利用可能になります。CA 証明書の値は base 64 でデコードされた形式ですが、NetBackup でクレデンシャルを作成するときに再度エンコードする必要があります。

4 トークン: ランチャ管理サーバーの UI で左側のパネルを開き、[Explore Cluster]セクションで保護するクラスタに移動し、右上隅にある[Kubeconfig]アイコンをクリックします。

- ([Download KubeConfig]を使用して) ダウンロードした Kubeconfig ファイルから、一時ファイル <token-value-file> に二重引用符を除いたトークンの値を抽出します。
- これらのトークンと cacert の両方のフィールドは、Kubernetes の NetBackup クレデンシャルに追加するために、base64 エンコード形式であることが必要です。
- 次の base64 コマンドを使用して、これらの抽出された両方の値の base64 エンコードバージョンを取得する方法を示します。

  #Linux VM を使用して、この手順の値をエンコードします #注意: フラグ -w0 には 0 の記号ではなく数字のゼロを使用します。
  #CA 証明書:
  Cat <cacert-value-file>| base64 -w0
  この出力を、NetBackup のクレデンシャル作成ページの[CA 証明書 (CA certificate)]フィールドに貼り付けます。
  #トークン:
  この出力を、NetBackup Web UI のクレデンシャル作成ページの[トークン (Token)]フィールドに貼り付けます。

- これらの値を、NetBackup Web UI の[クレデンシャルの管理 (Credential management)]、[指定したクレデンシャル (Named Credentials)]、[追加 (Add)]で使用して、NetBackup に有効なランチャクレデンシャルを追加します。
- クレデンシャルが作成されたら、次のクラスタ情報の出力に示されている名前を使用して、NetBackup に Kubernetes クラスタを追加します。

クラスタ情報の出力を取得するには、次のコマンドを実行します。

1. クラスタ情報の出力は、次の例の形式である必要があります。`[root@master-0~] # kubectl cluster-info`
2. Kubernetes コントロールプレーンは、https://<rancher-hostname>/k8s/clusters/c-m-zjrfft56 で実行されます
3. CoreDNS は、https://<rancher-hostname>/k8s/clusters/c-m-zjrfft56/api/v1/ で実行されます

   namespaces/kube-system/services/rke2-coredns-rke2-coredns:udp-53/proxy
4. 出力から、API サーバーのエンドポイント全体 (https:// を含む) を抽出します。形式は、https://<rancher-hostname>/k8s/clusters/c-m-zjrfft56 のパターンになるはずです
5. NetBackup Web UI の[作業負荷 (Workloads)]、[Kubernetes]、[Kubernetes クラスタ (Kubernetes clusters)]、[追加 (Add)]で、ランチャクラスタ名全体を追加します。
6. [Kubernetes クラスタの追加 (Add Kubernetes cluster)]ページで、URL またはエンドポイントに関連付けられているオプションを選択して、(https://) を含むエンドポイントに基づいてクラスタを追加できるようにします。

   メモ: エンドポイントベースのアプローチを使用して追加したクラスタ名は編集できません。そのようなクラスタ名は、削除および再追加だけを行えます。
7. NetBackup Web UI (エンドポイントまたは URL) の入力フィールドに、上記で抽出したクラスタ情報の出力を入力します。
8. 先に進み、手順 1 から 4 で準備したクレデンシャルを選択または作成します。
9. クレデンシャルが検証されると、クラスタが正常に追加されます。自動検証と検出がトリガされます。
10. 自動検出が正常に実行されたら、ユーザーは手動のクレデンシャル検証と検出を試み、すべてが正常に動作していることを確認します。

11 NetBackup でランチャ管理クラスタを追加します。

12 BFS (スナップショットからのバックアップ) 機能を設定するバックアップサーバー証明書シークレットとデータムーバー configmap を作成します。

次に、推奨されるセットアップガイドに従って、残りの構成手順に進みます。

9

Kubernetes 資産のリカバリ

この章では以下の項目について説明しています。

リカバリポイントの検索と検証

NetBackup バージョン 10.0 以降では、スナップショットからのリストア操作とバックアップコピーからのリストア操作による Kubernetes 資産のリカバリをサポートしています。

メモ: リカバリ後、新しく作成された名前空間、永続ボリューム、その他のリソースには、新しいシステム生成 UID が割り当てられます。

NetBackup は、Kubernetes 作業負荷のバックアップコピーの完了状態または未完了状態を通じてバックアップイメージの検証を実行するのに役立ちます。NetBackup では、未完了のバックアップコピーからリストア操作を実行できません。

Kubernetes 名前空間に対応するリカバリポイントは、複数のイメージで構成されます。一部のイメージのコピーが利用できない可能性があるため、リカバリポイントは未完了である可能性があります。このようなリカバリポイントは未完了としてマークされます。

リカバリポイントの検証を実行するには

1. 左側の［作業負荷 (Workloads)］で、［Kubernetes］をクリックします。
2. ［名前空間 (Namespaces)］タブで、リカバリする資産の名前空間をクリックします。

