Astra Control Center 23.07 文件

Astra Control Center

NetApp
November 27, 2023

This PDF was generated from https://docs.netapp.com/zh-tw/astra-control-center-2307/index.html on November 27, 2023. Always check docs.netapp.com for the latest.

目錄

Astra Control Center 23.07 文件

版本資訊

我們很高興在此發表最新版的Astra Control Center。

- "本版Astra Control Center內容"
- "已知問題"
- "已知限制"

請透過成為來傳送有關文件的意見反應 "GitHub貢獻者" 或傳送電子郵件至doccomments@netapp.com。

Astra Control Center版本的新功能

我們很高興在此發表最新版的Astra Control Center。

2023 年 7 月 31 日（ 23.07.0 ）

新功能與支援

- "支援在擴充組態中使用 NetApp MetroCluster 做為儲存後端"
- "支援使用 Longhorn 做為儲存後端"
- "應用程式現在可以從同一個 Kubernetes 叢集在 ONTAP 後端之間複寫"
- "Astra Control Center 現在支援「 userPrincipalName 」做為遠端（ LDAP ）使用者的替代登入屬性"
- "使用 Astra Control Center 進行複寫容錯移轉後、可以執行新的執行掛鉤類型「容錯移轉後」"
- Clone 工作流程現在僅支援即時複製（託管應用程式的目前狀態）。若要從快照或備份複製、請使用 "還原工作流程"。

已知問題與限制

- "此版本的已知問題"
- "此版本的已知限制"

2023 年 5 月 18 日（ 23.04.2 ）

詳細資料

此適用於 Astra Control Center （ 23.04.0 ）的修補程式版本（ 23.04.2 ）可提供支援 "Kubernetes CSI 外部快照器 v6.1.0" 並修正下列問題：

- 使用執行掛鉤時、就地應用程式還原的錯誤
- 貯體服務的連線問題

2023 年 4 月 25 日（ 23.04.0 ）

詳細資料

新功能與支援

- "根據預設、新 Astra Control Center 安裝會啟用 90 天試用版授權"
- "更強大的執行掛勾功能、提供更多篩選選項"
- "現在可以在使用 Astra Control Center 進行複寫容錯移轉後執行執行攔截程式"
- "支援將 Volume 從「 ONTAP NAS 經濟型儲存」等級移轉至「 ONTAP NAS 」儲存等級"
- "支援在還原作業期間包含或排除應用程式資源"
- "支援管理純資料應用程式"

已知問題與限制

- "此版本的已知問題"
- "此版本的已知限制"

2022年11月22日（22.11.0）

詳細資料

新功能與支援

- "支援橫跨多個命名空間的應用程式"
- "支援將叢集資源納入應用程式定義"
- "透過角色型存取控制（RBAC）整合、強化LDAP驗證"
- "新增對Kubernetes 1.25和Pod安全許可（PSA）的支援"
- "增強備份、還原及複製作業的進度報告功能"

已知問題與限制

- "此版本的已知問題"
- "此版本的已知限制"

2022年9月8日（22.08.1）

詳細資料

此適用於Astra Control Center（22.08.0）的修補程式版本（22.08.1）可利用NetApp SnapMirror修正應用程式複寫中的小錯誤。

2022年8月10日（22.08.0）

詳細資料

新功能與支援

- "使用NetApp SnapMirror技術進行應用程式複寫"
- "改善應用程式管理工作流程"
- "增強的執行掛勾功能、讓您自行執行"

> NetApp針對特定應用程式提供的預設快照前及後執行掛勾已在此版本中移除。如果您升級至此版本、但未提供您專屬的快照執行掛勾、Astra Control將僅擷取損毀一致的快照。請造訪 "NetApp Verda" GitHub儲存庫提供範例執行攔截指令碼、您可以根據環境進行修改。

- "支援VMware Tanzu Kubernetes Grid整合版（TKGI）"
- "支援Google Anthos"
- "LDAP組態（透過Astra Control API）"

已知問題與限制

- "此版本的已知問題"
- "此版本的已知限制"

2022年4月26日（22.04.0）

詳細資料

新功能與支援

- "命名空間角色型存取控制（RBAC）"
- "支援Cloud Volumes ONTAP 功能"
- "Astra Control Center的一般入侵能力"
- "從Astra Control移除鏟斗"
- "支援VMware Tanzu產品組合"

已知問題與限制

- "此版本的已知問題"
- "此版本的已知限制"

2021年12月14日（21.12）

詳細資料

新功能與支援

- "應用程式還原"
- "執行掛勾"
- "支援以命名空間範圍運算子部署的應用程式"
- "支援上游Kubernetes和Rancher"
- "Astra Control Center升級"
- "Red Hat作業系統集線器選項"

已解決的問題

- "已解決此版本的問題"

已知問題與限制

- "此版本的已知問題"
- "此版本的已知限制"

2021年8月5日（21.08）

詳細資料

Astra Control Center正式推出。

- "它是什麼"
- "瞭解架構與元件"
- "開始使用所需的一切"
- "安裝" 和 "設定"
- "管理" 和 "保護" 應用程式
- "管理儲存庫" 和 "儲存後端"
- "管理帳戶"
- "利用API自動化"

如需詳細資訊、請參閱

- "此版本的已知問題"
- "此版本的已知限制"
- "舊版Astra Control Center文件"

已知問題

已知問題可識別可能導致您無法成功使用本產品版本的問題。

下列已知問題會影響目前的版本：

- 如果在管理叢集之後新增volumesnapshotClass、則應用程式備份和快照將會失敗
- [應用程式複製在以設定的儲存類別部署應用程式之後失敗]
- 使用 Astra Control Center 管理叢集時、如果 Kribeconfig 檔案包含多個內容、就會失敗
- 監控 Pod 可能會在 Istio 環境中當機
- 當Astra Trident離線時、應用程式資料管理作業會因內部服務錯誤（500）而失敗

如果在管理叢集之後新增**volumesnapshotClass**、則應用程式備份和快照將會失敗

備份與快照無法使用 `UI 500 error` 在此案例中。因應措施是重新整理應用程式清單。

應用程式複製在以設定的儲存類別部署應用程式之後失敗

在部署應用程式並明確設定儲存類別之後（例如、 `helm install …-set global.storageClass=netapp-cvs-perf-extreme`）之後、若想要複製應用程式、則目標叢集必須擁有原本指定的儲存類別。
將具有明確設定儲存類別的應用程式複製到沒有相同儲存類別的叢集、將會失敗。在此案例中沒有任何恢復步驟。

使用 **Astra Control Center** 管理叢集時、如果 **Kribeconfig** 檔案包含多個內容、就會失敗

您無法在其中使用多個叢集和內容的Kbeconfig。請參閱 "知識庫文章" 以取得更多資訊。

監控 **Pod** 可能會在 **Istio** 環境中當機

如果您在 Istio 環境中將 Astra Control Center 與 Cloud Insights 配對 `telegraf-rs` Pod 可能當機。因應措施是執行下列步驟：

1. 尋找當機的 Pod ：

   ```
   kubectl -n netapp-monitoring get pod | grep Error
   ```

   您應該會看到類似下列的輸出：

   ```
   NAME READY STATUS RESTARTS AGE
   telegraf-rs-fhhrh 1/2 Error 2 (26s ago) 32s
   ```

2. 重新啟動當機的 Pod 、更換 `<pod_name_from_output>` 使用受影響 Pod 的名稱：

```
kubectl -n netapp-monitoring delete pod <pod_name_from_output>
```

您應該會看到類似下列的輸出：

```
pod "telegraf-rs-fhhrh" deleted
```

3. 確認 Pod 已重新啟動、且未處於錯誤狀態：

```
kubectl -n netapp-monitoring get pod
```

您應該會看到類似下列的輸出：

```
NAME READY STATUS RESTARTS AGE
telegraf-rs-rrnsb 2/2 Running 0 11s
```

當Astra Trident離線時、應用程式資料管理作業會因內部服務錯誤（500）而失敗

如果應用程式叢集上的Astra Trident離線（並重新連線）、而且在嘗試應用程式資料管理時遇到500個內部服務錯誤、請重新啟動應用程式叢集中的所有Kubernetes節點、以還原功能。

如需詳細資訊、請參閱

- "已知限制"

已知限制

已知限制指出本產品版本不支援的平台、裝置或功能、或是無法與產品正確互通的平台、裝置或功能。請仔細檢閱這些限制。

叢集管理限制

- 同一個叢集無法由兩個Astra Control Center執行個體管理
- Astra Control Center無法管理兩個名稱相同的叢集

角色型存取控制（RBAC）限制

- 具有命名空間RBAC限制的使用者可以新增及取消管理叢集
- [具有命名空間限制的成員必須先將命名空間新增至限制、才能存取複製或還原的應用程式]

應用程式管理限制

- [單一命名空間中的多個應用程式無法一起還原至不同的命名空間]
- Astra Control 不支援每個命名空間使用多個儲存類別的應用程式

- Astra Control不會自動指派雲端執行個體的預設值區段
- [使用傳遞參考運算子安裝的應用程式複製可能會失敗]
- [不支援使用憑證管理程式之應用程式的就地還原作業]
- 不支援啟用OLM且叢集範圍內的營運者部署應用程式
- 不支援以Helm 2部署的應用程式
- 具有特定快照控制器版本的 Kubernetes 1.25 或更新版本叢集快照可能會失敗
- 在移除Astra Control Center執行個體期間、可能無法保留備份與快照

一般限制

- LDAP使用者和群組限制
- Astra Control Center中的S3鏟斗未報告可用容量
- Astra Control Center不會驗證您為Proxy伺服器輸入的詳細資料
- 現有連線至Postgres Pod會導致故障
- <<「活動」頁面最多可顯示 100000 個事件>>

同一個叢集無法由兩個Astra Control Center執行個體管理

如果您想要管理另一個Astra Control Center執行個體上的叢集、您應該先進行 "取消管理叢集" 在另一個執行個體上進行管理之前、請先從管理該執行個體的執行個體進行管理。從管理中移除叢集之後、請執行下列命令、確認叢集未受管理：

```
oc get pods n -netapp-monitoring
```

該命名空間中不應有執行的Pod、或命名空間不應存在。如果其中任一項為真、則叢集不受管理。

Astra Control Center無法管理兩個名稱相同的叢集

如果您嘗試新增的叢集名稱與已存在的叢集名稱相同、則作業將會失敗。如果您尚未變更Kubernetes組態檔中的叢集名稱預設值、則此問題最常發生在標準Kubernetes環境中。

因應措施如下：

1. 編輯您的 `kubeadm-config` 組態對應：

   ```
   kubectl edit configmaps -n kube-system kubeadm-config
   ```

2. 變更 `clusterName` 欄位值來源 `kubernetes` （Kubernetes預設名稱）至唯一的自訂名稱。
3. 編輯Kbeconfig (`.kube/config`) 。
4. 從更新叢集名稱 `kubernetes` 唯一的自訂名稱 (`xyz-cluster` 的範例中使用）。同時進行更新 `clusters` 和 `contexts` 本範例所示的章節：

```
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data:
ExAmPLERb2tCcjZ5K3E2Njk4eQotLExAMpLEORCBDRVJUSUZJQ0FURS0txxxxXX==
    server: https://x.x.x.x:6443
  name: xyz-cluster
contexts:
- context:
    cluster: xyz-cluster
    namespace: default
    user: kubernetes-admin
  name: kubernetes-admin@kubernetes
current-context: kubernetes-admin@kubernetes
```

具有命名空間RBAC限制的使用者可以新增及取消管理叢集

不應允許具有命名空間RBAC限制的使用者新增或取消管理叢集。由於目前的限制、Astra無法防止此類使用者取消管理叢集。

具有命名空間限制的成員必須先將命名空間新增至限制、才能存取複製或還原的應用程式

任何 `member` 具有命名空間名稱/ID之RBAC限制的使用者、可以將應用程式複製或還原至同一叢集上的新命名空間、或是組織帳戶中的任何其他叢集。不過、相同的使用者無法存取新命名空間中的複製或還原應用程式。在複製或還原作業建立新命名空間之後、帳戶管理員/擁有者即可編輯 `member` 使用者帳戶和更新角色限制、讓受影響的使用者能夠授予新命名空間的存取權。

單一命名空間中的多個應用程式無法一起還原至不同的命名空間

如果您在單一命名空間中管理多個應用程式（在Astra Control中建立多個應用程式定義）、則無法將所有應用程式還原至不同的單一命名空間。您需要將每個應用程式還原至各自獨立的命名空間。

Astra Control 不支援每個命名空間使用多個儲存類別的應用程式

Astra Control 支援每個命名空間使用單一儲存類別的應用程式。當您將應用程式新增至命名空間時、請確定該應用程式與命名空間中的其他應用程式具有相同的儲存類別。

Astra Control不會自動指派雲端執行個體的預設值區段

Astra Control不會自動指派任何雲端執行個體的預設儲存區。您需要手動設定雲端執行個體的預設儲存區。如果未設定預設儲存區、您將無法在兩個叢集之間執行應用程式複製作業。

使用傳遞參考運算子安裝的應用程式複製可能會失敗

Astra Control支援以命名空間範圍運算子安裝的應用程式。這些運算子通常採用「傳遞值」而非「傳遞參照」架構來設計。以下是一些遵循這些模式的營運者應用程式：

- "Apache K8ssandra"

> K8ssandra 支援原位還原作業。若要還原新命名空間或叢集的作業、必須先關閉應用程式的原始執行個體。這是為了確保傳遞的對等群組資訊不會導致跨執行個體通訊。不支援複製應用程式。

- "Jenkins CI"
- "Percona XtraDB叢集"

Astra Control可能無法複製以「傳遞參考」架構設計的操作員（例如CockroachDB操作員）。在這些類型的複製作業中、複製的操作員會嘗試從來源操作員參考Kubernetes機密、儘管在複製程序中有自己的新秘密。由於Astra Control不知道來源營運者的Kubernetes機密資料、因此複製作業可能會失敗。

> 在複製作業期間、需要IngresClass資源或Webhooks才能正常運作的應用程式、不得在目的地叢集上定義這些資源。

不支援使用憑證管理程式之應用程式的就地還原作業

本版Astra Control Center不支援與憑證管理員就地還原應用程式。支援將作業還原至不同的命名空間和複製作業。

不支援啟用**OLM**且叢集範圍內的營運者部署應用程式

Astra Control Center不支援使用叢集範圍的運算子進行應用程式管理活動。

不支援以**Helm 2**部署的應用程式

如果您使用Helm來部署應用程式、Astra Control Center需要Helm版本3。完全支援使用Helm 3部署的應用程式管理及複製（或從Helm 2升級至Helm 3）。如需詳細資訊、請參閱 "Astra Control Center需求"。

具有特定快照控制器版本的 **Kubernetes 1.25** 或更新版本叢集快照可能會失敗

如果叢集上安裝 Snapshot 控制器 API 的 v1beta1 版、執行 1.25 版或更新版本的 Kubernetes 叢集快照可能會失敗。

因應措施是在升級現有 Kubernetes 1.25 或更新版本的安裝時執行下列動作：

1. 移除任何現有的 Snapshot CRD 和任何現有的 Snapshot 控制器。
2. "解除安裝Astra Trident"。
3. "安裝 Snapshot CRD 和 Snapshot 控制器"。
4. "安裝最新的 Astra Trident 版本"。
5. "建立 Volume SnapshotClass"。

在移除**Astra Control Center**執行個體期間、可能無法保留備份與快照

如果您擁有評估授權、請務必儲存您的帳戶ID、以免在Astra Control Center故障時發生資料遺失（如果您未傳送ASUP）。

LDAP使用者和群組限制

Astra Control Center支援最多5、000個遠端群組和10、000個遠端使用者。

Astra Control 不支援具有 DN 的 LDAP 實體（使用者或群組）、該 DN 包含具有結尾空格或結尾空格的 RDN 。

Astra Control Center中的S3鏟斗未報告可用容量

在備份或複製由Astra Control Center管理的應用程式之前、請先查看ONTAP 資訊庫（英文）或StorageGRID 資訊庫（英文）管理系統中的庫位資訊。

Astra Control Center不會驗證您為Proxy伺服器輸入的詳細資料

請務必做到 "輸入正確的值" 建立連線時。

現有連線至Postgres Pod會導致故障

當您在Postgres Pod上執行作業時、不應直接在Pod內連線以使用psql命令。Astra Control需要psql存取來凍結及解出資料庫。如果有預先存在的連線、則快照、備份或複製都會失敗。

「活動」頁面最多可顯示 100000 個事件

Astra Control 活動頁面最多可顯示 100,000 個事件。若要檢視所有記錄的事件、請使用擷取事件 "Astra Control API"。

如需詳細資訊、請參閱

- "已知問題"

開始使用

瞭解Astra Control

Astra Control是Kubernetes應用程式資料生命週期管理解決方案、可簡化狀態應用程式的作業。輕鬆保護、備份、複寫及移轉Kubernetes工作負載、並即時建立運作中的應用程式複本。

功能

Astra Control為Kubernetes應用程式資料生命週期管理提供關鍵功能：

- 自動管理持續儲存
- 建立應用程式感知的隨需快照與備份
- 自動化原則導向的快照與備份作業
- 將應用程式和資料從一個Kubernetes叢集移轉到另一個叢集
- 使用 NetApp SnapMirror 技術（ Astra Control Center ）將應用程式複寫到遠端系統
- 將應用程式從接移複製到正式作業
- 視覺化應用程式健全狀況與保護狀態
- 使用Web UI或API來實作備份與移轉工作流程

部署模式

Astra Control提供兩種部署模式：

- * Astra Control Service*：NetApp管理的服務、可在多個雲端供應商環境中、以及自行管理的Kubernetes叢集、提供Kubernetes叢集的應用程式感知資料管理功能。
- * Astra Control Center*：自我管理的軟體、可針對在內部部署環境中執行的Kubernetes叢集、提供應用程式感知資料管理功能。Astra Control Center 也可以安裝在多個雲端供應商環境中、搭配 NetApp Cloud Volumes ONTAP 儲存後端。

	Astra控制服務	**Astra**控制中心
如何提供？	是NetApp提供的完整託管雲端服務	可下載、安裝及管理的軟體
它的代管位置為何？	在NetApp首選的公有雲上	在您自己的Kubernetes叢集上
如何更新？	由NetApp管理	您可以管理任何更新

	Astra控制服務	Astra控制中心
支援的儲存後端有哪些？	• Amazon網路服務： ◦ Amazon EBS ◦ Amazon FSX for NetApp ONTAP 產品 ◦ "Cloud Volumes ONTAP" • Google Cloud： ◦ Google持續磁碟 ◦ NetApp Cloud Volumes Service ◦ "Cloud Volumes ONTAP" • Microsoft Azure： ◦ Azure託管磁碟 ◦ Azure NetApp Files ◦ "Cloud Volumes ONTAP" • 自我管理叢集： ◦ Amazon EBS ◦ Google持續磁碟 ◦ Azure託管磁碟 ◦ "Cloud Volumes ONTAP"	• NetApp ONTAP AFF 的功能與FAS 功能 • "Cloud Volumes ONTAP"

Astra Control Service的運作方式

Astra Control Service是NetApp託管的雲端服務、隨時可用最新功能進行更新。它利用多個元件來實現應用程式資料生命週期管理。

Astra Control Service的高層級運作方式如下：

- 您可以設定雲端供應商並註冊Astra帳戶、開始使用Astra Control Service。
  - 對於GKE叢集、Astra Control Service使用 "適用於Cloud Volumes Service Google Cloud的NetApp解決方案" 或Google持續磁碟做為持續磁碟區的儲存後端。
  - 對於高峰叢集、Astra Control Service使用 "Azure NetApp Files" 或Azure託管磁碟做為持續磁碟區的儲存後端。
  - 對於Amazon EKS叢集、Astra Control Service使用 "Amazon彈性區塊存放區" 或 "Amazon FSX for NetApp ONTAP 產品" 作為持續磁碟區的儲存後端。
- 您將第一部Kubernetes運算新增至Astra Control Service。Astra Control Service接著會執行下列作業：
  - 在雲端供應商帳戶中建立物件存放區、以儲存備份複本。

    在Azure中、Astra Control Service也會為Blob容器建立資源群組、儲存帳戶和金鑰。
  - 在叢集上建立新的管理員角色和Kubernetes服務帳戶。

- 使用新的管理員角色進行安裝 "Astra Trident" 在叢集上建立一個或多個儲存類別。
- 如果您使用NetApp雲端服務儲存產品做為儲存後端、Astra Control Service會使用Astra Trident來為應用程式配置持續的磁碟區。如果您使用Amazon EBS或Azure託管磁碟做為儲存後端、則需要安裝供應商專屬的SCSI驅動程式。安裝說明請參閱 "設定Amazon Web Services" 和 "使用Azure託管磁碟來設定Microsoft Azure"。

- 此時、您可以將應用程式新增至叢集。將在新的預設儲存類別上配置持續磁碟區。
- 然後使用Astra Control Service來管理這些應用程式、並開始建立快照、備份和複製。

Astra Control的免費方案可讓您管理帳戶中最多10個命名空間。如果您想要管理10多個項目、則必須將「免費方案」升級為「優質方案」、以設定帳單。

Astra控制中心的運作方式

Astra Control Center可在您自己的私有雲端本機執行。

Astra Control Center 支援 Kubernetes 叢集、搭配 Astra Trident 型儲存類別、以及 ONTAP 9.5 以上的儲存後端。

在雲端連線的環境中、Astra Control Center使用Cloud Insights 「資訊中心」來提供進階的監控和遙測功能。若缺乏Cloud Insights 支援鏈接、Astra Control Center可提供有限（7天數據）的監控與遙測功能、並透過開放式指標端點匯出至Kubernetes原生監控工具（例如Prometheus和Grafana）。

Astra Control Center已完全整合AutoSupport 至整套的功能、可Active IQ 為使用者和NetApp支援人員提供疑難排解和使用資訊。

您可以使用 90 天內嵌評估授權、試用 Astra Control Center 。在評估 Astra Control Center 時、您可以透過電子郵件和社群選項獲得支援。此外、您也可以從產品內的支援儀表板存取知識庫文章和文件。

若要安裝及使用Astra Control Center、您必須符合特定需求 "需求"。

Astra Control Center的高層級運作方式如下：

- 您可以在本機環境中安裝Astra Control Center。深入瞭解如何操作 "安裝Astra Control Center"。
- 您可以完成以下設定工作：
  - 設定授權。
  - 新增第一個叢集。
  - 新增新增叢集時發現的儲存後端。
  - 新增物件存放區儲存應用程式備份。

深入瞭解如何操作 "設定Astra控制中心"。

您可以將應用程式新增至叢集。或者、如果叢集中已有一些應用程式正在管理中、您可以使用Astra Control Center來管理這些應用程式。然後、使用Astra Control Center建立快照、備份、複製及複寫關係。

以取得更多資訊

- "Astra Control Service文件"

- "Astra Control Center文件"
- "Astra Trident文件"
- "使用Astra Control API"
- "本文檔 Cloud Insights"
- "本文檔 ONTAP"

Astra Control Center需求

開始驗證作業環境、應用程式叢集、應用程式、授權和網頁瀏覽器的整備度。確保您的環境符合這些需求、以部署和操作 Astra Control Center 。

- 支援的主機叢集 Kubernetes 環境
- [主機叢集資源需求]
- Astra Trident的需求
- [儲存設備後端]
- [映像登錄]
- Astra Control Center 授權
- 不需要授權ONTAP
- [網路需求]
- 內部部署Kubernetes叢集的入口
- [支援的網頁瀏覽器]
- [應用程式叢集的其他需求]

支援的主機叢集 **Kubernetes** 環境

Astra Control Center 已通過下列 Kubernetes 主機環境的驗證：

確保您選擇架設 Astra Control Center 的 Kubernetes 環境符合環境正式文件中所述的基本資源需求。

Kubernetes 在主機叢集上的發佈	支援的版本
Azure Stack HCI 上的 Azure Kubernetes 服務	Azure Stack HCI 21H2 和 22H2 、含 1.24.x 和 1.5.x
Google Anthos	1.14 至 1.16 （請參閱 Google Anthos 入口要求）
Kubernetes （上游）	1.25 至 1.27 （ Kubernetes 1.25 或更新版本需要 Astra Trident 22.10 或更新版本）
Rancher Kubernetes引擎（RKE）	RKE 1.3 搭配 Rancher Manager 2.6 RKE 1.4 搭配 Rancher Manager 2.7 RKE 2 （ v1.24.x ）搭配 Rancher 2.6 RKE 2 （ v1.5.x ）搭配 Rancher 2.7
Red Hat OpenShift Container Platform	4.11 至 4.13

主機叢集資源需求

除了環境的資源需求之外、Astra Control Center還需要下列資源：

> 這些需求假設Astra Control Center是營運環境中唯一執行的應用程式。如果環境正在執行其他應用程式、請相應調整這些最低需求。

- *CPU 擴充 * ：主機環境中所有節點的 CPU 都必須啟用 AVX 擴充功能。
- * 工作節點 * ：總計至少 3 個工作節點、每個節點有 4 個 CPU 核心和 12GB RAM

Astra Trident的需求

確保您符合下列特定環境需求的 Astra Trident 要求：

- * 與 Astra Control Center 搭配使用的最低版本 * ： Astra Trident 22.10 或更新版本已安裝及設定。
- * SnapMirror 複寫 * ：安裝 Astra Trident 22.10 或更新版本以進行 SnapMirror 型應用程式複寫。
- * 對於 Kubernetes 1.25 或更新版本的支援 * ：針對 Kubernetes 1.25 或更新版本叢集安裝 Astra Trident 22.10 或更新版本（升級至 Kubernetes 1.25 或更新版本之前、您必須先升級至 Astra Trident 22.10 ）
- *Astra Trident 的 ONTAP 組態 * ：
  - * 儲存類別 * ：在叢集上至少設定一個 Astra Trident 儲存類別。如果已設定預設儲存類別、請確定它是唯一具有預設指定的儲存類別。
  - * 儲存驅動程式和工作節點 * ：確保叢集中的工作節點已設定適當的儲存驅動程式、以便 Pod 與後端儲存設備互動。Astra Control Center支援ONTAP Astra Trident提供的下列支援資訊驅動程式：
    - `ontap-nas`
    - `ontap-san`
    - `ontap-san-economy` （此儲存類別類型無法使用應用程式複寫）
    - `ontap-nas-economy` （此儲存類別類型無法使用快照、複寫原則和保護原則）

儲存設備後端

確保您擁有支援的後端、且具有足夠的容量。

- * 所需的儲存後端容量 * ：可用容量至少 500 GB
- * 支援的後端 * ： Astra Control Center 支援下列儲存後端：
  - NetApp ONTAP 9.8 或更新版本的 AFF 、 FAS 和 ASA 系統
  - NetApp ONTAP Select 9.8 或更新版本
  - NetApp Cloud Volumes ONTAP 9.8 或更新版本
  - Longhorn 1.5.0 或更新版本
    - 需要手動建立 Volume SnapshotClass 物件。請參閱 "Longhorn 文件" 以取得相關指示。
  - NetApp MetroCluster
    - 託管 Kubernetes 叢集必須採用彈性組態。

不需要授權**ONTAP**

若要使用Astra Control Center、請視ONTAP 您需要完成的工作而定、確認您擁有下列各項的版次授權：

- FlexClone
- SnapMirror：選用。僅使用SnapMirror技術複寫至遠端系統時才需要。請參閱 "SnapMirror授權資訊"。
- S3授權：選用。僅適用於SS3鏟斗ONTAP

若要檢查ONTAP 您的不實系統是否有必要的授權、請參閱 "管理ONTAP 不需購買的授權"。

NetApp MetroCluster

當您使用 NetApp MetroCluster 做為儲存後端時、必須在您使用的 Astra Trident 驅動程式中、將 SVM 管理 LIF 指定為後端選項。

若要設定 MetroCluster LIF 、請參閱 Astra Trident 文件、以取得每個驅動程式的詳細資訊：

- "SAN"
- "NAS"

映像登錄

您必須擁有現有的私有 Docker 映像登錄、才能將 Astra Control Center 建置映像推送至該登錄。您需要提供映像登錄的URL、以便上傳映像。

Astra Control Center 授權

Astra Control Center 需要 Astra Control Center 授權。安裝 Astra Control Center 時、已啟動內嵌式 90 天試用版授權、可用於 4 、 800 個 CPU 單元。如果您需要更多容量或不同的評估條款、或想要升級至完整授權、您可以向 NetApp 取得不同的評估授權或完整授權。您需要授權來保護應用程式和資料。

您可以報名免費試用 Astra Control Center 。您可以註冊註冊 "請按這裡"。

若要設定授權、請參閱 "使用90天試用版授權"。

若要深入瞭解授權的運作方式、請參閱 "授權"。

網路需求

設定您的營運環境、確保 Astra Control Center 能夠正常通訊。需要下列網路組態：

- * FQDN 位址 * ：您必須擁有 Astra Control Center 的 FQDN 位址。
- * 存取網際網路 * ：您應該判斷是否有外部存取網際網路的權限。如果您沒有、部分功能可能會受到限制、例如從NetApp Cloud Insights 接收監控和數據資料、或是將支援組合傳送至 "NetApp 支援網站"。
- * 連接埠存取 * ：裝載 Astra Control Center 的作業環境使用下列 TCP 連接埠進行通訊。您應確保這些連接埠可透過任何防火牆、並設定防火牆、以允許來自Astra網路的任何HTTPS輸出流量。有些連接埠需要在裝載Astra Control Center的環境與每個託管叢集之間進行連線（視情況而定）。

您可以在雙堆疊Kubernetes叢集中部署Astra Control Center、Astra Control Center則可管理已設定為雙堆疊作業的應用程式和儲存後端。如需雙堆疊叢集需求的詳細資訊、請參閱 "Kubernetes文件"。

來源	目的地	連接埠	傳輸協定	目的
用戶端 PC	Astra控制中心	443..	HTTPS	UI / API存取：確保此連接埠在裝載Astra Control Center的叢集與每個受管理叢集之間都開啟
度量使用者	Astra Control Center工作節點	9090	HTTPS	度量資料通訊：確保每個託管叢集都能存取裝載Astra Control Center的叢集上的此連接埠（需要雙向通訊）
Astra控制中心	託管Cloud Insights版的服務 (https://www.netapp.com/cloud-services/cloud-insights/)	443..	HTTPS	通訊Cloud Insights
Astra控制中心	Amazon S3儲存貯體供應商	443..	HTTPS	Amazon S3儲存通訊
Astra控制中心	NetApp AutoSupport (https://support.netapp.com)	443..	HTTPS	NetApp AutoSupport通訊

內部部署Kubernetes叢集的入口

您可以選擇網路入侵Astra控制中心的用途類型。依預設、Astra Control Center會將Astra Control Center閘道（服務/網路）部署為整個叢集的資源。Astra Control Center也支援使用服務負載平衡器（如果環境允許）。如果您想要使用服務負載平衡器、但尚未設定一個、則可以使用MetalLB負載平衡器自動將外部IP位址指派給服務。在內部DNS伺服器組態中、您應該將Astra Control Center所選的DNS名稱指向負載平衡的IP位址。

負載平衡器應使用與Astra Control Center工作節點IP位址位於同一子網路中的IP位址。

如需詳細資訊、請參閱 "設定入口以進行負載平衡"。

Google Anthos 入口要求

在 Google Anthos 叢集上代管 Astra Control Center 時、請注意 Google Antos 預設包含 MetalLB 負載平衡器和 Istio 入口服務、讓您在安裝期間只需使用 Astra Control Center 的一般入口功能即可。請參閱 "設定Astra控制中心" 以取得詳細資料。

支援的網頁瀏覽器

Astra Control Center支援最新版本的Firefox、Safari和Chrome、最低解析度為1280 x 720。

應用程式叢集的其他需求

如果您打算使用這些Astra Control Center功能、請謹記以下要求：

- 應用程式叢集需求： "叢集管理需求"
  - 受管理的應用程式需求： "應用程式管理需求"
  - 應用程式複寫的其他需求： "複寫先決條件"

下一步

檢視 "快速入門" 總覽：

Astra Control Center快速入門

以下是使用Astra Control Center所需的步驟總覽。每個步驟中的連結都會帶您前往提供更多詳細資料的頁面。

檢閱**Kubernetes**叢集需求

確保您的環境符合下列需求：

- Kubernetes叢集*
- "確保您的主機叢集符合作業環境需求"
- "設定內部部署Kubernetes叢集的負載平衡入口"

儲存整合

- "確保您的環境包含Astra Trident支援的版本"
- "準備工作節點"
- "設定Astra Trident儲存後端"
- "設定Astra Trident儲存類別"
- "安裝Astra Trident Volume Snapshot控制器"
- "建立Volume Snapshot類別"

不包含認證資料 ONTAP

- "設定ONTAP 驗證資料"

下載並安裝**Astra Control Center**

完成下列安裝工作：

- "從 NetApp 支援網站 下載頁面下載 Astra 控制中心"
- 取得NetApp授權檔案：

- 如果您正在評估 Astra Control Center 、則已包含內嵌評估授權
- "如果您已購買Astra Control Center、請產生授權檔案"

- "安裝Astra Control Center"
- "執行其他選用的組態步驟"

3 完成一些初始設定工作

完成一些基本工作以開始：

- "新增授權"
- "為叢集管理做好準備"
- "新增叢集"
- "新增儲存後端"
- "新增儲存庫"

4 使用**Astra**控制中心

完成 Astra Control Center 設定後、請使用 Astra Control UI 或 "Astra Control API" 若要開始管理及保護應用程式：

- "管理應用程式"：定義要管理的資源。
- "保護應用程式"：設定保護原則、並複寫、複製及移轉應用程式。
- "管理帳戶"：使用者、角色、 LDAP 、認證等。
- "也可以連接Cloud Insights 到"：查看系統健全狀況的指標。

以取得更多資訊

- "使用Astra Control API"
- "升級Astra Control Center"
- "取得Astra Control的協助"

安裝總覽

選擇並完成下列其中一個Astra Control Center安裝程序：

- "使用標準程序安裝Astra Control Center"
- "（如果您使用Red Hat OpenShift）使用OpenShift作業系統集線器安裝Astra Control Center"
- "安裝Astra Control Center搭配Cloud Volumes ONTAP 一套功能性儲存後端"

視您的環境而定、安裝Astra Control Center之後可能需要額外的組態：

- "安裝後設定Astra Control Center"

使用標準程序安裝**Astra Control Center**

若要安裝Astra Control Center、請從NetApp 支援網站 下列網址下載安裝套件、並執行下列步驟。您可以使用此程序、在連線網際網路或無線環境中安裝Astra Control Center。

展開以進行其他安裝程序

- *使用RedHat OpenShift操作員中樞*安裝：請使用此功能 "替代程序" 使用作業系統集線器在OpenShift上安裝Astra Control Center。
- 以**Cloud Volumes ONTAP** 支援功能的方式在公有雲上安裝：使用 "這些程序" 若要在Amazon Web Services（AWS）、Google Cloud Platform（GCP）或Microsoft Azure中安裝Astra Control Center、並提供Cloud Volumes ONTAP 一套支援整合式儲存後端的功能。

如需Astra Control Center安裝程序的示範、請參閱 "這段影片"。

開始之前

- "開始安裝之前、請先準備好環境以進行Astra Control Center部署"。
- 如果您已設定或想要在環境中設定Pod安全性原則、請熟悉Pod安全性原則、以及這些原則如何影響Astra Control Center安裝。請參閱 "Pod 安全限制"。
- 確保所有API服務均處於健全狀態且可供使用：

```
kubectl get apiservices
```

- 確保您打算使用的Astra FQDN可路由傳送至此叢集。這表示您在內部DNS伺服器中有DNS項目、或是使用已註冊的核心URL路由。
- 如果叢集中已存在憑證管理程式、您需要執行某些作業 "必要步驟" 因此Astra Control Center不會嘗試安裝自己的憑證管理程式。依預設、Astra Control Center會在安裝期間安裝自己的憑證管理程式。

在第三個故障網域或次要站台中部署 Astra Control Center 。這是應用程式複寫和無縫災難恢復的建議。

步驟

若要安裝Astra Control Center、請執行下列步驟：

- 下載並擷取Astra Control Center
- 安裝NetApp Astra kubecl外掛程式
- [將映像新增至本機登錄]
- [設定具有驗證需求之登錄的命名空間和機密]
- 安裝Astra Control Center操作員
- 設定Astra控制中心
- 完整的Astra控制中心和操作員安裝
- [驗證系統狀態]

- [設定入口以進行負載平衡]
- 登入Astra Control Center UI

> 請勿刪除Astra Control Center運算子（例如、 `kubectl delete -f astra_control_center_operator_deploy.yaml`）在Astra Control Center安裝或操作期間、隨時避免刪除Pod。

下載並擷取**Astra Control Center**

1. 下載包含Astra Control Center的套裝組合 (`astra-control-center-[version].tar.gz`) "Astra Control Center 下載頁面"。
2. （建議但可選）下載Astra Control Center的憑證與簽名套件 (`astra-control-center-certs-[version].tar.gz`）驗證套件的簽名。

   展開以取得詳細資料

   ```
   tar -vxzf astra-control-center-certs-[version].tar.gz
   ```

   ```
   openssl dgst -sha256 -verify certs/AstraControlCenter-public.pub
   -signature certs/astra-control-center-[version].tar.gz.sig astra-
   control-center-[version].tar.gz
   ```

   隨即顯示輸出 `Verified OK` 驗證成功之後。

3. 從Astra Control Center套裝組合擷取映像：

   ```
   tar -vxzf astra-control-center-[version].tar.gz
   ```

安裝**NetApp Astra kubecl**外掛程式

您可以使用 NetApp Astra kubectl 命令列外掛程式、將影像推送至本機 Docker 儲存庫。

開始之前

NetApp為不同的CPU架構和作業系統提供外掛程式二進位檔。執行此工作之前、您必須先瞭解您的CPU和作業系統。

如果您已從先前的安裝中安裝外掛程式、 "請確定您擁有最新版本" 完成這些步驟之前。

步驟

1. 列出可用的 NetApp Astra Kubectl 外掛程式二進位檔：

   > KECBECTl外掛程式庫是tar套件的一部分、會擷取到資料夾中 `kubectl-astra`。

```
ls kubectl-astra/
```

2. 將作業系統和 CPU 架構所需的檔案移至目前路徑、並將其重新命名為 `kubectl-astra`：

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

將映像新增至本機登錄

1. 為您的Container引擎完成適當的步驟順序：

Docker

1. 切換到tar檔案的根目錄。您應該會看到 `acc.manifest.bundle.yaml` 檔案與這些目錄：

   ```
   acc/
   kubectl-astra/
   acc.manifest.bundle.yaml
   ```

2. 將Astra Control Center映像目錄中的套件映像推送到本機登錄。執行之前、請先進行下列替換 `push-images` 命令：

   - 以<BUNDLE_FILE> Astra Control套裝組合檔案的名稱取代 (`acc.manifest.bundle.yaml`）。
   - 以<MY_FULL_REGISTRY_PATH> Docker儲存庫的URL取代支援；例如 "<a href="https://<docker-registry>"" class="bare">https://<docker-registry>"。
   - 以<MY_REGISTRY_USER> 使用者名稱取代。
   - 以<MY_REGISTRY_TOKEN> 登錄的授權權杖取代。