3 [リカバリポイント (Recovery points)]タブをクリックします。

4 [リカバリポイント (Recovery points)]タブには、すべてのリカバリポイントがバックアップの日時およびコピーとともに表示されます。

リカバリポイントの横にあるコピー数のボタンをクリックすると、場所、デフォルトのコピー、コピーの種類、完了状態が表示されます。

完了状態は、リストア操作を実行するために選択したリカバリポイントを検証するのに役立ちます。

未完了のバックアップコピー、進行中のバックアップ、イメージの期限切れ、ハードウェア障害、またはネットワーク通信の問題には、複数の理由が考えられます。

スナップショットからのリストア

NetBackup は、単一のリストアジョブを使用して、リカバリポイントでのすべてのバックアップイメージをリストアできる、スナップショットからのリストア機能を備えています。アクティビティモニターにスナップショットジョブからのリストアを表示できます。

スナップショットからリストアするには

1 左側で[作業負荷 (Workloads)]、[Kubernetes]の順に選択します。

2 [名前空間 (Namespace)]タブで、リカバリする資産の名前空間をクリックします。

3 [リカバリポイント (Recovery points)]タブをクリックします。

[リカバリポイント (Recovery points)]タブには、すべてのリカバリポイントがバックアップの日時およびコピーとともに表示されます。フィルタを設定して、表示されたリカバリポイントをフィルタ処理できます。[日付 (Date)]列の日付をクリックすると、リカバリポイントの詳細が表示されます。[リカバリポイントの詳細 (Recovery points details)]ダイアログには、構成マップ、Secret、永続ボリューム、ポッドなど、バックアップされたリソースが表示されます。これらのリソースについて詳しくは、https://kubernetes.io/docs/reference/kubernetes-api/ を参照してください。

4 リストアするリカバリポイントを見つけます。

5 [コピー (Copies)]列で、[# コピー (# copies)]ボタンをクリックします。たとえば、コピーが 2 つある場合、ボタンには[2 コピー (2 copies)]と表示されます。

6 コピーのリストで、スナップショットコピーを見つけます。[処理 (Actions)]、[リストア (Restore)]の順に選択します。

7 [リカバリターゲット (Recovery target)]ページに、ターゲットクラスタが自動入力されます。

メモ: 代替クラスタのリストアは、スナップショットコピーではサポートされません。

---

8 [宛先名前空間を指定 (Specify destination namespace)]で、次のいずれかのリストアオプションを選択します。

- バックアップされた元の名前空間をリストアに使用する場合は[元の名前空間を使用 (Use original namespace)]。デフォルトでは、このオプションが選択されています。
- リストアに代替名前空間を使用する場合は[代替名前空間を使用 (Use alternate namespace)]。その後、[次へ (Next)]をクリックします。

9 [リカバリするリソース形式の選択 (Select resource types to recover)]で、次のいずれかのリストアするリソース形式を選択します。

- すべてのリソース形式をリカバリする場合は[すべてのリソース形式 (All resource types)]。デフォルトでは、このオプションが選択されています。
- 選択したリソース形式のみをリカバリする場合は[選択されたリソース形式のリカバリ (Recover selected resource types)]。

メモ: [リカバリするリソース形式の選択 (Select resource types to recover)]オプションは、上級ユーザー向けです。リストアするリソースの選択に注意しないと、リストア後に完全に機能する名前空間が得られない場合があります。

---

10 [リカバリする永続ボリューム要求の選択 (Select Persistent volume claims to recover)]で、次のいずれかのリカバリする永続ボリューム要求を選択します。

- すべての永続ボリューム要求をリカバリする場合は[すべての永続ボリューム要求 (All Persistent volume claims)]。デフォルトでは、このオプションが選択されています。
- 選択した永続ボリューム要求をリカバリする場合は[選択された永続ボリューム要求のリカバリ (Recover selected Persistent volume claims)]。

メモ: [選択されたリソース形式のリカバリ (Recover selected resource types)]でオプションを選択せずに空の永続ボリューム要求を含めるオプションが選択されている場合、永続ボリューム要求はリストアされません。

[選択された永続ボリューム要求のリカバリ (Recover selected persistent volume claims)]でオプションを選択しない場合、[リカバリオプション (Recovery options)]セクションで永続ボリューム要求は空になり、永続ボリューム要求はリストアされません。

メモ: [永続ボリュームのみリストア (Restore only persistent volume)]を使用すると、選択した永続ボリューム要求で、永続ボリュームのみをリストアするよう切り替えることができます。この設定により、対応する永続ボリューム要求が作成されることはありません。

11 [エラー戦略 (Failure strategy)]セクションをクリックして、リカバリのためのエラー戦略オプションを表示します。

12 [リカバリのためのエラー戦略を選択 (Select failure strategy to recover)]で、次のリカバリのためのエラー戦略のいずれかを選択します。

メモ: メタデータまたは PVC のリストア中にエラーが発生すると、選択したエラー戦略に従ってリストアジョブが実行されます。

- [即座に終了 (Fail Fast)]を選択すると、エラーが発生した場合にリストアを終了します。
- [先に進む (Proceed Ahead)]を選択すると、次の PVC のリストアを続行します。親イメージ (最初のイメージ) のリストアが失敗した場合、リストアジョブは終了します。
- [再試行 (Retry)]では、メタデータまたは PVC リストアの再試行回数を指定します。再試行後もリストアが失敗した場合、リストアジョブは終了します。