     ```
     kubectl astra packages push-images -m <BUNDLE_FILE> -r
     <MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p
     <MY_REGISTRY_TOKEN>
     ```

Podman

1. 切換到tar檔案的根目錄。您應該會看到這個檔案和目錄：

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. 登入您的登錄：

   ```
   podman login <YOUR_REGISTRY>
   ```

3. 針對您使用的Podman版本、準備並執行下列其中一個自訂指令碼。以包含任何子目錄的儲存庫URL取代<MY_FULL_REGISTRY_PATH> 。

   ```
   <strong>Podman 4</strong>
   ```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

```
<strong>Podman 3</strong>
```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

指令碼所建立的映像路徑應如下所示、視登錄組態而定：

```
https://netappdownloads.jfrog.io/docker-astra-control-
prod/netapp/astra/acc/23.07.0-25/image:version
```

設定具有驗證需求之登錄的命名空間和機密

1. 匯出 Astra Control Center 主機叢集的 Kribeconfig ：

```
export KUBECONFIG=[file path]
```

完成安裝之前、請確定您的 Kupeconfig 指向您要安裝 Astra Control Center 的叢集。

2. 如果您使用需要驗證的登錄、則需要執行下列動作：

展開步驟

a. 建立 `netapp-acc-operator` 命名空間：

```
kubectl create ns netapp-acc-operator
```

b. 為建立秘密 `netapp-acc-operator` 命名空間。新增Docker資訊並執行下列命令：

預留位置 `your_registry_path` 應與您先前上傳的影像位置相符（例如、`[Registry_URL]/netapp/astra/astracc/23.07.0-25`）。

```
kubectl create secret docker-registry astra-registry-cred -n
netapp-acc-operator --docker-server=[your_registry_path] --docker
-username=[username] --docker-password=[token]
```

如果在產生機密之後刪除命名空間、請重新建立命名空間、然後重新產生命名空間的機密。

c. 建立 `netapp-acc` （或自訂命名）命名空間。

```
kubectl create ns [netapp-acc or custom namespace]
```

d. 為建立秘密 `netapp-acc` （或自訂命名）命名空間。新增Docker資訊並執行下列命令：

```
kubectl create secret docker-registry astra-registry-cred -n
[netapp-acc or custom namespace] --docker
-server=[your_registry_path] --docker-username=[username]
--docker-password=[token]
```

安裝Astra Control Center操作員

1. 變更目錄：

```
cd manifests
```

2. 編輯Astra Control Center營運者部署Yaml (`astra_control_center_operator_deploy.yaml`）以參考您的本機登錄和機密。

```
vim astra_control_center_operator_deploy.yaml
```

> 附註的Y反 洗錢範例遵循下列步驟。

a. 如果您使用需要驗證的登錄、請取代的預設行 `imagePullSecrets: []` 提供下列功能：

```
imagePullSecrets: [{name: astra-registry-cred}]
```

b. 變更 `ASTRA_IMAGE_REGISTRY` 適用於 `kube-rbac-proxy` 映像到您在中推入映像的登錄路徑 上一步。

c. 變更 `ASTRA_IMAGE_REGISTRY` 適用於 `acc-operator-controller-manager` 映像到您在中推入映像的登錄路徑 上一步。

展開範例 **Astra** 控制 **center_operer_deploy.yaml**

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
      - args:
        - --secure-listen-address=0.0.0.0:8443
        - --upstream=http://127.0.0.1:8080/
        - --logtostderr=true
        - --v=10
        image: ASTRA_IMAGE_REGISTRY/kube-rbac-proxy:v4.8.0
        name: kube-rbac-proxy
        ports:
        - containerPort: 8443
          name: https
      - args:
        - --health-probe-bind-address=:8081
        - --metrics-bind-address=127.0.0.1:8080
        - --leader-elect
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        - name: ACCOP_HELM_INSTALLTIMEOUT
          value: 5m
        image: ASTRA_IMAGE_REGISTRY/acc-operator:23.07.25
        imagePullPolicy: IfNotPresent
        livenessProbe:
          httpGet:
```

```
            path: /healthz
            port: 8081
          initialDelaySeconds: 15
          periodSeconds: 20
        name: manager
        readinessProbe:
          httpGet:
            path: /readyz
            port: 8081
          initialDelaySeconds: 5
          periodSeconds: 10
        resources:
          limits:
            cpu: 300m
            memory: 750Mi
          requests:
            cpu: 100m
            memory: 75Mi
        securityContext:
          allowPrivilegeEscalation: false
      imagePullSecrets: []
      securityContext:
        runAsUser: 65532
      terminationGracePeriodSeconds: 10
```

3. 安裝Astra Control Center操作員：

```
kubectl apply -f astra_control_center_operator_deploy.yaml
```

展開範例回應：

```
namespace/netapp-acc-operator created
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.as
tra.netapp.io created
role.rbac.authorization.k8s.io/acc-operator-leader-election-role
created
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role
created
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
created
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role
created
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding created
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding created
configmap/acc-operator-manager-config created
service/acc-operator-controller-manager-metrics-service created
deployment.apps/acc-operator-controller-manager created
```

4. 確認Pod正在執行：

```
kubectl get pods -n netapp-acc-operator
```

設定**Astra**控制中心

1. 編輯Astra Control Center自訂資源（CR）檔案 (`astra_control_center.yaml`）進行帳戶、支援、登錄及其他必要設定：

```
vim astra_control_center.yaml
```

附註的Y反 洗錢範例遵循下列步驟。

2. 修改或確認下列設定：

<code>accountName</code>

設定	指導	類型	範例
`accountName`	變更 `accountName` 字串至您要與Astra Control Center帳戶建立關聯的名稱。只能有一個帳戶名稱。	字串	`Example`

<code>astraVersion</code>

設定	指導	類型	範例
`astraVersion`	要部署的Astra Control Center版本。此設定不需要任何動作、因為此值將預先填入。	字串	`23.07.0-25`

<code>astraAddress</code>

設定	指導	類型	範例
`astraAddress`	變更 `astraAddress` 字串至您要在瀏覽器中使用的FQDN（建議）或IP位址、以存取Astra Control Center。此位址定義Astra Control Center在資料中心的找到方式、以及當您完成配置時、從負載平衡器配置的相同FQDN或IP位址 "Astra Control Center需求"。 附註：請勿使用 `http://` 或 `https://` 地址中。複製此FQDN以供在中使用 後續步驟。	字串	`astra.example.com`

<code>autoSupport</code>

您在本節中的選擇決定您是否會參與NetApp主動式支援應用程式NetApp Active IQ 功能、以及資料的傳送位置。需要網際網路連線（連接埠4442）、所有支援資料都會匿名。

設定	使用	指導	類型	範例
`autoSupport.enrolled`	也可以 `enrolled` 或 `url` 必須選取欄位	變更 `enrolled` for 解決方案AutoSupport `false` 適用於沒有網際網路連線或無法保留的網站 `true` 適用於連線站台。的設定 `true` 可將匿名資料傳送至 NetApp 以供支援之用。預設選項為 `false` 並表示不會將任何支援資料傳送給NetApp。	布林值	`false` （此值為預設值）
`autoSupport.url`	也可以 `enrolled` 或 `url` 必須選取欄位	此URL決定匿名資料的傳送位置。	字串	`https://support.netapp.com/asupprod/post/1.0/postAsup`

<code>email</code>

設定	指導	類型	範例
`email`	變更 `email` 字串至預設的初始系統管理員位址。複製此電子郵件地址以供在中使用 後續步驟。此電子郵件地址將作為初始帳戶登入UI的使用者名稱、並會收到Astra Control中事件的通知。	字串	`admin@example.com`

<code>firstName</code>

設定	指導	類型	範例
`firstName`	與Astra帳戶相關聯的預設初始系統管理員的名字。第一次登入後、此處使用的名稱會顯示在UI的標題中。	字串	`SRE`

<code>LastName</code>

設定	指導	類型	範例
`lastName`	與Astra帳戶相關聯的預設初始管理員姓氏。第一次登入後、此處使用的名稱會顯示在UI的標題中。	字串	`Admin`

<code>imageRegistry</code>

您在本節中的選擇定義了裝載Astra應用程式映像、Astra Control Center運算子和Astra Control Center Helm儲存庫的容器映像登錄。

設定	使用	指導	類型	範例
`imageRegistry.name`	必要	您在中推入映像的映像登錄名稱 上一步。請勿使用 `http://` 或 `https://` 在登錄名稱中。	字串	`example.registry.com/astra`
`imageRegistry.secret`	如果您輸入的字串則為必要 `imageRegistry.name' requires a secret.` `IMPORTANT: If you are using a registry that does not require authorization, you must delete this` `secret` 行內 `imageRegistry` 否則安裝將會失敗。	用來驗證映像登錄的Kubernetes機密名稱。	字串	`astra-registry-cred`

<code>storageClass</code>

設定	指導	類型	範例
`storageClass`	變更 `storageClass` 價值來源 `ontap-gold` 至安裝所需的另一個 Astra Trident storageClass 資源。執行命令 `kubectl get sc` 以判斷您現有的已設定儲存類別。必須在資訊清單檔案中輸入其中一個 Astra Trident 型儲存類別 (`astra-control-center-<version>.manifest`）、並將用於Astra PV。如果未設定、則會使用預設的儲存類別。 附註：如果已設定預設儲存類別、請確定它是唯一具有預設附註的儲存類別。	字串	`ontap-gold`

<code>volumeReclaimPolicy</code>

設定	指導	類型	選項
`volumeReclaimPolicy`	這為Astra的PV設定回收原則。將此原則設定為 `Retain` 刪除Astra後保留持續磁碟區。將此原則設定為 `Delete` 刪除Astra後刪除持續磁碟區。如果未設定此值、則會保留PV。	字串	• `Retain` （這是預設值） • `Delete`

<code>ingressType</code>

設定	指導	類型	選項
`ingressType`	使用下列其中一種入口類型： **`Generic`** (`ingressType: "Generic"`)（預設） 如果您使用另一個入口控制器、或偏好使用自己的入口控制器、請使用此選項。部署Astra Control Center之後、您需要設定 "入口控制器" 使用URL公開Astra Control Center。 **`AccTraefik`** (`ingressType: "AccTraefik"`) 如果您不想設定入口控制器、請使用此選項。這會部署Astra控制中心 `traefik` 作為Kubernetes負載平衡器類型服務的閘道。 Astra Control Center使用「負載平衡器」類型的服務 (`svc/traefik`（在Astra Control Center命名空間中）、並要求指派可存取的外部IP位址。如果您的環境允許負載平衡器、但您尚未設定負載平衡器、則可以使用MetalLB或其他外部服務負載平衡器、將外部IP位址指派給服務。在內部DNS伺服器組態中、您應該將Astra Control Center所選的DNS名稱指向負載平衡的IP位址。 附註：如需「負載平衡器」和入口服務類型的詳細資訊、請參閱 "需求"。	字串	• `Generic`（這是預設值） • `AccTraefik`

<code>scaleSize</code>

設定	指導	類型	選項
`scaleSize`	Astra 預設會使用高可用度（ HA ） `scaleSize` 的 `Medium` ，用於在 HA 中部署大多數服務並部署多個複本以實現冗餘。與 `scaleSize` 做為 `Small` 、 Astra 將減少所有服務的複本數量、但基本服務除外、以減少使用量。 秘訣： `Medium` 部署包含約 100 個 Pod （不包括暫時性工作負載）。100 個 Pod 以三個主節點和三個工作節點組態為基礎）。請注意、在您的環境中、每個 Pod 的網路限制可能是個問題、特別是在考慮災難恢復案例時。	字串	• `Small` • `Medium` （這是預設值）

<code>astraResourcesScaler</code>

設定	指導	類型	選項
`astraResourcesScaler`	適用的擴充選項適用於適用的適用範圍。依預設、Astra Control Center會針對Astra內的大部分元件設定資源要求來進行部署。此組態可讓Astra Control Center軟體堆疊在應用程式負載和擴充性增加的環境中、發揮更佳效能。 不過、在使用較小開發或測試叢集的案例中、則是使用「CR」欄位 `astraResourcesScalar` 可能設為 `Off` 。這會停用資源要求、並允許在較小的叢集上部署。	字串	• `Default` （這是預設值） • `Off`

<code>additionalValues</code>

在 Astra Control Center CR 中新增下列額外值、以避免在 23.07 安裝中出現已知問題：

```
additionalValues:
    polaris-keycloak:
      livenessProbe:
        initialDelaySeconds: 180
      readinessProbe:
        initialDelaySeconds: 180
```

- 對於 Astral Control Center 和 Cloud Insights 通訊、依預設會停用 TLS 憑證驗證。您可以在中新增下一節、以啟用 Cloud Insights 與 Astra 控制中心主機叢集和託管叢集之間通訊的 TLS 憑證驗證 `additionalValues`。

```
 additionalValues:
   netapp-monitoring-operator:
     config:
       ciSkipTlsVerify: false
   cloud-insights-service:
     config:
       ciSkipTlsVerify: false
   telemetry-service:
     config:
       ciSkipTlsVerify: false
```

<code>crds</code>

您在本節中的選擇決定Astra Control Center應如何處理客戶需求日。

設定	指導	類型	範例
`crds.externalCertManager`	如果您使用外部憑證管理程式、請變更 `externalCertManager` 至 `true`。預設值 `false` 讓Astra Control Center在安裝期間安裝自己的憑證管理程式客戶檔案。 CRD是整個叢集的物件、安裝這些物件可能會影響叢集的其他部分。您可以使用此旗標向Astra控制中心發出訊號、表示這些客戶需求日將由Astra控制中心外部的叢集管理員安裝及管理。	布林值	`False` （此值為預設值）
`crds.externalTraefik`	依預設、Astra Control Center會安裝必要的Traefik客戶需求日。CRD是整個叢集的物件、安裝這些物件可能會影響叢集的其他部分。您可以使用此旗標向Astra控制中心發出訊號、表示這些客戶需求日將由Astra控制中心外部的叢集管理員安裝及管理。	布林值	`False` （此值為預設值）

在完成安裝之前、請務必為您的組態選擇正確的儲存類別和入口類型。

展開範例 **Astra** 控制中心 **.yaml**

```
apiVersion: astra.netapp.io/v1
kind: AstraControlCenter
metadata:
  name: astra
spec:
  accountName: "Example"
  astraVersion: "ASTRA_VERSION"
  astraAddress: "astra.example.com"
  autoSupport:
    enrolled: true
  email: "[admin@example.com]"
  firstName: "SRE"
  lastName: "Admin"
  imageRegistry:
    name: "[your_registry_path]"
    secret: "astra-registry-cred"
  storageClass: "ontap-gold"
  volumeReclaimPolicy: "Retain"
  ingressType: "Generic"
  scaleSize: "Medium"
  astraResourcesScaler: "Default"
  additionalValues:
    polaris-keycloak:
      livenessProbe:
        initialDelaySeconds: 180
      readinessProbe:
        initialDelaySeconds: 180
  crds:
    externalTraefik: false
    externalCertManager: false
```

完整的**Astra**控制中心和操作員安裝

1. 如果您尚未在上一步中執行此動作、請建立 `netapp-acc` （或自訂）命名空間：

```
kubectl create ns [netapp-acc or custom namespace]
```

2. 在中安裝Astra Control Center `netapp-acc` （或自訂）命名空間：

```
kubectl apply -f astra_control_center.yaml -n [netapp-acc or custom
namespace]
```

Astra Control Center 駕駛員將自動檢查環境需求。遺失 "需求" 可能導致安裝失敗、或 Astra Control Center 無法正常運作。請參閱 下一節 檢查與自動系統檢查相關的警告訊息。

驗證系統狀態

您可以使用kubectl命令來驗證系統狀態。如果您偏好使用OpenShift、您可以使用相似的相關命令來進行驗證步驟。

步驟

1. 確認安裝程序未產生與驗證檢查相關的警告訊息：

```
kubectl get acc [astra or custom Astra Control Center CR name] -n
[netapp-acc or custom namespace] -o yaml
```

Astra Control Center 操作者記錄中也會報告其他警告訊息。

2. 修正自動化需求檢查所回報的環境問題。

您可以確保環境符合、以修正問題 "需求" 適用於 Astra Control Center 。

3. 驗證是否已成功安裝所有系統元件。

```
kubectl get pods -n [netapp-acc or custom namespace]
```

每個Pod的狀態應為 `Running`。部署系統Pod可能需要幾分鐘的時間。

展開以取得範例回應

```
NAME                                          READY   STATUS
RESTARTS     AGE
acc-helm-repo-6cc7696d8f-pmhm8                1/1     Running     0
9h
activity-597fb656dc-5rd4l                     1/1     Running     0
9h
activity-597fb656dc-mqmcw                     1/1     Running     0
9h
api-token-authentication-62f84                1/1     Running     0
9h
api-token-authentication-68nlf                1/1     Running     0
9h
api-token-authentication-ztgrm                1/1     Running     0
9h
asup-669d4ddbc4-fnmwp                         1/1     Running     1
(9h ago)   9h
authentication-78789d7549-lk686               1/1     Running     0
9h
bucketservice-65c7d95496-24x7l                1/1     Running     3
(9h ago)   9h
cert-manager-c9f9fbf9f-k8zq2                  1/1     Running     0
9h
cert-manager-c9f9fbf9f-qjlzm                  1/1     Running     0
9h
cert-manager-cainjector-dbbbd8447-b5qll       1/1     Running     0
9h
cert-manager-cainjector-dbbbd8447-p5whs       1/1     Running     0
9h
cert-manager-webhook-6f97bb7d84-4722b         1/1     Running     0
9h
cert-manager-webhook-6f97bb7d84-86kv5         1/1     Running     0
9h
certificates-59d9f6f4bd-2j899                 1/1     Running     0
9h
certificates-59d9f6f4bd-9d9k6                 1/1     Running     0
9h
certificates-expiry-check-28011180--1-8lkxz   0/1     Completed   0
9h
cloud-extension-5c9c9958f8-jdhrp              1/1     Running     0
9h
cloud-insights-service-5cdd5f7f-pp8r5         1/1     Running     0
9h
composite-compute-66585789f4-hxn5w            1/1     Running     0
```

```
9h
composite-volume-68649f68fd-tb7p4             1/1     Running     0
9h
credentials-dfc844c57-jsx92                   1/1     Running     0
9h
credentials-dfc844c57-xw26s                   1/1     Running     0
9h
entitlement-7b47769b87-4jb6c                  1/1     Running     0
9h
features-854d8444cc-c24b7                     1/1     Running     0
9h
features-854d8444cc-dv6sm                     1/1     Running     0
9h
fluent-bit-ds-9tlv4                           1/1     Running     0
9h
fluent-bit-ds-bpkcb                           1/1     Running     0
9h
fluent-bit-ds-cxmwx                           1/1     Running     0
9h
fluent-bit-ds-jgnhc                           1/1     Running     0
9h
fluent-bit-ds-vtr6k                           1/1     Running     0
9h
fluent-bit-ds-vxqd5                           1/1     Running     0
9h
graphql-server-7d4b9d44d5-zdbf5               1/1     Running     0
9h
identity-6655c48769-4pwk8                     1/1     Running     0
9h
influxdb2-0                                   1/1     Running     0
9h
keycloak-operator-55479d6fc6-slvmt            1/1     Running     0
9h
krakend-f487cb465-78679                       1/1     Running     0
9h
krakend-f487cb465-rjsxx                       1/1     Running     0
9h
license-64cbc7cd9c-qxsr8                      1/1     Running     0
9h
login-ui-5db89b5589-ndb96                     1/1     Running     0
9h
loki-0                                        1/1     Running     0
9h
metrics-facade-8446f64c94-x8h7b               1/1     Running     0
9h
monitoring-operator-6b44586965-pvcl4          2/2     Running     0
```

```
9h
nats-0                                            1/1     Running     0
9h
nats-1                                            1/1     Running     0
9h
nats-2                                            1/1     Running     0
9h
nautilus-85754d87d7-756qb                         1/1     Running     0
9h
nautilus-85754d87d7-q8j7d                         1/1     Running     0
9h
openapi-5f9cc76544-7fnjm                          1/1     Running     0
9h
openapi-5f9cc76544-vzr7b                          1/1     Running     0
9h
packages-5db49f8b5-lrzhd                          1/1     Running     0
9h
polaris-consul-consul-server-0                    1/1     Running     0
9h
polaris-consul-consul-server-1                    1/1     Running     0
9h
polaris-consul-consul-server-2                    1/1     Running     0
9h
polaris-keycloak-0                                1/1     Running     2
(9h ago)   9h
polaris-keycloak-1                                1/1     Running     0
9h
polaris-keycloak-2                                1/1     Running     0
9h
polaris-keycloak-db-0                             1/1     Running     0
9h
polaris-keycloak-db-1                             1/1     Running     0
9h
polaris-keycloak-db-2                             1/1     Running     0
9h
polaris-mongodb-0                                 1/1     Running     0
9h
polaris-mongodb-1                                 1/1     Running     0
9h
polaris-mongodb-2                                 1/1     Running     0
9h
polaris-ui-66fb99479-qp9gq                        1/1     Running     0
9h
polaris-vault-0                                   1/1     Running     0
9h
polaris-vault-1                                   1/1     Running     0
```

```
9h
polaris-vault-2                                   1/1     Running     0
9h
public-metrics-76fbf9594d-zmxzw                   1/1     Running     0
9h
storage-backend-metrics-7d7fbc9cb9-lmd25          1/1     Running     0
9h
storage-provider-5bdd456c4b-2fftc                 1/1     Running     0
9h
task-service-87575df85-dnn2q                      1/1     Running     3
(9h ago)   9h
task-service-task-purge-28011720--1-q6w4r         0/1     Completed   0
28m
task-service-task-purge-28011735--1-vk6pd         1/1     Running     0
13m
telegraf-ds-2r2kw                                 1/1     Running     0
9h
telegraf-ds-6s9d5                                 1/1     Running     0
9h
telegraf-ds-96jl7                                 1/1     Running     0
9h
telegraf-ds-hbp84                                 1/1     Running     0
9h
telegraf-ds-plwzv                                 1/1     Running     0
9h
telegraf-ds-sr22c                                 1/1     Running     0
9h
telegraf-rs-4sbg8                                 1/1     Running     0
9h
telemetry-service-fb9559f7b-mk9l7                 1/1     Running     3
(9h ago)   9h
tenancy-559bbc6b48-5msgg                          1/1     Running     0
9h
traefik-d997b8877-7xpf4                           1/1     Running     0
9h
traefik-d997b8877-9xv96                           1/1     Running     0
9h
trident-svc-585c97548c-d25z5                      1/1     Running     0
9h
vault-controller-88484b454-2d6sr                  1/1     Running     0
9h
vault-controller-88484b454-fc5cz                  1/1     Running     0
9h
vault-controller-88484b454-jktld                  1/1     Running     0
9h
```

4. （選用）觀看 `acc-operator` 監控進度的記錄：

```
kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-
operator -c manager -f
```

> `accHost` 叢集登錄是最後一項作業、如果失敗、也不會導致部署失敗。如果記錄中指出叢集登錄失敗、您可以透過再次嘗試登錄 "在UI中新增叢集工作流程" 或API。

5. 當所有Pod都在執行時、請確認安裝成功 (`READY` 是 `True`）並取得您登入Astra Control Center時所使用的初始設定密碼：

```
kubectl get AstraControlCenter -n [netapp-acc or custom namespace]
```

回應：

```
NAME    UUID                                   VERSION     ADDRESS
READY
astra   9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f  23.07.0-25
10.111.111.111  True
```

> 複製UUID值。密碼是 `ACC-` 接著是UUID值 (`ACC-[UUID]` 或者、在此範例中、 `ACC-9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f`）。

設定入口以進行負載平衡

您可以設定Kubernetes入口控制器來管理外部服務存取。如果您使用的預設值、這些程序會提供入口控制器的設定範例 `ingressType: "Generic"` Astra Control Center自訂資源 (`astra_control_center.yaml`）。如果您指定、則不需要使用此程序 `ingressType: "AccTraefik"` Astra Control Center自訂資源 (`astra_control_center.yaml`）。

部署Astra Control Center之後、您需要設定入口控制器、以URL顯示Astra Control Center。

設定步驟視您使用的入口控制器類型而有所不同。Astra Control Center支援多種入站控制器類型。這些設定程序提供一些常見入口控制器類型的範例步驟。

開始之前

- 必要的 "入口控制器" 應已部署。
- 。 "入口等級" 應已建立對應於入口控制器的。

Istio入侵步驟

1. 設定Istio入口。

   此程序假設使用「預設」組態設定檔來部署Istio。

2. 收集或建立Ingress閘道所需的憑證和私密金鑰檔案。

   您可以使用CA簽署或自我簽署的憑證。一般名稱必須是Astra位址（FQDN）。

   命令範例：

   ```
   openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key
   -out tls.crt
   ```

3. 建立秘密 `tls secret name` 類型 `kubernetes.io/tls` 中的TLS私密金鑰和憑證 `istio-system namespace` 如TLS機密所述。

   命令範例：

   ```
   kubectl create secret tls [tls secret name] --key="tls.key"
   --cert="tls.crt" -n istio-system
   ```

   機密名稱應與相符 `spec.tls.secretName` 提供於 `istio-ingress.yaml` 檔案：

4. 在中部署入口資源 `netapp-acc` （或自訂命名）命名空間、使用v1資源類型作為架構 (`istio-Ingress.yaml` 在本例中使用）：

```
apiVersion: networking.k8s.io/v1
kind: IngressClass
metadata:
  name: istio
spec:
  controller: istio.io/ingress-controller
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress
  namespace: [netapp-acc or custom namespace]
spec:
  ingressClassName: istio
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: [ACC address]
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: traefik
            port:
              number: 80
```

5. 套用變更：

```
kubectl apply -f istio-Ingress.yaml
```

6. 檢查入侵狀態：

```
kubectl get ingress -n [netapp-acc or custom namespace]
```

回應：

```
NAME    CLASS HOSTS              ADDRESS         PORTS   AGE
ingress istio astra.example.com 172.16.103.248  80, 443 1h
```

7. 完成Astra Control Center安裝。

適用於**Nginvin像** 控制器的步驟

1. 建立類型的秘密 `kubernetes.io/tls` 中的TLS私密金鑰和憑證 `netapp-acc` （或自訂命名）命名空間、如所述 "TLS機密"。
2. 在中部署入口資源 `netapp-acc` （或自訂命名）命名空間、使用v1資源類型作為架構 (`nginx-Ingress.yaml` 在本例中使用）：

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: netapp-acc-ingress
  namespace: [netapp-acc or custom namespace]
spec:
  ingressClassName: [class name for nginx controller]
  tls:
  - hosts:
    - <ACC address>
    secretName: [tls secret name]
  rules:
  - host: <ACC address>
    http:
      paths:
        - path:
          backend:
            service:
              name: traefik
              port:
                number: 80
          pathType: ImplementationSpecific
```

3. 套用變更：

```
kubectl apply -f nginx-Ingress.yaml
```

NetApp建議將Ngin像 控制器安裝為部署、而非 `daemonSet`。

OpenShift入口控制器的步驟

1. 取得您的憑證、取得可供OpenShift路由使用的金鑰、憑證和CA檔案。
2. 建立OpenShift路由：

```
oc create route edge --service=traefik --port=web -n [netapp-acc or
custom namespace] --insecure-policy=Redirect --hostname=<ACC
address> --cert=cert.pem --key=key.pem
```

登入**Astra Control Center UI**

安裝Astra Control Center之後、您將變更預設管理員的密碼、並登入Astra Control Center UI儀表板。

步驟

1. 在瀏覽器中、輸入 FQDN （包括 `https://` 字首） `astraAddress` 在中 `astra_control_center.yaml` 請於何時進行 您安裝了Astra Control Center。
2. 收到提示時、請接受自我簽署的憑證。

   您可以在登入後建立自訂憑證。

3. 在Astra Control Center登入頁面、輸入您使用的值 `email` 在中 `astra_control_center.yaml` 請於何時進行 您安裝了Astra Control Center，然後輸入初始設定密碼 (`ACC-[UUID]`）。

   如果您輸入錯誤密碼三次、系統將鎖定管理員帳戶15分鐘。

4. 選擇*登入*。
5. 出現提示時變更密碼。

   如果這是您第一次登入、但您忘記密碼、而且尚未建立其他管理使用者帳戶、請聯絡 "NetApp支援" 以取得密碼恢復協助。

6. （選用）移除現有的自我簽署TLS憑證、並以取代 "由憑證授權單位（CA）簽署的自訂TLS憑證"。

疑難排解安裝

如果有任何服務存在 `Error` 狀態、您可以檢查記錄。尋找400到500範圍內的API回應代碼。這些都表示發生故障的地點。

選項

- 若要檢查Astra控制中心的操作員記錄、請輸入下列內容：

```
kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-
operator -c manager -f
```

- 若要檢查 Astra Control Center CR 的輸出：

```
kubectl get acc -n [netapp-acc or custom namespace] -o yaml
```

下一步

- （選用）視您的環境而定、請在安裝後完成 "組態步驟"。
- 執行以完成部署 "設定工作"。

設定外部憑證管理程式

如果Kubernetes叢集中已存在憑證管理程式、您需要執行一些必要步驟、使Astra Control Center不會安裝自己的憑證管理程式。

步驟

1. 確認您已安裝憑證管理程式：

```
kubectl get pods -A | grep 'cert-manager'
```

回應範例：

```
cert-manager   essential-cert-manager-84446f49d5-sf2zd              1/1
Running    0      6d5h
cert-manager   essential-cert-manager-cainjector-66dc99cc56-9ldmt   1/1
Running    0      6d5h
cert-manager   essential-cert-manager-webhook-56b76db9cc-fjqrq      1/1
Running    0      6d5h
```

2. 為建立憑證/金鑰配對 `astraAddress` FQDN：

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key -out
tls.crt
```

回應範例：

```
Generating a 2048 bit RSA private key
..................+++
.......................................+++
writing new private key to 'tls.key'
```

3. 使用先前產生的檔案建立秘密：

```
kubectl create secret tls selfsigned-tls --key tls.key --cert tls.crt -n
<cert-manager-namespace>
```

回應範例：

```
secret/selfsigned-tls created
```

4. 建立 `ClusterIssuer` 以下*確切*的檔案包含您的命名空間位置 `cert-manager` 已安裝Pod：

```
apiVersion: cert-manager.io/v1
kind: ClusterIssuer
metadata:
  name: astra-ca-clusterissuer
  namespace: <cert-manager-namespace>
spec:
  ca:
    secretName: selfsigned-tls
```

```
kubectl apply -f ClusterIssuer.yaml
```

回應範例：

```
clusterissuer.cert-manager.io/astra-ca-clusterissuer created
```

5. 確認 `ClusterIssuer` 已正確啟動。 `Ready` 必須是 `True` 在繼續之前：

```
kubectl get ClusterIssuer
```

回應範例：

```
NAME                      READY   AGE
astra-ca-clusterissuer    True    9s
```

6. 完成 "Astra Control Center安裝程序"。有 "Astra Control Center叢集Yaml所需的組態步驟" 您可在其中變更CRD值、以表示外部安裝了憑證管理程式。您必須在安裝期間完成此步驟、Astra Control Center才能辨識外部憑證管理程式。

使用OpenShift作業系統集線器安裝Astra Control Center

如果您使用Red Hat OpenShift、可以使用Red Hat認證的操作員來安裝Astra Control Center。請使用此程序從安裝Astra Control Center "Red Hat生態系統目錄" 或使用Red Hat OpenShift Container Platform。

完成此程序之後、您必須返回安裝程序、才能完成 "剩餘步驟" 以驗證安裝是否成功並登入。

開始之前

- 符合環境先決條件： "開始安裝之前、請先準備好環境以進行Astra Control Center部署"。
- 健全的叢集運算子與API服務：
  - 從OpenShift叢集確保所有叢集操作員都處於健全狀態：

```
oc get clusteroperators
```

  - 從OpenShift叢集、確保所有API服務都處於健全狀態：

```
oc get apiservices
```

- * FQDN位址*：取得資料中心Astra Control Center的FQDN位址。
- * OpenShift權限*：取得必要的權限並存取Red Hat OpenShift Container Platform、以執行所述的安裝步驟。
- 已設定的憑證管理程式：如果叢集中已存在憑證管理程式、您需要執行某些作業 "必要步驟" 因此Astra Control Center不會安裝自己的憑證管理程式。依預設、Astra Control Center會在安裝期間安裝自己的憑證管理程式。
- * Kubernetes入口控制器*：如果您有一個Kubernetes入口控制器來管理外部服務存取、例如叢集中的負載平衡、您就需要將其設定為與Astra Control Center搭配使用：
  a. 建立運算子命名空間：

```
oc create namespace netapp-acc-operator
```

  b. "完成設定" 適用於您的入口控制器類型。

步驟

- 下載並擷取Astra Control Center
- 安裝NetApp Astra kubecl外掛程式
- [將映像新增至本機登錄]
- [尋找操作員安裝頁面]
- [安裝操作員]
- 安裝Astra Control Center

下載並擷取**Astra Control Center**

1. 下載包含Astra Control Center的套裝組合 (`astra-control-center-[version].tar.gz`) "Astra Control Center 下載頁面"。
2. （建議但可選）下載Astra Control Center的憑證與簽名套件 (`astra-control-center-certs-[version].tar.gz`）驗證套件的簽名。

   展開以取得詳細資料

   ```
   tar -vxzf astra-control-center-certs-[version].tar.gz
   ```

   ```
   openssl dgst -sha256 -verify certs/AstraControlCenter-public.pub
   -signature certs/astra-control-center-[version].tar.gz.sig astra-
   control-center-[version].tar.gz
   ```

   隨即顯示輸出 `Verified OK` 驗證成功之後。

3. 從Astra Control Center套裝組合擷取映像：

   ```
   tar -vxzf astra-control-center-[version].tar.gz
   ```

安裝**NetApp Astra kubecl**外掛程式

您可以使用 NetApp Astra kubectl 命令列外掛程式、將影像推送至本機 Docker 儲存庫。

開始之前

NetApp為不同的CPU架構和作業系統提供外掛程式二進位檔。執行此工作之前、您必須先瞭解您的CPU和作業系統。

步驟

1. 列出可用的NetApp Astra kubectl外掛程式二進位檔、並記下作業系統和CPU架構所需的檔案名稱：

   KECBECTl外掛程式庫是tar套件的一部分、會擷取到資料夾中 `kubectl-astra`。

   ```
   ls kubectl-astra/
   ```

2. 將正確的二進位檔移至目前路徑、並將其重新命名為 `kubectl-astra`：

   ```
   cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
   ```

將映像新增至本機登錄

1. 為您的Container引擎完成適當的步驟順序：

Docker

1. 切換到tar檔案的根目錄。您應該會看到 `acc.manifest.bundle.yaml` 檔案與這些目錄：

   ```
   acc/
   kubectl-astra/
   acc.manifest.bundle.yaml
   ```

2. 將Astra Control Center映像目錄中的套件映像推送到本機登錄。執行之前、請先進行下列替換 `push-images` 命令：

   - 以<BUNDLE_FILE> Astra Control套裝組合檔案的名稱取代 (`acc.manifest.bundle.yaml`）。
   - 以<MY_FULL_REGISTRY_PATH> Docker儲存庫的URL取代支援；例如 "<a href="https://<docker-registry>"" class="bare">https://<docker-registry>"。
   - 以<MY_REGISTRY_USER> 使用者名稱取代。
   - 以<MY_REGISTRY_TOKEN> 登錄的授權權杖取代。

     ```
     kubectl astra packages push-images -m <BUNDLE_FILE> -r
     <MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p
     <MY_REGISTRY_TOKEN>
     ```

Podman

1. 切換到tar檔案的根目錄。您應該會看到這個檔案和目錄：

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. 登入您的登錄：

   ```
   podman login <YOUR_REGISTRY>
   ```

3. 針對您使用的Podman版本、準備並執行下列其中一個自訂指令碼。以包含任何子目錄的儲存庫URL取代<MY_FULL_REGISTRY_PATH> 。

   ```
   <strong>Podman 4</strong>
   ```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

```
<strong>Podman 3</strong>
```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

指令碼所建立的映像路徑應如下所示、視登錄組態而定：