メモ: 選択したエラー戦略がアクティビティモニターに表示されます。

13 [次へ (Next)]をクリックします。

14 [リカバリオプション (Recovery options)]ページで、[リカバリの開始 (Start recovery)]をクリックしてリカバリのエントリを送信します。

15 [アクティビティモニター (Activity monitor)]で、[ジョブ ID (Job ID)]をクリックし、リストアジョブの詳細を表示します。

メモ: NetBackup Kubernetes のリストアでは、単一ジョブを使用してすべての永続ボリューム要求と 1 つの名前空間をリストアします。[アクティビティモニター (Activity monitor)]でログを表示して、永続ボリューム、永続ボリューム要求、またはメタデータのリストアを追跡できます。

バックアップコピーからのリストア

選択した名前空間に複数の PVC が存在する場合、バックアップからの NetBackup リストアも並行して行われます。(名前空間にリストアする PVC が少なくとも 1 つ含まれる場合) バックアップジョブからリストアを開始すると、親子階層が作成されます。ここで、親はオーケストレータとして機能し、子ジョブの状態を監視します。最初の子ジョブはメタデータをリストアし、その後 PVC が並行してリストアされます。

メモ: メタデータのリストアが失敗した場合、これ以上のジョブはリストア操作のために送信されません。メタデータが正常にリストアされると、PVC はバッチで並列してリストアされます。

スナップショットからのリストアで説明したのと同じ手順に従い、コピー形式として[バックアップ (Backup)]を選択します。代替ターゲットクラスタにもリストアできます。

バックアップコピーからリストアするには

1. 左側で[作業負荷 (Workloads)]、[Kubernetes]の順に選択します。
2. [名前空間 (Namespace)]タブで、リカバリする資産の名前空間をクリックします。[リカバリポイント (Recovery points)]タブをクリックします。
3. [リカバリポイント (Recovery points)]タブには、すべてのリカバリポイントがバックアップの日時およびコピーとともに表示されます。フィルタを設定して、表示されたリカバリポイントをフィルタ処理できます。[日付 (Date)]列の日付をクリックすると、リカバリポイントの詳細が表示されます。[リカバリポイントの詳細 (Recovery points details)]ダイアログには、ConfigMap、Secret、永続ボリューム、ポッドなど、バックアップされたリソースが表示されます。これらのリソースについて詳しくは、https://kubernetes.io/docs/reference を参照してください。
4. リストアするリカバリポイントを見つけます。
5. [コピー (Copies)]列で、[# コピー (# copies)]ボタンをクリックします。たとえば、コピーが 2 つある場合、ボタンには[2 コピー (2 copies)]と表示されます。
6. コピーのリストで、[バックアップ (Backup)]コピーを見つけます。[処理 (Actions)]、[リストア (Restore)]の順に選択します。
7. [リカバリターゲット (Recovery target)]ページで、資産を同じソースクラスタにリカバリすることが自動的に入力されます。[次へ (Next)]をクリックします。
8. [宛先名前空間を指定 (Specify destination namespace)]で、次のいずれかのリストアオプションを選択します。
   - [元の名前空間を使用 (Use original namespace)]を選択して、元の名前空間を使用します。デフォルトでは、このオプションが選択されています。
   - [代替名前空間を使用 (Use alternate namespace)]を選択して、代替名前空間を入力し、[次へ (Next)]をクリックします。

9 [リカバリするリソース形式の選択 (Select resource types to recover)]で、次のいずれかのリストアするリソース形式を選択します。

- すべてのリソース形式をリカバリする場合は[すべてのリソース形式 (All resource types)]。デフォルトでは、このオプションが選択されています。
- 選択したリソース形式のみをリカバリする場合は[選択されたリソース形式のリカバリ (Recover selected resource types)]。

10 [リカバリする永続ボリューム要求の選択 (Select Persistent volume claims to recover)]で、次のいずれかのリカバリする永続ボリューム要求を選択します。

- すべての永続ボリューム要求をリカバリする場合は[すべての永続ボリューム要求 (All Persistent volume claims)]。デフォルトでは、このオプションが選択されています。
- 選択した永続ボリューム要求をリカバリする場合は[選択された永続ボリューム要求のリカバリ (Recover selected Persistent volume claims)]。

メモ: [選択された永続ボリューム要求のリカバリ (Recover selected Persistent volume claims)]でオプションを選択せずに[次へ (Next)]をクリックした場合、[リカバリオプション (Recovery Options)]セクションで永続ボリューム要求は空になり、永続ボリューム要求はリストアされません。

[選択された永続ボリューム要求のリカバリ (Recover selected persistent volume claims)]でオプションを選択しない場合、[リカバリオプション (Recovery options)]セクションで永続ボリューム要求は空になり、永続ボリューム要求はリストアされません。

メモ: [永続ボリュームのみリストア (Restore only persistent volume)]を使用すると、選択した永続ボリューム要求で、永続ボリュームのみをリストアするよう切り替えることができます。この設定により、対応する永続ボリューム要求が作成されることはありません。