```
https://netappdownloads.jfrog.io/docker-astra-control-
prod/netapp/astra/acc/23.07.0-25/image:version
```

尋找操作員安裝頁面

1. 請完成下列其中一個程序、以存取操作員安裝頁面：

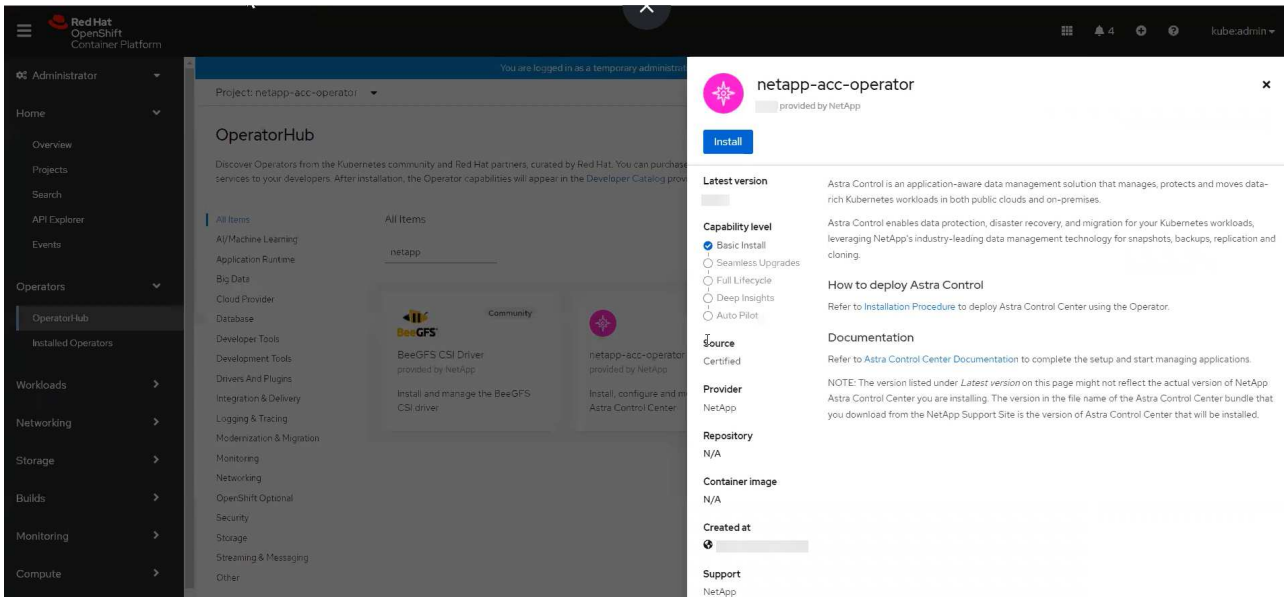
   ◦ 從Red Hat Openshift Web主控台：

i. 登入OpenShift Container Platform UI。

ii. 從側功能表中、選取*運算子>運算子中樞*。

> 您只能使用此運算子升級至 Astra Control Center 的目前版本。

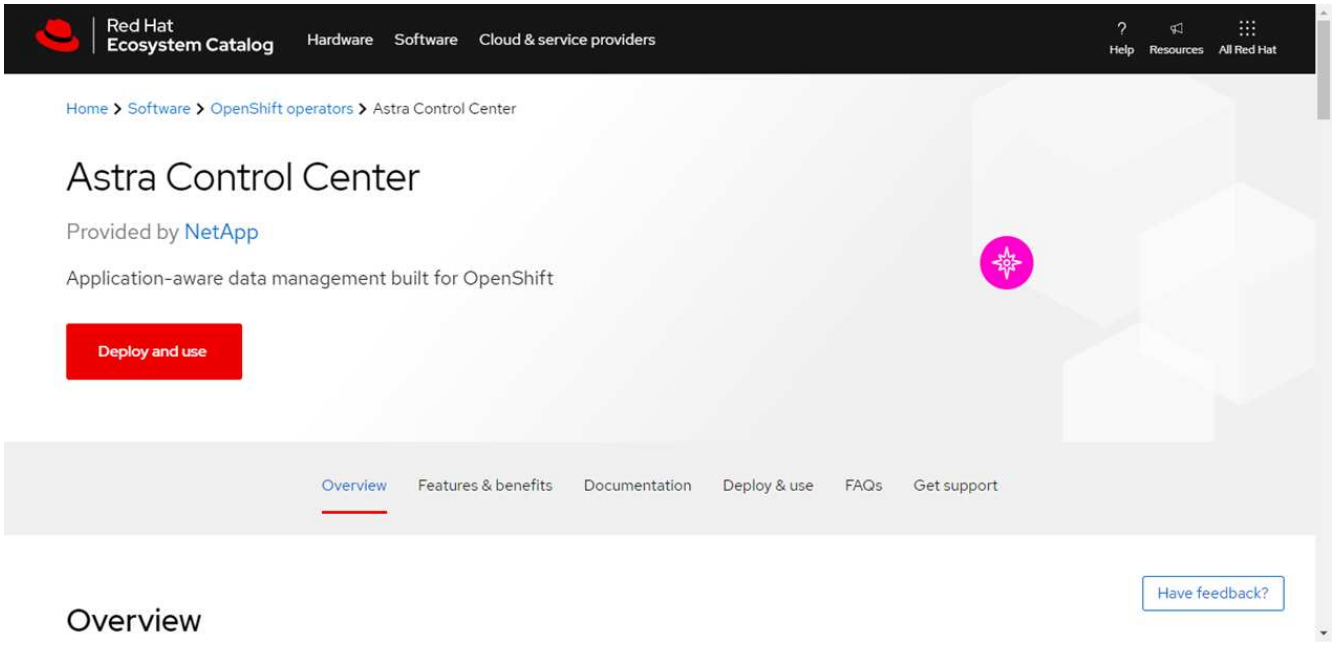
iii. 搜尋並選擇NetApp Astra Control Center營運者。



◦ 從Red Hat生態系統目錄：

i. 選擇NetApp Astra Control Center "營運者"。

ii. 選擇*部署和使用*。



安裝操作員

1. 完成*安裝操作員*頁面並安裝操作員：

此運算子可用於所有叢集命名空間。

a. 選取運算子命名空間或 `netapp-acc-operator` 命名空間將會自動建立、做為操作員安裝的一部分。

b. 選取手動或自動核准策略。

建議手動核准。每個叢集只能執行單一運算子執行個體。

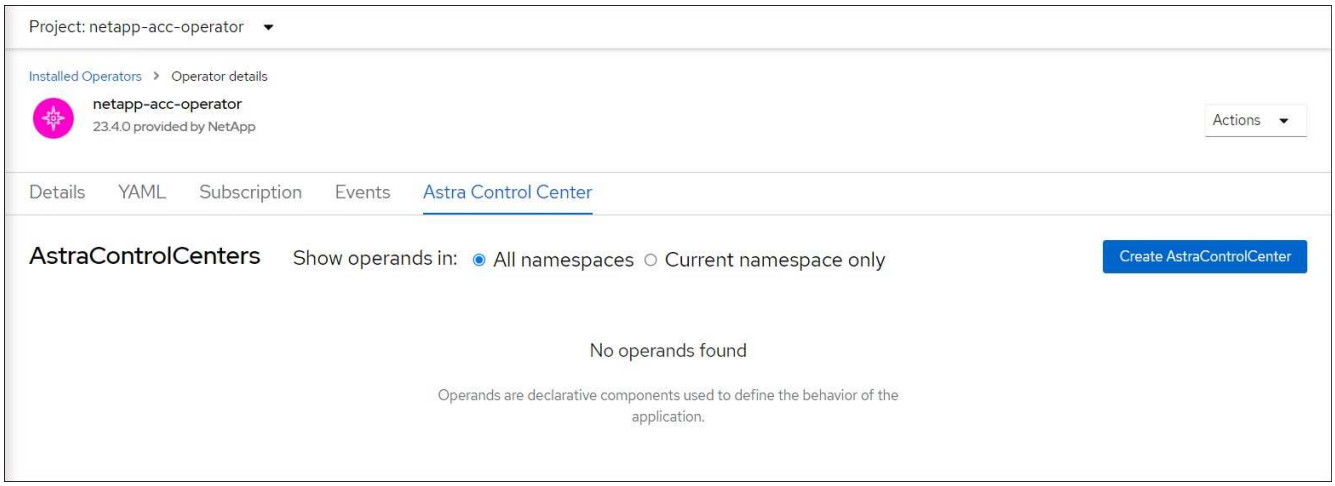
c. 選擇*安裝*。

如果您選擇手動核准策略、系統會提示您核准此操作員的手動安裝計畫。

2. 從主控台移至「作業系統集線器」功能表、確認操作員已成功安裝。

安裝Astra Control Center

1. 從Astra控制中心操作員* Astra控制中心*索引標籤內的主控台、選取*建立適用的*。



2. 完成 `Create AstraControlCenter` 表單欄位：

a. 保留或調整Astra Control Center名稱。

b. 新增Astra Control Center的標籤。

c. 啟用或停用自動支援。建議保留「自動支援」功能。

d. 輸入Astra Control Center FQDN或IP位址。請勿進入 `http://` 或 `https://` 在「地址」欄位中。

e. 輸入 Astra Control Center 版本、例如 23.07.0-25 。

f. 輸入帳戶名稱、電子郵件地址和管理員姓氏。

g. 選擇的Volume回收原則 `Retain`、 `Recycle`或 `Delete`。預設值為 `Retain`。

h. 選取安裝的 scaleSize 。

Astra 預設會使用高可用度（ HA） `scaleSize` 的 `Medium`，用於在 HA 中部署大多數服務並部署多個複本以實現冗餘。與 `scaleSize` 做為 `Small`、 Astra 將減少所有服務的複本數量、但基本服務除外、以減少使用量。

i. 選取入口類型：

- **Generic** (ingressType: "Generic") （預設）

  如果您使用另一個入口控制器、或偏好使用自己的入口控制器、請使用此選項。部署Astra Control Center之後、您需要設定 "入口控制器" 使用URL公開Astra Control Center。

- **AccTraefik** (ingressType: "AccTraefik")

  如果您不想設定入口控制器、請使用此選項。這會部署Astra控制中心 `traefik` 閘道即Kubernetes「負載平衡器」類型服務。

Astra Control Center使用「負載平衡器」類型的服務 (`svc/traefik` （在Astra Control Center命名空間中）、並要求指派可存取的外部IP位址。如果您的環境允許負載平衡器、但您尚未設定負載平衡器、則可以使用MetalLB或其他外部服務負載平衡器、將外部IP位址指派給服務。在內部DNS伺服器組態中、您應該將Astra Control Center所選的DNS名稱指向負載平衡的IP位址。

如需「負載平衡器」和入口服務類型的詳細資訊、請參閱 "需求"。

a. 在*映像登錄*中、輸入您的本機容器映像登錄路徑。請勿進入 `http://` 或 `https://` 在「地址」欄位中。

b. 如果您使用需要驗證的映像登錄、請輸入映像秘密。

如果您使用需要驗證的登錄、 在叢集上建立秘密。

c. 輸入管理員名字。

d. 設定資源擴充。

e. 提供預設的儲存類別。

如果已設定預設儲存類別、請確定它是唯一具有預設註釋的儲存類別。

f. 定義客戶需求日處理偏好設定。

3. 選取「Yaml」檢視以檢閱您所選的設定。

4. 選取 `Create`。

建立登錄機密

如果您使用需要驗證的登錄、請在Openshift叢集上建立密碼、然後在中輸入密碼名稱 `Create AstraControlCenter` 表單欄位。

1. 為Astra Control Center運算子建立命名空間：

```
oc create ns [netapp-acc-operator or custom namespace]
```

2. 在此命名空間中建立秘密：

```
oc create secret docker-registry astra-registry-cred n [netapp-acc-
operator or custom namespace] --docker-server=[your_registry_path]
--docker username=[username] --docker-password=[token]
```

Astra Control僅支援Docker登錄機密。

3. 填寫中的其餘欄位 「Create」（建立）「吧！Control Center」表單欄位。

下一步

完成 "剩餘步驟" 若要驗證Astra Control Center是否安裝成功、請設定入口控制器（選用）、然後登入UI。此外、您還需要執行 "設定工作" 安裝完成後。

安裝Astra Control Center搭配Cloud Volumes ONTAP 一套功能性儲存後端

有了Astra Control Center、您就能在混合雲環境中使用自我管理的Kubernetes叢集和Cloud Volumes ONTAP 實例來管理應用程式。您可以在內部部署的Kubernetes叢集或雲端環境中的其中一個自我管理Kubernetes叢集上部署Astra Control Center。

有了其中一項部署、您就能使用Cloud Volumes ONTAP 下列其中一項部署、以下列方式執行應用程式資料管理作業：將NetApp當成儲存後端。您也可以將S3儲存區設定為備份目標。

若要在Amazon Web Services（AWS）、Google Cloud Platform（GCP）和Microsoft Azure中安裝Astra Control Center、並搭配Cloud Volumes ONTAP 使用整套儲存後端、請視您的雲端環境而定、執行下列步驟。

- 在Amazon Web Services中部署Astra Control Center
- 在Google Cloud Platform中部署Astra Control Center
- 在Microsoft Azure中部署Astra Control Center

您可以使用自我管理的Kubernetes叢集、例如OpenShift Container Platform（OCP）、在發佈版本中管理應用程式。只有自我管理的OCP叢集已通過驗證、可用於部署Astra Control Center。

在Amazon Web Services中部署Astra Control Center

您可以在Amazon Web Services（AWS）公有雲上的自我管理Kubernetes叢集上部署Astra Control Center。

AWS所需的功能

在AWS中部署Astra Control Center之前、您需要下列項目：

- Astra Control Center授權。請參閱 "Astra Control Center授權要求"。
- "符合Astra Control Center的要求"。
- NetApp Cloud Central帳戶
- 如果使用OCP、則Red Hat OpenShift Container Platform（OCP）權限（位於命名空間層級以建立Pod）
- AWS認證資料、存取ID和秘密金鑰、具備可讓您建立儲存區和連接器的權限

- AWS帳戶彈性容器登錄（ECR）存取與登入
- 存取Astra Control UI所需的AWS託管區域和Route 53項目

AWS的作業環境需求

Astra Control Center需要下列AWS作業環境：

- Red Hat OpenShift Container Platform 4.11 至 4.13

確保您選擇裝載Astra Control Center的作業環境符合環境正式文件中所述的基本資源需求。

除了環境的資源需求之外、Astra Control Center還需要下列資源：

元件	需求
後端**NetApp Cloud Volumes ONTAP** 功能儲存容量	至少提供300 GB
工作者節點（**AWS EC2**需求）	總共至少3個工作節點、每個節點有4個vCPU核心和12GB RAM
負載平衡器	服務類型「負載平衡器」可用於將入口流量傳送至作業環境叢集中的服務
FQDN	將Astra Control Center的FQDN指向負載平衡IP位址的方法
Astra Trident（安裝於**NetApp BlueXP**（前身為**Cloud Manager**）的**Kubernetes**叢集探索中）	Astra Trident 22.10 或更新版本已安裝及設定、 NetApp ONTAP 9.8 或更新版本則做為儲存後端
映像登錄	NetApp 提供登錄、可讓您用來取得 Astra 控制中心建置映像： http://netappdownloads.jfrog.io/docker-astra-control-prod 請聯絡 NetApp 支援部門、取得在 Astra 控制中心安裝過程中使用此映像登錄的說明。 如果您無法存取 NetApp 映像登錄、則必須擁有現有的私有登錄、例如 AWS 彈性容器登錄（ ECR ）、您可以將 Astra 控制中心建置映像推送至該登錄。您需要提供映像登錄的URL、以便上傳映像。 Astra Control Center託管叢集和託管叢集必須能夠存取相同的映像登錄、才能使用還原型映像來備份和還原應用程式。

元件	需求
Astra Trident / ONTAP Estra組態	Astra Control Center需要建立儲存類別、並將其設為預設儲存類別。Astra Control Center支援下列ONTAP 將Kubernetes叢集匯入NetApp BlueXP（前身為Cloud Manager）時所建立的支援功能。這些資料由Astra Trident提供： • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

這些需求假設Astra Control Center是營運環境中唯一執行的應用程式。如果環境正在執行其他應用程式、請相應調整這些最低需求。

AWS登錄權杖會在12小時內過期、之後您必須更新Docker映像登錄機密。

AWS部署總覽

以下是安裝Astra Control Center for AWS的程序總覽、Cloud Volumes ONTAP 其中包含以作為儲存後端的功能。

以下將詳細說明每個步驟。

1. 確保您擁有足夠的IAM權限。
2. 在AWS上安裝RedHat OpenShift叢集。
3. 設定 AWS。
4. 設定適用於AWS的NetApp BlueXP。
5. 安裝AWS的Astra Control Center。

確保您擁有足夠的IAM權限

確保您擁有足夠的IAM角色和權限、可讓您安裝RedHat OpenShift叢集和NetApp BlueXP（前身為Cloud Manager）Connector。

請參閱 "初始 AWS 認證資料"。

在AWS上安裝RedHat OpenShift叢集

在AWS上安裝RedHat OpenShift Container Platform叢集。

如需安裝指示、請參閱 "在OpenShift Container Platform的AWS上安裝叢集"。

接下來、設定 AWS 來建立虛擬網路、設定 EC2 運算執行個體、以及建立 AWS S3 儲存區。如果您無法存取 NetApp Astra 控制中心影像登錄、您也需要建立「彈性容器登錄」（ ECR ）來主控 Astra Control Center 影像、並將影像推送至此登錄。

請遵循AWS文件完成下列步驟。請參閱 "AWS安裝文件"。

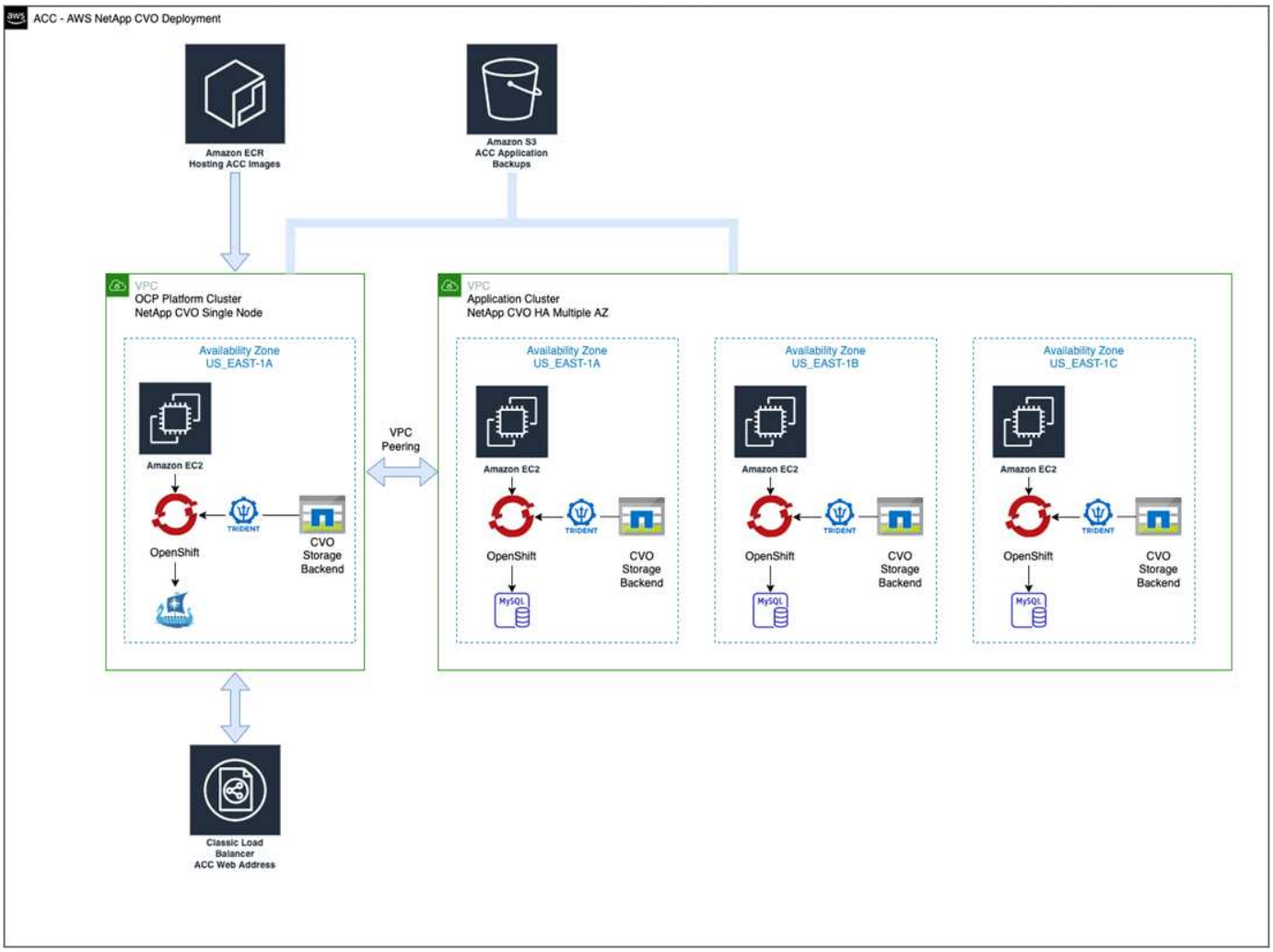
1. 建立AWS虛擬網路。
2. 檢閱EC2運算執行個體。這可以是AWS中的裸機伺服器或VM。
3. 如果執行個體類型尚未符合主節點和工作節點的Astra最低資源需求、請在AWS中變更執行個體類型以符合Astra需求。 請參閱 "Astra Control Center需求"。
4. 建立至少一個AWS S3儲存區來儲存備份。
5. （選用）如果您無法存取 NetApp 映像登錄、請執行下列步驟：
   a. 建立 AWS 彈性容器登錄（ ECR ）以裝載所有 Astra Control Center 影像。

      如果您未建立ECR、Astra Control Center將無法從含有Cloud Volumes ONTAP AWS後端的支援的叢集存取監控資料。此問題是因為您嘗試使用Astra Control Center探索及管理的叢集無法存取AWS ECR。

   b. 將 Astra Control Center 影像推送到您定義的登錄。

AWS Elastic Container登錄（ECR）權杖會在12小時後過期、導致跨叢集複製作業失敗。從Cloud Volumes ONTAP 針對AWS設定的功能進行的功能區管理儲存後端時、就會發生此問題。若要修正此問題、請再次向ECR驗證、並產生新的秘密、讓複製作業順利恢復。

以下是AWS部署範例：



設定適用於**AWS**的**NetApp BlueXP**

使用NetApp BlueXP（前身為Cloud Manager）建立工作區、新增AWS連接器、建立工作環境、以及匯入叢集。

請遵循BlueXP文件完成下列步驟。請參閱下列內容：

- "開始使用Cloud Volumes ONTAP AWS的功能"。
- "使用BlueXP在AWS中建立連接器"

步驟

1. 將您的認證資料新增至BlueXP。
2. 建立工作區。
3. 新增AWS的連接器。選擇AWS做為供應商。
4. 為您的雲端環境建立工作環境。
    a. 位置：「 Amazon Web Services （ AWS ）」
    b. 類型：Cloud Volumes ONTAP 「EHA」
5. 匯入OpenShift叢集。叢集將連線至您剛建立的工作環境。
    a. 選擇* K8s*>*叢集清單*>*叢集詳細資料*、即可檢視NetApp叢集詳細資料。

b. 請注意右上角的 Astra Trident 版本。

c. 請注意Cloud Volumes ONTAP 、顯示NetApp為資源配置程式的叢集儲存類別。

這會匯入您的Red Hat OpenShift叢集、並將其指派為預設儲存類別。您可以選取儲存類別。Astra Trident 會在匯入和探索程序中自動安裝。

6. 請注意此Cloud Volumes ONTAP 功能部署中的所有持續磁碟區和磁碟區。

可作為單一節點或高可用度運作。Cloud Volumes ONTAP如果已啟用HA、請記下在AWS中執行的HA狀態和節點部署狀態。

安裝AWS的Astra Control Center

遵循標準 "Astra Control Center安裝說明"。

AWS使用一般S3儲存區類型。

在Google Cloud Platform中部署Astra Control Center

您可以在Google Cloud Platform（GCP）公有雲上的自我管理Kubernetes叢集上部署Astra Control Center。

GCP的必備功能

在 GCP 中部署 Astra Control Center 之前、您需要下列項目：

- Astra Control Center授權。請參閱 "Astra Control Center授權要求"。
- "符合Astra Control Center的要求"。
- NetApp Cloud Central帳戶
- 如果使用的是 OCP 、 Red Hat OpenShift Container Platform （ OCP ） 4.11 至 4.13
- 如果使用OCP、則Red Hat OpenShift Container Platform（OCP）權限（位於命名空間層級以建立Pod）
- GCP服務帳戶具備權限、可讓您建立貯體和連接器

GCP 的營運環境需求

確保您選擇裝載Astra Control Center的作業環境符合環境正式文件中所述的基本資源需求。

除了環境的資源需求之外、Astra Control Center還需要下列資源：

元件	需求
後端NetApp Cloud Volumes ONTAP 功能儲存容量	至少提供300 GB
工作者節點（GCP運算需求）	總共至少3個工作節點、每個節點有4個vCPU核心和12GB RAM
負載平衡器	服務類型「負載平衡器」可用於將入口流量傳送至作業環境叢集中的服務

元件	需求
FQDN（GCP DNS區域）	將Astra Control Center的FQDN指向負載平衡IP位址的方法
Astra Trident（安裝於NetApp BlueXP（前身為Cloud Manager）的Kubernetes叢集探索中）	Astra Trident 22.10 或更新版本已安裝及設定、 NetApp ONTAP 9.8 或更新版本則做為儲存後端
映像登錄	NetApp 提供登錄、可讓您用來取得 Astra 控制中心建置映像： http://netappdownloads.jfrog.io/docker-astra-control-prod 請聯絡 NetApp 支援部門、取得在 Astra 控制中心安裝過程中使用此映像登錄的說明。 如果無法存取 NetApp 映像登錄、您必須擁有現有的私有登錄、例如 Google Container 登錄、才能將 Astra 控制中心建置映像推送至該登錄。您需要提供映像登錄的URL、以便上傳映像。 您必須啟用匿名存取、才能拉出還原映像進行備份。
Astra Trident / ONTAP Estra組態	Astra Control Center需要建立儲存類別、並將其設為預設儲存類別。Astra Control Center支援下列ONTAP 將Kubernetes叢集匯入NetApp BlueXP時所建立的物件庫伯內特儲存類別。這些資料由Astra Trident提供： • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

這些需求假設Astra Control Center是營運環境中唯一執行的應用程式。如果環境正在執行其他應用程式、請相應調整這些最低需求。

GCP 部署總覽

以下是將Astra Control Center安裝在GCP的自我管理OCP叢集上的程序總覽、Cloud Volumes ONTAP 其中包含以作儲存後端的功能。

以下將詳細說明每個步驟。

1. 在 GCP 上安裝 RedHat OpenShift 叢集。
2. 建立GCP專案和虛擬私有雲端。
3. 確保您擁有足夠的IAM權限。

4. 設定 GCP。
5. 為 GCP 設定 NetApp BlueXP。
6. 安裝Astra Control Center for GCP。

在 GCP 上安裝 RedHat OpenShift 叢集

第一步是在GCP上安裝RedHat OpenShift叢集。

如需安裝指示、請參閱下列內容：

- "在GCP中安裝OpenShift叢集"
- "建立GCP服務帳戶"

建立GCP專案和虛擬私有雲端

建立至少一個GCP專案和虛擬私有雲端（VPC）。

OpenShift可能會建立自己的資源群組。此外、您也應該定義GCP VPC。請參閱OpenShift文件。

您可能想要建立平台叢集資源群組和目標應用程式OpenShift叢集資源群組。

確保您擁有足夠的IAM權限

確保您擁有足夠的IAM角色和權限、可讓您安裝RedHat OpenShift叢集和NetApp BlueXP（前身為Cloud Manager）Connector。

請參閱 "初始GCP認證與權限"。

設定 GCP

接下來、設定 GCP 以建立 VPC 、設定運算執行個體、以及建立 Google Cloud Object Storage 。如果您無法存取 NetApp Astra 控制中心影像登錄、您也需要建立 Google Container 登錄、以裝載 Astra Control Center 影像、並將影像推送至此登錄。

請依照 GCP 文件完成下列步驟。請參閱在GCP中安裝OpenShift叢集。

1. 在您計畫用於具有CVO後端的OCP叢集的GCP中建立GCP專案和VPC。
2. 檢閱運算執行個體。這可以是 GCP 中的裸機伺服器或 VM 。
3. 如果執行個體類型尚未符合主要節點和工作節點的 Astra 最低資源需求、請在 GCP 中變更執行個體類型、以符合 Astra 需求。請參閱 "Astra Control Center需求"。
4. 建立至少一個GCP雲端儲存庫來儲存備份。
5. 建立儲存貯體存取所需的機密。
6. （選用）如果您無法存取 NetApp 映像登錄、請執行下列步驟：
   a. 建立 Google Container 登錄以裝載 Astra Control Center 映像。
   b. 設定所有Astra Control Center映像的Google Container登錄存取權、以供Docker推/拉。

      範例： Astra Control Center 影像可透過輸入下列指令碼、推送至此登錄：