---

11 [エラー戦略 (Failure strategy)]セクションをクリックして、リカバリのためのエラー戦略オプションを表示します。

12 [リカバリのためのエラー戦略を選択 (Select failure strategy to recover)]で、次のリカバリのためのエラー戦略のいずれかを選択します。

メモ: メタデータまたは PVC のリストア中にエラーが発生すると、選択したエラー戦略に従ってリストアジョブが実行されます。

- [即座に終了 (Fail Fast)]を選択すると、エラーが発生した場合にリストアを終了します。

- このリストアエラー戦略は、最初のエラーが発生したときにリストアジョブを終了するのに役立ちます。
- 現在のバッチの残りのすべての実行中のリストアジョブは完了でき、それ以降のバッチはリストアのために送信されません。

- ［先に進む (Proceed Ahead)］を選択すると、次の PVC のリストアを続行します。親イメージ (最初のイメージ) のリストアが失敗した場合、リストアジョブは終了します。
  - この戦略は、進行中のバッチで PVC リストアのいずれかが失敗した場合に、残りの PVC のリストアを進めるのに役立ちます。
  - メタデータリストアが失敗すると、最終的なジョブは失敗としてマークされ、リストア用の PVC は送信されません。
  - この場合、部分的な成功とマーク付けされた最終的なジョブ状態と、失敗状態の PVC のリストが親ジョブの［アクティビティモニター (Activity Monitor)］タブに表示されます。
- ［再試行 (Retry)］では、メタデータまたは PVC リストアの再試行回数を指定します。再試行後もリストアが失敗した場合、リストアジョブは終了します。
  - このエラー戦略は、リストアジョブの開始時に構成可能な、失敗した PVC またはメタデータのリストアジョブを再試行するのに役立ちます。
  - 再試行の最大数にかかわらずリストアジョブが失敗した場合、失敗とマーク付けされたジョブは失敗し、それ以降のバッチはリストアのために送信されません。

メモ: 選択したエラー戦略がアクティビティモニターに表示されます。

- ［次へ (Next)］をクリックします。

13 ［リカバリの開始 (Start recovery)］をクリックしてリカバリのエントリを送信します。

14 ［アクティビティモニター (Activity monitor)］で、［ジョブ ID (Job ID)］をクリックし、リストアジョブの詳細を表示します。

15 ［ジョブの詳細 (Job Details)］ページで、［詳細 (Details)］タブをクリックします。リストアジョブのシーケンス (リストア前、データの移動、リストア後のジョブ) が表示されます。

メモ: 親ジョブを取り消して、リストア操作を取り消すことができます。親ジョブによって、実行中のすべての子リストアジョブが終了します。

構成の変更

並列 PVC リストアのバッチサイズは bp.conf で構成できます。ユーザーは目的のバッチサイズを設定するために bp.conf ファイルにキー「KUBERNETES_RESTORE_FROM_BACKUP_COPY_PARALLEL_RESTORE_BATCH_SIZE」を追加できます。これはオプションのキーであり、定義されていない場合は値 5 を受け取ります。

バッチサイズに割り当てることができる最小値は 1 であるのに対し、最大値は 100 です。

bpsetconfig コマンドを NetBackup プライマリサーバーで使用して、バッチサイズを更新できます。

10

Kubernetes での FIPS モードの有効化

この章では以下の項目について説明しています。

Kubernetes での FIPS (連邦情報処理標準) モードの有効化

NetBackup Kubernetes 10.3 リリースでは、RedHat ベースの NetBackup 配備に FIPS サポートを提供します。NetBackup、Kubernetes Operator、データムーバーに関連するすべての Kubernetes 作業負荷コンポーネントは、FIPS モードで実行する必要があります。FIPS サポートを実現するには、これらのすべてのコンポーネントで満たす必要がある特定の要件があります。

システム要件

NetBackup Kubernetes 作業負荷の FIPS サポートのシステム要件を次に示します。

名前	パラメータ

NetBackup プライマリおよびメディア	■ プライマリとメディアの両方が、FIPS が有効になっている RHEL-8 システムをベースにした NetBackup 10.2.1 に配備されている必要があります。
	■ RHEL OS は RHEL8 より新しいバージョンである必要があります。
	■ 次のコマンドを使用して、RedHat マシンのバージョンを確認できます。
	`cat /etc/Redhat-release`
	■ 次のコマンドを使用して、基盤となるシステムで FIPS が有効になっているかどうかを確認できます。
	`FIPS-MODE-SETUP--CHECK`
	■ 詳しくは、次のコマンドのマニュアルページのエントリを確認してください。
	`fips-mode-setup`

Kubernetes クラスタ	■ Kubernetes クラスタは、FIPS 対応モードで配備する必要があります。 ■ FIPS モードで Kubernetes クラスタを配備するプロセスはベンダーによって異なります。 ■ たとえば、FIPS 対応の Openshift を配備します

構成パラメータ

NetBackup Kubernetes 作業負荷の FIPS モードの構成パラメータを次に示します。

構成	パラメータ
NetBackup プライマリおよびメディア	NetBackup プロセスを FIPS モードで実行できるようにします。 ■ 次のキーで <Netbackup-Installation-Path>/netbackup/bp.conf を更新します。 *NB_FIPS_MODE = ENABLE* ■ FIPS モードの NetBackup について詳しくは、『NetBackup™ セキュリティおよび暗号化ガイド』の「NetBackup ドメインでの FIPS モードの構成」セクションを参照してください。

構成	パラメータ
NetBackup Kubernetes Operator	FIPS モードを有効にするには、次のいずれかを実行します。 ■ Helm Chart から values.yaml ファイルでパラメータ fipsMode の値を ENABLE に更新します。 ■ バックアップオペレータでパラメータ NB_FIPS_MODE の値を ENABLE に更新します。

メモ: NetBackup Kubernetes 作業負荷が実行されているすべてのシステムが FIPS 準拠であることを確認します。

FIPS のトラブルシューティング

AIR (自動イメージレプリケーション) 操作への影響:

- FIPS 対応環境の AIR では、追加の構成を行う必要があります。
- サポートサイトの <KB-Article> を更新してください。
- CLI (コマンドラインインターフェース) で次のコマンドを実行します。

```
/usr/openv/java/jre/bin/keytool/keytool -storetype BCFKS
-providerpath
/usr/openv/wmc/webserver/lib/ccj-3.0.1.jar -providerclass
com.safelogic.cryptocomply.jcajce.provider.CryptoComplyFipsProvider
-importcert -trustcacerts -file <target CA certificate file (pem
encoded)> -keystore
NB_INSTALL_DIR/var/global/wsl/credentials/cacerts.bcfks -storepass
<password from the /usr/openv/var/global/jkskey file>
-alias <alias name of the trusted certificate entry to be added>
```

11

Kubernetes の問題のトラブルシューティング

この章では以下の項目について説明しています。

- NetBackup プライマリサーバー、メディアサーバー、Kubernetes サーバー間の接続性チェック

プライマリサーバーのアップグレード時のエラー: NBCheck が失敗する

NetBackup プライマリサーバーのバージョン 9.1 から 10.0 へのアップグレードが失敗し、重要でない NBCheck エラーが発生します。

エラーメッセージ: テストにより、{{ポリシーの数}} 個の有効な Kubernetes ポリシーが見つかりました。NetBackup インスタンスに有効な Kubernetes ポリシーがある場合、このテストは失敗します。(The test found {{no. of policies}} active Kubernetes policy. This test fails if the NetBackup instance has any active Kubernetes policies.)

推奨処置: NetBackup をバージョン 10.0 にアップグレードする前に、プライマリサーバーで有効な Kubernetes ポリシーをすべて無効にします。

詳しくは、https://www.veritas.com/content/support を参照してください。

古いイメージのリストア時のエラー: 操作が失敗する

NetBackup 9.1 バージョンを使用して作成された古いイメージでは、Kubernetes のリストア操作が失敗します。

エラーメッセージ: 10.0 より前のバージョンの NetBackup のバックアップイメージでは、リストア操作はサポートされません。(Restore operation is not supported on the backup images of NetBackup older than 10.0 version.)

推奨処置: Velero コマンドを使用して古いイメージをリストアします。Velero は、安全にバックアップとリストアを行い、ディザスタリカバリを実行し、Kubernetes クラスタリソースと永続ボリュームを移行するためのオープンソースツールです。そのため、Velero から古いイメージをリストアするには、インストールがクラスタでの前提条件です。

NetBackup 管理者の Web UI からバックアップ名またはバックアップ ID を取得し、それを Velero コマンドで使用してリストアします。

詳しくは、https://www.veritas.com/content/support を参照してください。

永続ボリュームのリカバリ API でのエラー

NetBackup Kubernetes Operator バージョン 10.0 では、永続ボリュームのリカバリ API が削除され、サポートされません。NetBackup の古いバージョンでは、永続ボリュームのリストアにこの API が使用されていました。そのため、NetBackup 10.0 バージョンにアップグレードした場合、永続ボリュームのリカバリ API を使用してリストアすると、リストア操作は失敗します。

エラーメッセージ: NetBackup の Kubernetes リカバリ処理の再設計に伴い、Kubernetes 永続ボリュームのリカバリ API は使用できなくなり、製品から削除されました。(Kubernetes persistent volume recovery API is no longer in use and has been removed from the product due to redesign at NetBackup Kubernetes recovery process.)