```
gcloud auth activate-service-account <service account email address>
--key-file=<GCP Service Account JSON file>
```

此指令碼需要Astra Control Center資訊清單檔案和Google Image登錄位置。

範例：

```
manifestfile=astra-control-center-<version>.manifest
GCP_CR_REGISTRY=<target image registry>
ASTRA_REGISTRY=<source Astra Control Center image registry>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $GCP_CR_REGISTRY/$image
    docker push $GCP_CR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

1. 設定DNS區域。

為 GCP 設定 NetApp BlueXP

使用NetApp BlueXP（前身為Cloud Manager）建立工作區、將連接器新增至GCP、建立工作環境、以及匯入叢集。

請遵循BlueXP文件完成下列步驟。請參閱 "從GCP開始使用Cloud Volumes ONTAP"。

開始之前

- 以所需的IAM權限和角色存取GCP服務帳戶

步驟

1. 將您的認證資料新增至BlueXP。請參閱 "新增GCP帳戶"。
2. 新增 GCP 連接器。
   a. 選擇「GCP」作為供應商。
   b. 輸入GCP認證。請參閱 "從BlueXP在GCP中建立連接器"。
   c. 確認連接器正在執行、並切換至該連接器。
3. 為您的雲端環境建立工作環境。
   a. 地點：「 GCP 」
   b. 類型：Cloud Volumes ONTAP 「EHA」
4. 匯入OpenShift叢集。叢集將連線至您剛建立的工作環境。

a. 選擇* K8s*>*叢集清單*>*叢集詳細資料*、即可檢視NetApp叢集詳細資料。

b. 請注意右上角的Trident版本。

c. 請注意Cloud Volumes ONTAP 、顯示「NetApp」為資源配置程式的叢集儲存類別。

這會匯入您的Red Hat OpenShift叢集、並將其指派為預設儲存類別。您可以選取儲存類別。Astra Trident 會在匯入和探索程序中自動安裝。

5. 請注意此Cloud Volumes ONTAP 功能部署中的所有持續磁碟區和磁碟區。

可作為單一節點或高可用度（HA）運作。Cloud Volumes ONTAP如果 HA 已啟用、請注意 GCP 中執行的 HA 狀態和節點部署狀態。

安裝Astra Control Center for GCP

遵循標準 "Astra Control Center安裝說明"。

GCP使用通用S3儲存區類型。

1. 產生Docker祕密以擷取Astra Control Center安裝的映像：

```
kubectl create secret docker-registry <secret name> --docker
-server=<Registry location> --docker-username=_json_key --docker
-password="$(cat <GCP Service Account JSON file>)" --namespace=pcloud
```

在Microsoft Azure中部署Astra Control Center

您可以將Astra Control Center部署在Microsoft Azure公有雲上的自我管理Kubernetes叢集上。

Azure的必備功能

在Azure中部署Astra Control Center之前、您需要下列項目：

- Astra Control Center授權。請參閱 "Astra Control Center授權要求"。
- "符合Astra Control Center的要求"。
- NetApp Cloud Central帳戶
- 如果使用的是 OCP 、 Red Hat OpenShift Container Platform （ OCP ） 4.11 至 4.13
- 如果使用OCP、則Red Hat OpenShift Container Platform（OCP）權限（位於命名空間層級以建立Pod）
- Azure認證、具備可讓您建立儲存區和連接器的權限

Azure的營運環境需求

確保您選擇裝載Astra Control Center的作業環境符合環境正式文件中所述的基本資源需求。

除了環境的資源需求之外、Astra Control Center還需要下列資源：

請參閱 "Astra Control Center營運環境需求"。

元件	需求
後端**NetApp Cloud Volumes ONTAP** 功能儲存容量	至少提供300 GB
工作者節點（**Azure**運算需求）	總共至少3個工作節點、每個節點有4個vCPU核心和12GB RAM
負載平衡器	服務類型「負載平衡器」可用於將入口流量傳送至作業環境叢集中的服務
FQDN（**Azure DNS**區域）	將Astra Control Center的FQDN指向負載平衡IP位址的方法
Astra Trident（安裝於**NetApp BlueXP**的**Kubernetes**叢集探索中）	Astra Trident 22.10 或更新版本已安裝及設定、 NetApp ONTAP 9.8 或更新版本將用作儲存後端
映像登錄	NetApp 提供登錄、可讓您用來取得 Astra 控制中心建置映像：`http://netappdownloads.jfrog.io/docker-astra-control-prod` 請聯絡 NetApp 支援部門、取得在 Astra 控制中心安裝過程中使用此映像登錄的說明。 如果您無法存取 NetApp 映像登錄、您必須擁有現有的私有登錄、例如 Azure Container 登錄（ ACR ）、才能將 Astra 控制中心建置映像推送至該登錄。您需要提供映像登錄的URL、以便上傳映像。 您必須啟用匿名存取、才能拉出還原映像進行備份。
Astra Trident / ONTAP Estra組態	Astra Control Center需要建立儲存類別、並將其設為預設儲存類別。Astra Control Center支援下列ONTAP 將Kubernetes叢集匯入NetApp BlueXP時所建立的物件庫伯內特儲存類別。這些資料由Astra Trident提供： • `vsaworkingenvironment-<>-ha-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-ha-san csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-nas csi.trident.netapp.io` • `vsaworkingenvironment-<>-single-san csi.trident.netapp.io`

這些需求假設Astra Control Center是營運環境中唯一執行的應用程式。如果環境正在執行其他應用程式、請相應調整這些最低需求。

Azure部署總覽

以下是安裝Astra Control Center for Azure的程序總覽。

以下將詳細說明每個步驟。

1. 在Azure上安裝RedHat OpenShift叢集。
2. 建立Azure資源群組。
3. 確保您擁有足夠的IAM權限。
4. 設定Azure。
5. 設定適用於Azure的NetApp BlueXP（前身為Cloud Manager）。
6. 安裝及設定Azure的Astra Control Center。

在Azure上安裝RedHat OpenShift叢集

第一步是在Azure上安裝RedHat OpenShift叢集。

如需安裝指示、請參閱下列內容：

- "在Azure上安裝OpenShift叢集"。
- "安裝Azure帳戶"。

建立Azure資源群組

建立至少一個Azure資源群組。

> OpenShift可能會建立自己的資源群組。此外、您也應該定義Azure資源群組。 請參閱OpenShift文件。

您可能想要建立平台叢集資源群組和目標應用程式OpenShift叢集資源群組。

確保您擁有足夠的IAM權限

確保您擁有足夠的IAM角色和權限、可讓您安裝RedHat OpenShift叢集和NetApp BlueXP Connector。

請參閱 "Azure 認證與權限"。

設定Azure

接下來、將 Azure 設定為建立虛擬網路、設定運算執行個體、以及建立 Azure Blob 容器。如果您無法存取 NetApp Astra 控制中心影像登錄、您也需要建立 Azure Container 登錄（ACR）來主控 Astra Control Center 映像、並將映像推送至此登錄。

請依照Azure文件完成下列步驟。請參閱 "在Azure上安裝OpenShift叢集"。

1. 建立Azure虛擬網路。
2. 檢閱運算執行個體。這可以是Azure中的裸機伺服器或VM。
3. 如果執行個體類型尚未符合主節點和工作節點的Astra最低資源需求、請變更Azure中的執行個體類型以符

合Astra要求。 請參閱 "Astra Control Center需求"。

4. 建立至少一個Azure Blob容器來儲存備份。
5. 建立儲存帳戶。您需要儲存帳戶來建立容器、以便在Astra Control Center中作為儲存庫。
6. 建立儲存貯體存取所需的機密。
7. （選用）如果您無法存取 NetApp 映像登錄、請執行下列步驟：

   a. 建立 Azure Container 登錄（ ACR ）以裝載 Astra Control Center 映像。
   b. 為所有 Astra Control Center 影像設定 Docker 推 / 拉存取。
   c. 使用下列指令碼將 Astra Control Center 影像推入此登錄：

```
az acr login -n <AZ ACR URL/Location>
This script requires the Astra Control Center manifest file and your
Azure ACR location.
```

   範例：

```
manifestfile=astra-control-center-<version>.manifest
AZ_ACR_REGISTRY=<target image registry>
ASTRA_REGISTRY=<source Astra Control Center image registry>

while IFS= read -r image; do
    echo "image: $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRY/$image"
    root_image=${image%:*}
    echo $root_image
    docker pull $ASTRA_REGISTRY/$image
    docker tag $ASTRA_REGISTRY/$image $AZ_ACR_REGISTRYY/$image
    docker push $AZ_ACR_REGISTRY/$image
done < astra-control-center-22.04.41.manifest
```

8. 設定DNS區域。

設定適用於**Azure**的**NetApp BlueXP**（前身為**Cloud Manager**）

使用BlueXP（前身為Cloud Manager）建立工作區、將連接器新增至Azure、建立工作環境、以及匯入叢集。

請遵循BlueXP文件完成下列步驟。請參閱 "Azure中的BlueXP入門指南"。

開始之前

以所需的IAM權限和角色存取Azure帳戶

步驟

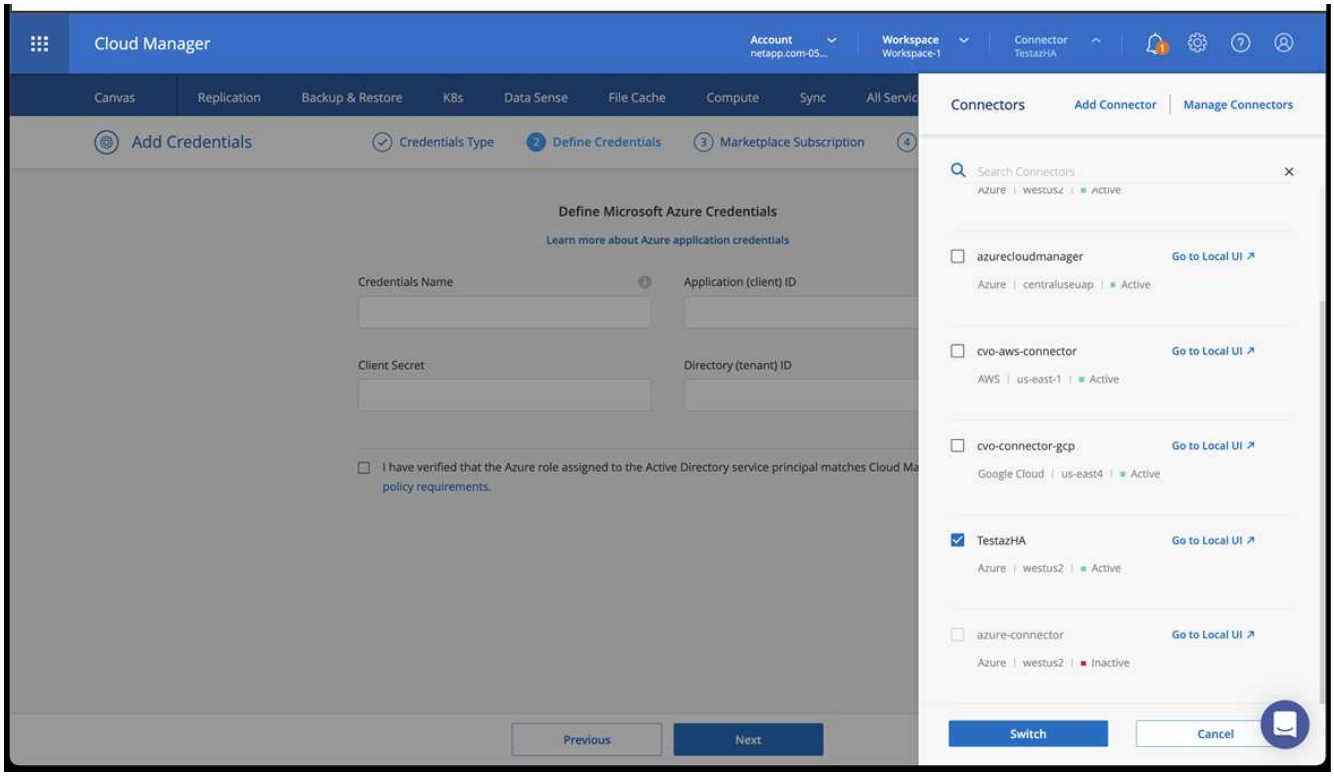
1. 將您的認證資料新增至BlueXP。
2. 新增Azure連接器。請參閱 "BlueXP原則"。

a. 選擇* Azure *作為供應商。

b. 輸入Azure認證資料、包括應用程式ID、用戶端機密和目錄（租戶）ID。

請參閱 "從BlueXPr在Azure中建立連接器"。

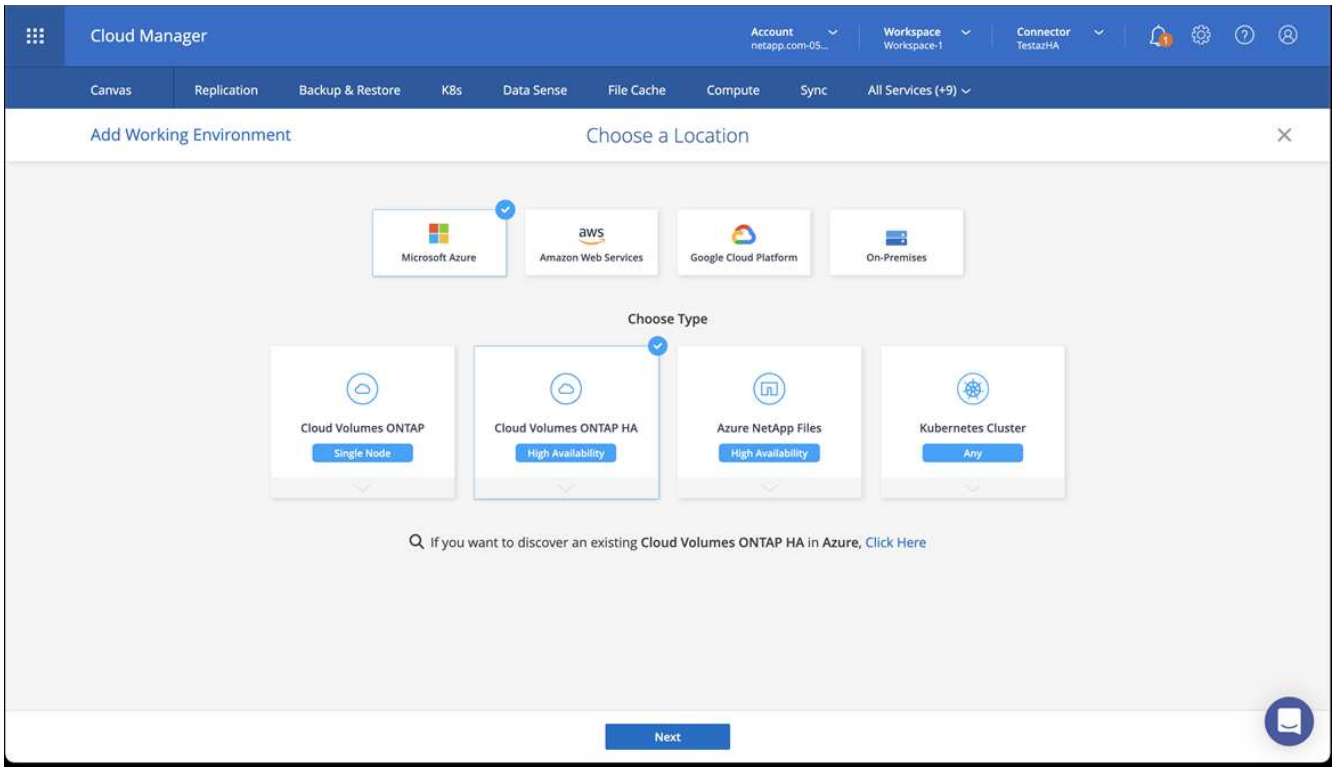
3. 確認連接器正在執行、並切換至該連接器。



4. 為您的雲端環境建立工作環境。

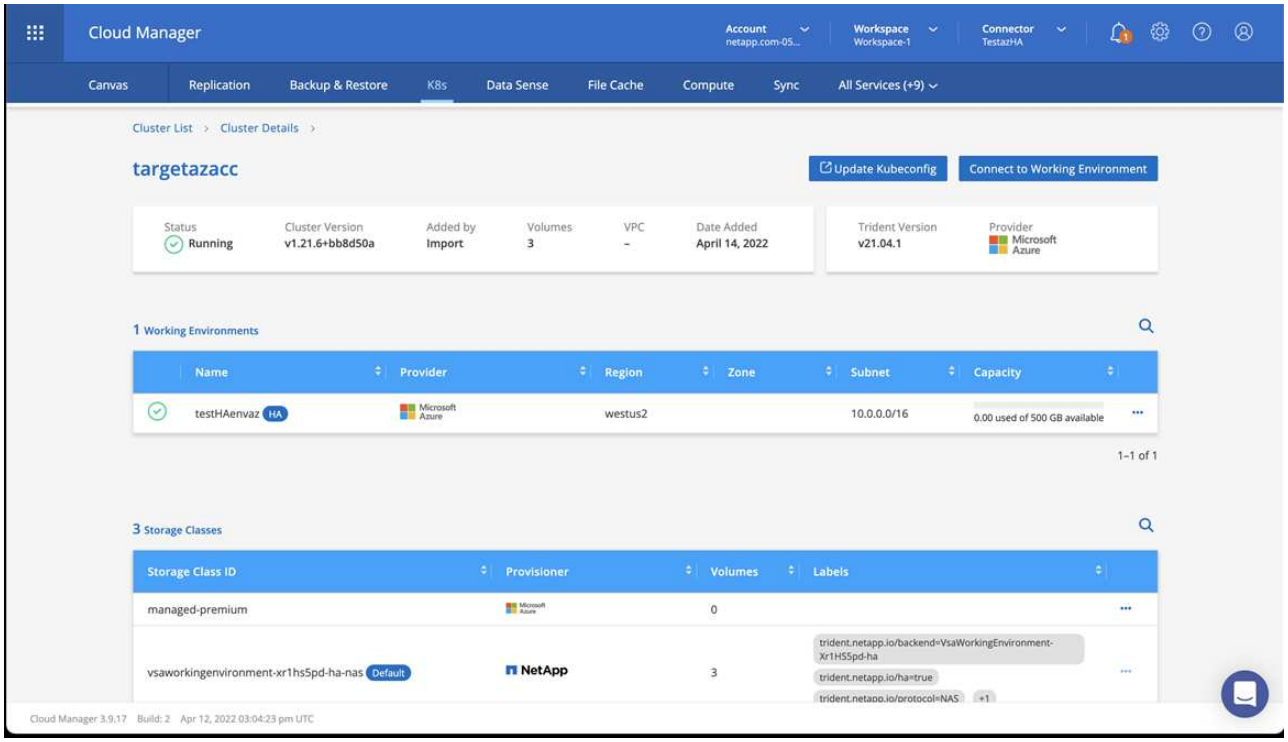
a. 位置：「Microsoft Azure」。

b. 輸入：Cloud Volumes ONTAP 「EHA」。



5. 匯入OpenShift叢集。叢集將連線至您剛建立的工作環境。

   a. 選擇* K8s*>*叢集清單*>*叢集詳細資料*、即可檢視NetApp叢集詳細資料。



   b. 請注意右上角的 Astra Trident 版本。

   c. 請注意Cloud Volumes ONTAP 、顯示NetApp為資源配置程式的叢集儲存類別。

這會匯入您的Red Hat OpenShift叢集、並指派預設的儲存類別。您可以選取儲存類別。

Astra Trident 會在匯入和探索程序中自動安裝。

6. 請注意此Cloud Volumes ONTAP 功能部署中的所有持續磁碟區和磁碟區。
7. 可作為單一節點或高可用度運作。Cloud Volumes ONTAP如果已啟用HA、請記下Azure中執行的HA狀態和節點部署狀態。

安裝及設定Azure的Astra Control Center

使用標準安裝Astra Control Center "安裝說明"。

使用Astra Control Center新增Azure儲存庫。請參閱 "設定Astra Control Center並新增鏟斗"。

安裝後設定Astra Control Center

視您的環境而定、安裝Astra Control Center之後可能需要額外的組態。

移除資源限制

某些環境使用資源配額和限制範圍物件、以防止命名空間中的資源消耗叢集上的所有可用CPU和記憶體。Astra Control Center並未設定上限、因此不符合這些資源。如果您的環境是以這種方式設定、則需要從您打算安裝Astra Control Center的命名空間中移除這些資源。

您可以使用下列步驟擷取及移除這些配額和限制。在這些範例中、命令輸出會在命令之後立即顯示。

步驟

1. 取得中的資源配額 `netapp-acc` （或自訂命名）命名空間：

```
kubectl get quota -n [netapp-acc or custom namespace]
```

回應：

```
NAME          AGE   REQUEST                                              LIMIT
pods-high     16s   requests.cpu: 0/20, requests.memory: 0/100Gi
limits.cpu: 0/200, limits.memory: 0/1000Gi
pods-low      15s   requests.cpu: 0/1, requests.memory: 0/1Gi
limits.cpu: 0/2, limits.memory: 0/2Gi
pods-medium   16s   requests.cpu: 0/10, requests.memory: 0/20Gi
limits.cpu: 0/20, limits.memory: 0/200Gi
```

2. 依名稱刪除所有資源配額：

```
kubectl delete resourcequota pods-high -n [netapp-acc or custom
namespace]
```

```
kubectl delete resourcequota pods-low -n [netapp-acc or custom
namespace]
```

```
kubectl delete resourcequota pods-medium -n [netapp-acc or custom
namespace]
```

3. 取得中的限制範圍 `netapp-acc` （或自訂命名）命名空間：

```
kubectl get limits -n [netapp-acc or custom namespace]
```

回應：

```
NAME              CREATED AT
cpu-limit-range   2022-06-27T19:01:23Z
```

4. 依名稱刪除限制範圍：

```
kubectl delete limitrange cpu-limit-range -n [netapp-acc or custom
namespace]
```

新增自訂**TLS**憑證

Astra Control Center預設使用自我簽署的TLS憑證來處理入站控制器流量（僅適用於特定組態）、以及使用網頁瀏覽器進行Web UI驗證。您可以移除現有的自我簽署TLS憑證、並以由憑證授權單位（CA）簽署的TLS憑證取代。

預設的自我簽署憑證可用於兩種類型的連線：

- HTTPS連線至Astra Control Center網路UI
- 入口控制器流量（僅當 `ingressType: "AccTraefik"` 內容已在中設定 `astra_control_center.yaml` 安裝Astra Control Center期間的檔案）

取代預設TLS憑證會取代用於驗證這些連線的憑證。

開始之前

- Kubernetes叢集已安裝Astra Control Center
- 管理存取叢集上要執行的命令Shell `kubectl` 命令
- 來自CA的私密金鑰和憑證檔案

移除自我簽署的憑證

移除現有的自我簽署TLS憑證。

1. 使用SSH、以管理使用者身分登入裝載Astra Control Center的Kubernetes叢集。
2. 使用下列取代命令尋找與目前憑證相關的TLS密碼 `<ACC-deployment-namespace>` 使用Astra Control Center部署命名空間：

```
kubectl get certificate -n <ACC-deployment-namespace>
```

3. 使用下列命令刪除目前安裝的機密與憑證：

```
kubectl delete cert cert-manager-certificates -n <ACC-deployment-
namespace>
```

```
kubectl delete secret secure-testing-cert -n <ACC-deployment-namespace>
```

使用命令列新增憑證

新增由CA簽署的TLS憑證。

1. 使用下列命令以CA的私密金鑰和憑證檔案建立新的TLS秘密，並以適當的資訊取代括弧<>中的引數：

```
kubectl create secret tls <secret-name> --key <private-key-filename>
--cert <certificate-filename> -n <ACC-deployment-namespace>
```

2. 使用下列命令和範例編輯叢集自訂資源定義（CRD）檔案、然後變更 `spec.selfSigned` 價值 `spec.ca.secretName` 若要參考您先前建立的TLS秘密：

```
kubectl edit clusterissuers.cert-manager.io/cert-manager-certificates -n
<ACC-deployment-namespace>
```

客戶需求日：

```
#spec:
#  selfSigned: {}

spec:
  ca:
    secretName: <secret-name>
```

3. 使用下列命令和輸出範例來驗證變更是否正確、以及叢集是否已準備好驗證憑證、取代 `<ACC-deployment-namespace>` 使用Astra Control Center部署命名空間：

```
kubectl describe clusterissuers.cert-manager.io/cert-manager-
certificates -n <ACC-deployment-namespace>
```

回應：

```
Status:
  Conditions:
    Last Transition Time:  2021-07-01T23:50:27Z
    Message:               Signing CA verified
    Reason:                KeyPairVerified
    Status:                True
    Type:                  Ready
Events:                    <none>
```

4. 建立 `certificate.yaml` 使用下列範例將方括弧<>中的預留位置值取代為適當資訊的檔案：

```
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  <strong>name: <certificate-name></strong>
  namespace: <ACC-deployment-namespace>
spec:
  <strong>secretName: <certificate-secret-name></strong>
  duration: 2160h # 90d
  renewBefore: 360h # 15d
  dnsNames:
  <strong>- <astra.dnsname.example.com></strong> #Replace with the
correct Astra Control Center DNS address
  issuerRef:
    kind: ClusterIssuer
    name: cert-manager-certificates
```

5. 使用下列命令建立憑證：

```
kubectl apply -f certificate.yaml
```

6. 使用下列命令和範例輸出來驗證憑證是否已正確建立、以及是否已使用您在建立期間所指定的引數（例如名稱、持續時間、續約期限及DNS名稱）。

```
kubectl describe certificate -n <ACC-deployment-namespace>
```

回應：

```
Spec:
  Dns Names:
    astra.example.com
  Duration:  125h0m0s
  Issuer Ref:
    Kind:         ClusterIssuer
    Name:         cert-manager-certificates
  Renew Before:  61h0m0s
  Secret Name:   <certificate-secret-name>
Status:
  Conditions:
    Last Transition Time:  2021-07-02T00:45:41Z
    Message:               Certificate is up to date and has not expired
    Reason:                Ready
    Status:                True
    Type:                  Ready
  Not After:               2021-07-07T05:45:41Z
  Not Before:              2021-07-02T00:45:41Z
  Renewal Time:            2021-07-04T16:45:41Z
  Revision:                1
Events:                    <none>
```

7. 編輯 TLS 儲存 CRD 以使用下列命令和範例指向您的新憑證密碼名稱、以適當的資訊取代括弧 <> 中的預留位置值

```
kubectl edit tlsstores.traefik.io -n <ACC-deployment-namespace>
```

客戶需求日：

```
...
spec:
  defaultCertificate:
    secretName: <certificate-secret-name>
```

8. 編輯「入口CRD TLS」選項、使用下列命令和範例指向新的憑證密碼、並以適當的資訊取代方括弧<>中的預留位置值：

```
kubectl edit ingressroutes.traefik.io -n <ACC-deployment-namespace>
```

客戶需求日：

```
...
 tls:
    secretName: <certificate-secret-name>
```

9. 使用網頁瀏覽器瀏覽至Astra Control Center的部署IP位址。
10. 確認憑證詳細資料與您安裝的憑證詳細資料相符。
11. 匯出憑證並將結果匯入網頁瀏覽器中的憑證管理程式。

設定Astra控制中心

安裝 Astra Control Center 、登入 UI 並變更密碼之後、您將需要設定授權、新增叢集、啟用驗證、管理儲存設備及新增儲存區。

工作

- 新增Astra Control Center授權
- 使用Astra Control為環境做好叢集管理準備
- [新增叢集]
- 在 ONTAP 儲存後端啟用驗證
- [新增儲存後端]
- [新增儲存庫]

新增Astra Control Center授權

安裝 Astra Control Center 時、已安裝內嵌評估授權。如果您正在評估 Astra Control Center 、可以跳過此步驟。

您可以使用Astra Control UI或新增授權 "Astra Control API"。

Astra Control Center授權會使用Kubernetes CPU單元來測量CPU資源、並計算指派給所有受管理Kubernetes叢集之工作節點的CPU資源。授權是根據vCPU使用率而定。如需如何計算授權的詳細資訊、請參閱 "授權"。

如果您的安裝量成長到超過授權的CPU單元數量、Astra Control Center會防止您管理新的應用程式。超過容量時會顯示警示。

若要更新現有的評估或完整授權、請參閱 "更新現有授權"。

開始之前

- 存取新安裝的Astra Control Center執行個體。

- 系統管理員角色權限。
- 答 "NetApp授權檔案" （If）。

步驟

1. 登入Astra Control Center UI。
2. 選擇*帳戶*>*授權*。
3. 選擇*新增授權*。
4. 瀏覽至您下載的授權檔案（NLF）。
5. 選擇*新增授權*。

「帳戶>*授權*」頁面會顯示授權資訊、到期日、授權序號、帳戶ID及使用的CPU單位。

如果您擁有評估授權、但並未將資料傳送AutoSupport 至效益分析系統、請務必儲存您的帳戶ID、以免發生Astra Control Center故障時發生資料遺失。

使用Astra Control為環境做好叢集管理準備

在新增叢集之前、您應確保符合下列先決條件。您也應該執行資格檢查、以確保叢集已準備好新增至Astra Control Center、並建立叢集管理的角色。

開始之前

- 確保叢集中的工作節點已設定適當的儲存驅動程式、以便Pod與後端儲存設備互動。
- 您的環境符合 "營運環境需求" 適用於Astra Trident與Astra Control Center。
- 如果您要使用參考私有憑證授權單位（ CA ）的 kubeconfig 檔案來新增叢集、請將下列行新增至 `cluster` kubeconfig 檔案的一節。這可讓 Astra Control 新增叢集：

```
insecure-skip-tls-verify: true
```

- 這是Astra Trident的版本 "由Astra Control Center支援" 已安裝：

  您可以 "部署Astra Trident" 使用 Astra Trident 運算子（手動或使用 Helm 圖表）或 `tridentctl`。在安裝或升級Astra Trident之前、請先檢閱 "支援的前端、後端及主機組態"。