推奨処置: NetBackup Kubernetes Operator バージョン 10.0 では、選択したリソースをバックアップからリカバリするように NetBackup がアップグレードされています。したがって、永続ボリュームまたは永続ボリューム要求をリカバリする場合は、NetBackup から永続ボリュームを選択し、宛先名前空間にリカバリできます。

詳しくは、https://www.veritas.com/content/support を参照してください。

リストア中のエラー: ジョブの最終状態で一部が失敗していると表示される

リストアジョブの最終状態で一部が失敗しており、リソース RoleBinding に固有の警告がいくつか表示されます。

表示される警告は、API グループ `groupauthorization.openshift.io` と `rbac.authorization.Kubernetes.io` のリソース RoleBinding に固有です。RoleBinding は、コントローラを使用して自動管理され、新しい名前空間を作成するときに作成されるためです。

推奨処置: 関連する RoleBinding リソースをリストアから除外するか、作成された警告を無視できます。

同じ名前空間でのリストア時のエラー

選択した PVC が名前空間にすでに存在する場合、元の名前空間に PVC をリストアすると失敗することがあります。

推奨処置:

- 代替名前空間のリストアを使用できます。
- [リカバリオプション (Recovery option)]で、リストア操作の実行中に既存の PVC と重複していない PVC を選択できます。

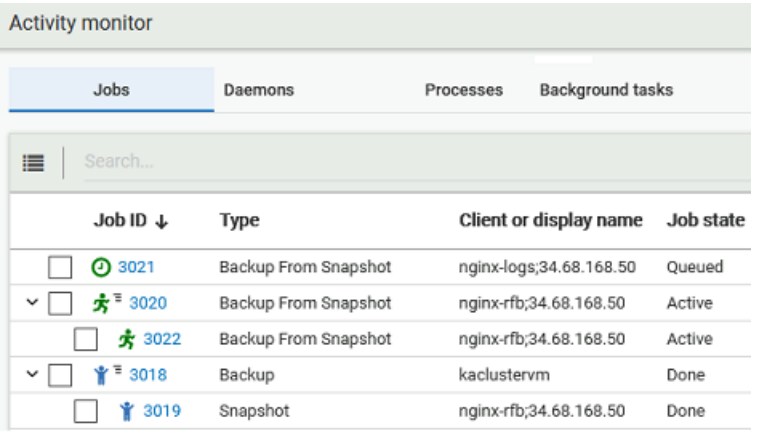
datamover ポッドが Kubernetes のリソース制限を超過

NetBackup は、2 つのリソース制限プロパティを使用して、Kubernetes 作業負荷における実行中のバックアップジョブの合計数を制御します。NetBackup バージョン 10.0 で

は、datamover ポッドが Kubernetes クラスタごとに設定されたリソース制限[バックアップ]と [スナップショットからのバックアップ]を超過します。

リソース制限の問題の例

例 1



Kubernetes クラスタあたりのスナップショットからのバックアップジョブのリソース制限は 1 に設定されています。

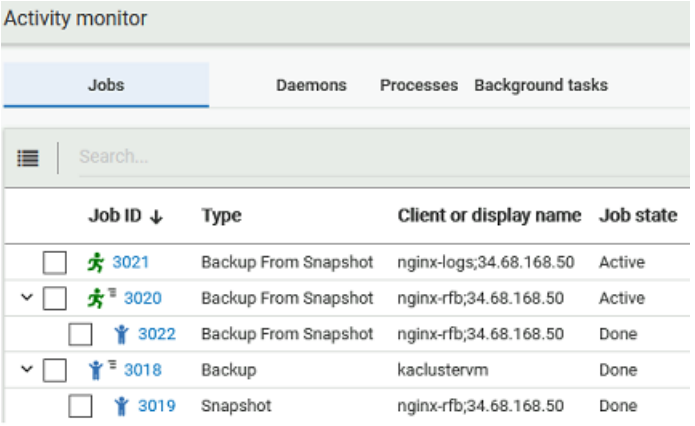
ジョブ ID 3020 と 3021 は、スナップショットからのバックアップの親ジョブです。datamover ポッドとそのクリーンアッププロセスの作成は、バックアップジョブのライフサイクルに含まれています。

ジョブ ID 3022 は子ジョブで、クラスタからストレージユニットへのデータ移動が行われます。

リソース制限の設定に基づき、ジョブ ID 3022 は実行状態であるのに対し、ジョブ ID 3021 はキューに投入された状態のままになります。バックアップジョブ ID 3022 が完了すると、親ジョブ ID 3021 が開始されます。

datamover ポッドをクリーンアップし、親ジョブ ID 3020 のライフサイクルを完了するプロセスを進めているため、ジョブ ID 3020 がまだ進行中であることに注意してください。

例 2



この段階で、NetBackup Kubernetes Operator が配備されている名前空間で 2 つの datamover ポッドが同時に実行されている場合があります。ジョブ ID 3020 の一環として作成された datamover ポッドはまだクリーンアップされていませんが、ジョブ 3021 の datamover ポッドの作成を開始しました。

スナップショットからのバックアップジョブが複数トリガされるビジー状態の環境では、リソース制限値を低く設定すると、バックアップジョブはほとんどの時間、キューに投入された状態になる可能性があります。

ただし、リソース制限の設定を高くすると、datamover ポッドがリソース制限で指定された数を超える場合があります。これにより、Kubernetes クラスタでリソースが不足する可能性があります。

3022 などのデータ移動ジョブは並行して実行されますが、クリーンアップアクティビティは順次処理されます。これは、datamover リソースのクリーンアップにかかる時間と組み合わせたときに、PVC または名前空間データのバックアップにかかる時間に近づくと、ジョブの完了がさらに遅延することになります。

データ移動とリソースのクリーンアップの合計時間がバックアップジョブと同じ場合、その後、永続ボリュームまたは名前空間データのバックアップジョブによって、ジョブの完了が遅れる可能性があります。

推奨処置: システムのリソースとパフォーマンスを確認し、それに応じてリソース制限値を設定します。この測定は、すべてのバックアップジョブで最高のパフォーマンスを実現するのに役立ちます。

リストア時のエラー: 高負荷のクラスタでジョブが失敗する

高負荷の Kubernetes クラスタではリストアジョブが失敗します。

エラーメッセージ: ERR - VxMS を初期化できません。ソケットにデータを書き込めません。peer により接続がリセットされました。(ERR - Unable to initialize VxMS, cannot write data to socket, Connection reset by peer.)

クライアント cluster.sample.domain.com のエラー bpbrm (pid=712755): ERR - VxMS を初期化できません。(Error bpbrm (pid=712755) from client cluster.sample.domain.com: ERR - Unable to initialize VxMS.)

エラー bptm (pid=712795) ソケットにデータを書き込めません。peer により接続がリセットされました。(Error bptm (pid=712795) cannot write data to socket, Connection reset by peer.)

推奨処置:リストア操作中にこの問題が発生した場合は、リストア操作を負荷の小さいクラスタで実行するか、クラスタがアイドル状態のときに実行する必要があります。

特定のクラスタ用に作成されたカスタムの Kubernetes の役割でジョブを表示できない

特定の Kubernetes クラスタで Kubernetes 作業負荷用にカスタムの RBAC の役割が作成されたら、システム管理者は、Kubernetes ジョブを表示する権限を明示的に付与する必要があります。そうしないと、Kubernetes 固有のジョブはすべて、表示されません。

システム管理者が Kubernetes ジョブを表示する権限を付与しない場合、ユーザーが表示できるジョブは次のとおりです。

- 階層表示のリストアジョブのみ。
- 一覧表示のスナップショットジョブとリストアジョブのみ。

作成されたカスタムベースの Kubernetes の役割で特定の Kubernetes クラスタのジョブを表示できない場合、次の手順を実行して表示権限を付与します。

表示権限を付与するには

1. 左側で[作業負荷 (Workload)]で[Kubernetes]をクリックします。
2. 右側で[Kubernetes 設定 (Kubernetes setting)]、[権限を管理 (Manage permissions)]の順にクリックします。
3. 対応する役割の横にある縦型の省略記号をクリックし、[編集 (Edit)]を選択します。
4. [権限の編集 (Edit permissions)]で、役割の[編集 (Edit)]権限と[ジョブの表示 (View jobs)]権限を選択し、[保存 (Save)]をクリックします。

   Kubernetes のカスタム役割のユーザーは、階層表示と一覧表示の両方で、バックアップジョブ、スナップショットジョブ、リストアジョブ、スナップショットからのバックアップジョブを表示できるようになります。

想定:

- 設定がアップグレードされると、ユーザーは次のものを表示できます。
  - 階層表示にある、既存のジョブからのリストアジョブのみ。
  - 一覧表示にある、既存のジョブからのスナップショットジョブとリストアジョブのみ。
- 選択した Kubernetes クラスタに対する権限を指定して Kubernetes のカスタム役割が作成されると、ユーザーはスナップショットジョブのみで操作をキャンセルおよび再開できます。

OperatorHub からインストールされたアプリケーションのリストア時に、選択されていない空の PVC が Openshift によって作成される

アプリケーションが OperatorHub カタログソースを介してインストールされる Openshift 環境では、ユーザーが、そのようなアプリケーション名前空間のバックアップから選択した PVC のリストアを実行しようとした場合、代わりにすべての PVC が作成されます。

この問題は、Openshift 環境では、リストア先の名前空間に必要なサイズで、選択されていない PVC がプロビジョニングされるために発生します。

メモ: このようなアプリケーションでは、ユーザーがリストア対象として PVC を選択しなくても、配備の構成に従って PVC は自動プロビジョニングされます。

Kubernetes ノードで PID の制限を超えると NetBackup Kubernetes Operator が応答しなくなる

Linux システムでは、ゾンビプロセスを消去するために PID 1 として実行される initd またはシステムプロセスが存在します。そのような initd プロセスを持たないコンテナでは、ゾンビプロセスが生成され続けます。

一定の期間が経過すると、このようなゾンビプロセスが蓄積され、Kubernetes ノードで設定されている PID の上限に達します。

NetBackup Kubernetes Operator では、nbcertcmdtool が証明書関連の操作を実行するために子プロセスを生成します。操作が完了すると、プロセスは孤立し、消去されません。最終的に PID の上限に達し、NetBackup Kubernetes Operator は応答しなくなります。

Error message: login pod/nbukops-controller-manager-67f5498bbb-gn9zw -c netbackupkops -n nbukops ERRO[0005] exec failed: container_linux.go:380: starting container process caused: read init-p: connection reset by peer a command that is terminated with exit code 1.