  - * 已設定 Astra Trident 儲存後端 * ：至少必須有一個 Astra Trident 儲存後端 "已設定" 在叢集上。
  - * 已設定的 Astra Trident 儲存類別 * ：至少必須有一個 Astra Trident 儲存類別 "已設定" 在叢集上。如果已設定預設儲存類別、請確定它是唯一具有預設註釋的儲存類別。
  - 已安裝並設定的**Astra Trident Volume Snapshot**控制器與**Volume Snapshot**類別：Volume Snapshot控制器必須是 "已安裝" 以便在Astra Control中建立快照。至少有一個Astra Trident `VolumeSnapshotClass` 過去了 "設定" 由系統管理員執行。
- *可存取的Kubeconfig *：您可以存取 "預設叢集 kubeconfig" 那 "您已在安裝期間進行設定"。
- 《支援》認證：您需要使用支援版的支援版支援系統上設定的支援認證和超級使用者與使用者ID、才能使用Astra Control Center來備份及還原應用程式ONTAP ONTAP ONTAP 。

在flexf2命令列中執行下列命令ONTAP ：

```
export-policy rule modify -vserver <storage virtual machine name>
-policyname <policy name> -ruleindex 1 -superuser sys
export-policy rule modify -vserver <storage virtual machine name>
-policyname <policy name> -ruleindex 1 -anon 65534
```

- *僅限Rancher *：在Rancher環境中管理應用程式叢集時、請在Rancher提供的Kusbeconfig檔案中修改應用程式叢集的預設內容、以使用控制面內容而非Rancher API伺服器內容。如此可減少Rancher API伺服器的負載、並改善效能。

執行資格檢查

執行下列資格檢查、確保您的叢集已準備好新增至Astra控制中心。

步驟

1. 檢查Astra Trident版本。

```
kubectl get tridentversions -n trident
```

如果 Astra Trident 存在、您會看到類似下列的輸出：

```
NAME      VERSION
trident   23.XX.X
```

如果 Astra Trident 不存在、您會看到類似下列的輸出：

```
error: the server doesn't have a resource type "tridentversions"
```

如果未安裝 Astra Trident 、或安裝的版本不是最新版本、則必須先安裝 Astra Trident 的最新版本、才能繼續。請參閱 "Astra Trident文件" 以取得相關指示。

2. 確保Pod正在執行：

```
kubectl get pods -n trident
```

3. 判斷儲存類別是否使用支援的 Astra Trident 驅動程式。置備程式名稱應為 `csi.trident.netapp.io`。請參閱下列範例：

```
kubectl get sc
```

回應範例：

```
NAME                    PROVISIONER             RECLAIMPOLICY
VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
ontap-gold (default)   csi.trident.netapp.io   Delete          Immediate
true                    5d23h
```

建立叢集角色庫比諾圖

您可以選擇性地為 Astra Control Center 建立有限權限或擴充權限管理員角色。這不是 Astra Control Center 設定的必要程序、因為您已將 Kribeconfig 設定為的一部分 "安裝程序"。

如果下列任一情況適用於您的環境、本程序可協助您建立個別的 Kubleconfig ：

- 您想要限制其管理叢集的 Astra Control 權限
- 您使用多個內容範圍、無法使用安裝期間設定的預設 Astra Control Kbeconfig 、或是具有單一內容的受限角色、都無法在您的環境中運作

開始之前

在完成程序步驟之前、請確定您要管理的叢集具備下列項目：

- 已安裝KECV1.23或更新版本
- 利用Astra Control Center來存取您要新增及管理的叢集

  在此程序中、您不需要透過KECBECVL存取執行Astra Control Center的叢集。

- 使用叢集管理權限來管理作用中內容的叢集的作用中KECBEConfig

步驟

1. 建立服務帳戶：

   a. 建立名為的服務帳戶檔案 `astracontrol-service-account.yaml`。

      視需要調整名稱和命名空間。如果在此處進行變更、您應該在下列步驟中套用相同的變更。

```
<strong>astracontrol-service-account.yaml</strong>
```

   +

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: astracontrol-service-account
  namespace: default
```

a. 套用服務帳戶：

```
kubectl apply -f astracontrol-service-account.yaml
```

2. 為要由 Astra Control 管理的叢集建立具有足夠權限的下列叢集角色之一：

   ◦ * 有限叢集角色 * ：此角色包含由 Astra Control 管理叢集所需的最低權限：

展開步驟

i. 建立 `ClusterRole` 例如、 `astra-admin-account.yaml`。

視需要調整名稱和命名空間。如果在此處進行變更、您應該在下列步驟中套用相同的變更。

```
<strong>astra-admin-account.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:

# Get, List, Create, and Update all resources
# Necessary to backup and restore all resources in an app
- apiGroups:
  - '*'
  resources:
  - '*'
  verbs:
  - get
  - list
  - create
  - patch

# Delete Resources
# Necessary for in-place restore and AppMirror failover
- apiGroups:
  - ""
  - apps
  - autoscaling
  - batch
  - crd.projectcalico.org
  - extensions
  - networking.k8s.io
  - policy
  - rbac.authorization.k8s.io
  - snapshot.storage.k8s.io
  - trident.netapp.io
  resources:
  - configmaps
  - cronjobs
  - daemonsets
  - deployments
```

```
  - horizontalpodautoscalers
  - ingresses
  - jobs
  - namespaces
  - networkpolicies
  - persistentvolumeclaims
  - poddisruptionbudgets
  - pods
  - podtemplates
  - podsecuritypolicies
  - replicasets
  - replicationcontrollers
  - replicationcontrollers/scale
  - rolebindings
  - roles
  - secrets
  - serviceaccounts
  - services
  - statefulsets
  - tridentmirrorrelationships
  - tridentsnapshotinfos
  - volumesnapshots
  - volumesnapshotcontents
  verbs:
  - delete

# Watch resources
# Necessary to monitor progress
- apiGroups:
  - ""
  resources:
  - pods
  - replicationcontrollers
  - replicationcontrollers/scale
  verbs:
  - watch

# Update resources
- apiGroups:
  - ""
  - build.openshift.io
  - image.openshift.io
  resources:
  - builds/details
  - replicationcontrollers
  - replicationcontrollers/scale
```

```
  - imagestreams/layers
  - imagestreamtags
  - imagetags
  verbs:
  - update

# Use PodSecurityPolicies
- apiGroups:
  - extensions
  - policy
  resources:
  - podsecuritypolicies
  verbs:
  - use
```

ii. （僅限 OpenShift 叢集）在的結尾處附加下列項目 `astra-admin-account.yaml` 檔案或之後 `# Use PodSecurityPolicies` 區段：

```
# OpenShift security
- apiGroups:
  - security.openshift.io
  resources:
  - securitycontextconstraints
  verbs:
  - use
```

iii. 套用叢集角色：

```
kubectl apply -f astra-admin-account.yaml
```

。* 擴充叢集角色 * ：此角色包含 Astra Control 所管理叢集的擴充權限。如果您使用多個內容範圍、且無法使用安裝期間設定的預設 Astra Control Kbeconfig 、或是具有單一內容的有限角色無法在您的環境中運作、則可以使用此角色：

以下內容 `ClusterRole` 步驟是 Kubernetes 的一般範例。請參閱 Kubernetes 散佈文件、以取得特定於您環境的指示。

展開步驟

i. 建立 `ClusterRole` 例如、 `astra-admin-account.yaml`。

視需要調整名稱和命名空間。如果在此處進行變更、您應該在下列步驟中套用相同的變更。

```
<strong>astra-admin-account.yaml</strong>
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: astra-admin-account
rules:
- apiGroups:
  - '*'
  resources:
  - '*'
  verbs:
  - '*'
- nonResourceURLs:
  - '*'
  verbs:
  - '*'
```

ii. 套用叢集角色：

```
kubectl apply -f astra-admin-account.yaml
```

3. 建立叢集角色與服務帳戶的叢集角色繫結：

a. 建立 `ClusterRoleBinding` 檔案已呼叫 `astracontrol-clusterrolebinding.yaml`。

視需要在建立服務帳戶時調整任何已修改的名稱和命名空間。

```
<strong>astracontrol-clusterrolebinding.yaml</strong>
```

+

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: astracontrol-admin
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: astra-admin-account
subjects:
- kind: ServiceAccount
  name: astracontrol-service-account
  namespace: default
```

a. 套用叢集角色繫結：

```
kubectl apply -f astracontrol-clusterrolebinding.yaml
```

4. 建立並套用權杖密碼：

a. 建立一個稱為的權杖秘密檔案 `secret-astracontrol-service-account.yaml`。

```
<strong>secret-astracontrol-service-account.yaml</strong>
```

```
apiVersion: v1
kind: Secret
metadata:
  name: secret-astracontrol-service-account
  namespace: default
  annotations:
    kubernetes.io/service-account.name: "astracontrol-service-
account"
type: kubernetes.io/service-account-token
```

b. 套用權杖密碼：

```
kubectl apply -f secret-astracontrol-service-account.yaml
```

5. 將權杖密碼新增至服務帳戶、將其名稱新增至 `secrets` Array （以下範例中的最後一行）：

```
kubectl edit sa astracontrol-service-account
```

```
apiVersion: v1
imagePullSecrets:
- name: astracontrol-service-account-dockercfg-48xhx
kind: ServiceAccount
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |

{"apiVersion":"v1","kind":"ServiceAccount","metadata":{"annotations":{},
"name":"astracontrol-service-account","namespace":"default"}}
  creationTimestamp: "2023-06-14T15:25:45Z"
  name: astracontrol-service-account
  namespace: default
  resourceVersion: "2767069"
  uid: 2ce068c4-810e-4a96-ada3-49cbf9ec3f89
secrets:
- name: astracontrol-service-account-dockercfg-48xhx
<strong>- name: secret-astracontrol-service-account</strong>
```

6. 列出取代的服務帳戶機密 `<context>` 正確的安裝環境：

```
kubectl get serviceaccount astracontrol-service-account --context
<context> --namespace default -o json
```

輸出的結尾應類似於下列內容：

```
"secrets": [
{ "name": "astracontrol-service-account-dockercfg-48xhx"},
{ "name": "secret-astracontrol-service-account"}
]
```

中每個元素的索引 `secrets` 陣列開頭為0。在上述範例中、索引為 `astracontrol-service-account-dockercfg-48xhx` 將為0、索引則為 `secret-astracontrol-service-account` 應該是1。在輸出中、記下服務帳戶密碼的索引編號。您在下一個步驟中需要此索引編號。

7. 產生以下的Kbeconfig：

a. 建立 `create-kubeconfig.sh` 檔案：更換 `TOKEN_INDEX` 在下列指令碼開頭、使用正確的值。

```
<strong>create-kubeconfig.sh</strong>
```

```
# Update these to match your environment.
```

```
# Replace TOKEN_INDEX with the correct value
# from the output in the previous step. If you
# didn't change anything else above, don't change
# anything else here.

SERVICE_ACCOUNT_NAME=astracontrol-service-account
NAMESPACE=default
NEW_CONTEXT=astracontrol
KUBECONFIG_FILE='kubeconfig-sa'

CONTEXT=$(kubectl config current-context)

SECRET_NAME=$(kubectl get serviceaccount ${SERVICE_ACCOUNT_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.secrets[TOKEN_INDEX].name}')
TOKEN_DATA=$(kubectl get secret ${SECRET_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.data.token}')

TOKEN=$(echo ${TOKEN_DATA} | base64 -d)

# Create dedicated kubeconfig
# Create a full copy
kubectl config view --raw > ${KUBECONFIG_FILE}.full.tmp

# Switch working context to correct context
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp config use-context
${CONTEXT}

# Minify
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp \
  config view --flatten --minify > ${KUBECONFIG_FILE}.tmp

# Rename context
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  rename-context ${CONTEXT} ${NEW_CONTEXT}

# Create token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-credentials ${CONTEXT}-${NAMESPACE}-token-user \
  --token ${TOKEN}

# Set context to use token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --user ${CONTEXT}-${NAMESPACE}-token
```

```
-user

# Set context to correct namespace
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --namespace ${NAMESPACE}

# Flatten/minify kubeconfig
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  view --flatten --minify > ${KUBECONFIG_FILE}

# Remove tmp
rm ${KUBECONFIG_FILE}.full.tmp
rm ${KUBECONFIG_FILE}.tmp
```

b. 請輸入命令以將其套用至Kubernetes叢集。

```
source create-kubeconfig.sh
```

8. （選用）將Kbeconfig重新命名為有意義的叢集名稱。

```
mv kubeconfig-sa YOUR_CLUSTER_NAME_kubeconfig
```

接下來呢？

現在您已確認已符合先決條件、您已經準備好了 新增叢集。

新增叢集

若要開始管理應用程式、請新增Kubernetes叢集、並將其當作運算資源來管理。您必須為Astra Control Center新增叢集、才能探索Kubernetes應用程式。

我們建議Astra Control Center先管理部署於上的叢集、再將其他叢集新增至Astra Control Center進行管理。需要管理初始叢集、才能傳送Kubmetrics資料和叢集相關資料、以供進行度量和疑難排解。

開始之前

- 新增叢集之前、請先檢閱並執行必要的 必要工作。

步驟

1. 從儀表板或叢集功能表瀏覽：
   - 從「資源摘要」的*「儀表板」中、從「叢集」窗格中選取「新增*」。
   - 在左側導覽區域中、選取*叢集*、然後從「叢集」頁面選取*新增叢集*。
2. 在打開的* Add Cluster-*（添加叢集）窗口中、上傳 `kubeconfig.yaml` 檔案或貼上的內容

`kubeconfig.yaml` 檔案：

- 。 `kubeconfig.yaml` 檔案應*僅包含一個叢集*的叢集認證資料。

如果您自行建立 `kubeconfig` 檔案中、您應該只定義*一個*內容元素。請參閱 "Kubernetes 文件" 以取得有關建立的資訊 `kubeconfig` 檔案：如果您使用為有限的叢集角色建立了Kbeconfig 上述程序請務必在本步驟中上傳或貼上該KECBEConnfig。

3. 提供認證名稱。根據預設、認證名稱會自動填入為叢集名稱。
4. 選擇*下一步*。
5. 選取要用於此Kubernetes叢集的預設儲存類別、然後選取* Next*。

您應該選取以 ONTAP 儲存設備為後盾的 Astra Trident 儲存類別。

6. 檢閱資訊、如果一切看起來都很好、請選取*新增*。

結果

叢集進入*探索*狀態、然後變更為*健全*。您現在正使用Astra Control Center來管理叢集。

在Astra Control Center中新增要管理的叢集之後、可能需要幾分鐘的時間來部署監控操作員。在此之前、通知圖示會變成紅色、並記錄*監控代理程式狀態檢查失敗*事件。您可以忽略這一點、因為當Astra Control Center取得正確狀態時、問題就能解決。如果幾分鐘內仍無法解決問題、請前往叢集並執行 `oc get pods -n netapp-monitoring` 做為起點。您需要查看監控操作員記錄、以偵錯問題。

在 ONTAP 儲存後端啟用驗證

Astra Control Center 提供兩種驗證 ONTAP 後端的模式：

- * 認證型驗證 * ：具有必要權限的 ONTAP 使用者的使用者名稱和密碼。您應該使用預先定義的安全登入角色、例如 admin 或 vsadmin 、以確保與 ONTAP 版本的最大相容性。
- * 憑證型驗證 * ： Astra 控制中心也可以使用安裝在後端的憑證與 ONTAP 叢集通訊。您應該使用用戶端憑證、金鑰和信任的 CA 憑證（如果使用）（建議使用）。

您可以稍後更新現有的後端、將某種驗證類型移至另一種方法。一次只支援一種驗證方法。

啟用認證型驗證

Astra Control Center 需要具備叢集範圍的認證 `admin` 與 ONTAP 後端通訊。您應該使用預先定義的標準角色、例如 `admin`。這可確保與未來 ONTAP 版本的前移相容性、這些版本可能會公開未來 Astra 控制中心版本所使用的功能 API 。

您可以建立自訂安全登入角色、並與 Astra Control Center 搭配使用、但不建議使用。

後端定義範例如下：

```
{
  "version": 1,
  "backendName": "ExampleBackend",
  "storageDriverName": "ontap-nas",
  "managementLIF": "10.0.0.1",
  "dataLIF": "10.0.0.2",
  "svm": "svm_nfs",
  "username": "admin",
  "password": "secret"
}
```

後端定義是唯一以純文字儲存認證的地方。建立或更新後端是唯一需要具備認證知識的步驟。因此、這是僅供管理員使用的操作、由 Kubernetes 或儲存管理員執行。

啟用憑證型驗證

Astra 控制中心可以使用憑證與新的和現有的 ONTAP 後端通訊。您應該在後端定義中輸入下列資訊。

- `clientCertificate`：用戶端憑證。
- `clientPrivateKey`：關聯的私鑰。
- `trustedCACertificate`：可信 CA 證書。如果使用信任的CA、則必須提供此參數。如果未使用信任的CA、則可忽略此問題。

您可以使用下列其中一種類型的憑證：

- 自我簽署的憑證
- 協力廠商憑證

啟用自我簽署憑證的驗證

典型的工作流程包括下列步驟。

步驟

1. 產生用戶端憑證和金鑰。產生時、請將一般名稱（ CN ）設定為 ONTAP 使用者、以驗證為。

```
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout k8senv.key
-out k8senv.pem -subj "/C=US/ST=NC/L=RTP/O=NetApp/CN=<common-name>"
```

2. 安裝用戶端類型的憑證 `client-ca` 以及 ONTAP 叢集上的金鑰。

```
security certificate install -type client-ca -cert-name <certificate-
name> -vserver <vserver-name>
security ssl modify -vserver <vserver-name> -client-enabled true
```

3. 確認 ONTAP 安全登入角色支援憑證驗證方法。

```
security login create -user-or-group-name vsadmin -application ontapi
-authentication-method cert -vserver <vserver-name>
security login create -user-or-group-name vsadmin -application http
-authentication-method cert -vserver <vserver-name>
```

4. 使用產生的憑證測試驗證。以管理 LIF IP 和 SVM 名稱取代 ONTAP Management LIF> 和 <vserver name> 。您必須確保LIF的服務原則設定為 `default-data-management`。

```
curl -X POST -Lk https://<ONTAP-Management-
LIF>/servlets/netapp.servlets.admin.XMLrequest_filer --key k8senv.key
--cert ~/k8senv.pem -d '<?xml version="1.0" encoding="UTF-8"?><netapp
xmlns=http://www.netapp.com/filer/admin version="1.21" vfiler="<vserver-
name>"><vserver-get></vserver-get></netapp>
```

5. 使用從上一步取得的值、在 Astra Control Center UI 中新增儲存後端。

啟用協力廠商憑證的驗證

如果您有協力廠商憑證、您可以使用這些步驟來設定憑證型驗證。

步驟

1. 產生私密金鑰和 CSR ：

```
openssl req -new -newkey rsa:4096 -nodes -sha256 -subj "/" -outform pem
-out ontap_cert_request.csr -keyout ontap_cert_request.key -addext
"subjectAltName = DNS:<ONTAP_CLUSTER_FQDN_NAME>,IP:<ONTAP_MGMT_IP>”
```

2. 將 CSR 傳遞至 Windows CA （協力廠商 CA ）、然後核發簽署的憑證。
3. 下載已簽署的憑證、並將其命名為「 ontap_signed_cert.crt 」
4. 從 Windows CA （協力廠商 CA ）匯出根憑證。
5. 命名此檔案 `ca_root.crt`

   您現在有下列三個檔案：

   - * 私密金鑰 * ： `ontap_signed_request.key` （這是 ONTAP 中伺服器憑證的對應金鑰。安裝伺服器憑證時需要此功能。）
   - * 簽署憑證 * ： `ontap_signed_cert.crt` （這在 ONTAP 中也稱為伺服器憑證 _ 。）
   - * 根 CA 憑證 * ： `ca_root.crt` （這在 ONTAP 中也稱為 _server-ca 憑證 _