推奨処置:

- PID の上限の超過の問題を解決するには、Initd スクリプトを使用します。Initd スクリプトは、コントローラポッドの親プロセスまたはエントリポイントスクリプトとして機能します。
  親プロセスは、子プロセスの完了後にゾンビプロセスを自分自身に接続し、永続的なゾンビプロセスを終了させます。また、コンテナを正常にシャットダウンするためにも役立ちます。Initd スクリプトは、NBUKOPs ビルドバージョン 10.0.1 で利用可能です。
- 既存の nbcertcmdtool ゾンビプロセスを削除するには、次の手順を実行します。

1. NetBackup オペレータポッドを記述し、コントローラポッドが実行されている Kubernetes ノードを見つけます。次のコマンドを実行します。

   ```
   kubectl describe -c netbackupkops <NB k8s operator pod name> -n
   <namespace>
   ```

2. 次のコマンドを実行し、Kubernetes ノードにログオンします。

   ```
   kubectl debug node/nodename
   ```

3. 次のコマンドを実行し、nbcertcmdtool ゾンビプロセスを終了します。

   ```
   ps -ef | grep "¥[nbcertcmdtool¥] <defunct>" | awk '{print $3}' |
   xargs kill -9
   ```

メモ: これらの手順により、そのワーカーノードのすべてのゾンビプロセスが終了します。ただし、この問題の解決は一時的です。永続的な解決策としては、Initd スクリプトを使用して新しい KOps ビルドを配備する必要があります。

NetBackup Kubernetes 10.1 におけるクラスタの編集中のエラー

Kubernetes クラスタの編集操作には問題があり、NetBackup Kubernetes 10.1 バージョンでは動作しません。

対処方法: クラスタを編集するには、まず保護計画から Kubernetes クラスタを削除してから、クラスタを再び追加する必要があります。

大きいサイズの PVC でスナップショットからのリストアが失敗する

大きいサイズの PVC で、設定されたポーリングタイムアウト時間内に PVC がバインドされないと、スナップショットからのバックアップと、スナップショットまたはバックアップからの

リストアが失敗します。この問題は、大容量のボリュームスナップショットのハイドレートにデフォルトの 15 分のタイムアウトより長い時間がかかるために発生します。

サイズが大きい PVC (例: 100 GB) でスナップショットからのリストアがエラーコード 5 で失敗する

Error messages: nbcs (pid=29228) timeout occurred while waiting for the persistent volume claim pvc-sample status to be in the bound phase

対処方法: バックアップオペレータ configmap でポーリングタイムアウトを大きくします。

- ConfigMap 名: <kops-name>-backup-operator-configuration
- 更新するキー: pollingTimeoutInMinutes

名前空間ファイルモードの PVC を別のファイルシステムにリストアすると部分的に失敗する

名前空間ファイルモードの PVC を別のファイルシステムにリストアすると、名前空間ボリュームが部分的に成功します。この場合、ソースファイルシステム以外のファイルシステムにファイルシステムオブジェクト (ファイルまたはディレクトリ) をリストアすると、互換性のないメタデータのリストアに失敗します。その結果、この操作は部分的に成功したリストアとして表示されます。

Error message: 7:38:57 AM - Error bpbrm (pid=30171) client restore EXIT STATUS 1: the requested operation was partially successful.

対処方法: 宛先ファイルシステムを確認し、ファイルが配置されていることを確認します。ファイルがリストアされるとき、実際にはデータに問題はありません。この部分的なエラーは、メタデータのリストアに問題があり、オペレータがそのことを認識する必要があるという助言として報告されます。

バックアップコピーからのリストアがイメージの不整合エラーで失敗する

古いバージョンのメディアサーバーをストレージに使用すると、バックアップコピーからのリストアがイメージの不整合エラーで失敗します。たとえば、ストレージに 10.1.1 より古いバージョンのメディアサーバー、バックアップコピーからのリストアに NetBackup バージョン 10.1.1 が使用されている場合などが該当します。

Error message: Sep 22, 2022 3:12:55 PM - Info tar (pid=1459) done. status: 229: events out of sequence - image inconsistency Sep 22, 2022 3:12:55 PM - Error bpbrm (pid=16523) client restore EXIT STATUS 229: events out of sequence - image inconsistency

対処方法: すべての Kubernetes ワークフローで、Kubernetes ファイルシステムベースのバックアップには、常にプライマリ、メディア、NetBackup Kubernetes Operator バージョン 10.1.1 を使用する必要があります。

NetBackup プライマリサーバー、メディアサーバー、Kubernetes サーバー間の接続性チェック

NetBackup プライマリホストと他のホスト間の接続を確認するには、次のコマンドを参照できます。

- NetBackup プライマリサーバーとの通信を容易にするために必要なポートを開いたら、次のコマンドを実行できます。
- NetBackup プライマリサーバー/メディアサーバーと Kubernetes クラスタ間の接続を確認するには、Kubernetes Operator のポッドから次のコマンドを実行します。

  ```
  curl -v telnet://<netbackup-server-host>:<port-no>
  ```

- NetBackup プライマリサーバーと Kubernetes クラスタ間の接続を確認するには、NetBackup プライマリホストから次のコマンドを実行します。

  ```
  curl -v telnet://<kubernetes-api-server-host>:<port-no>
  ```

メモ: これらの両方のコマンドに対する応答に、接続が正常に確立されたことが示されている必要があります。
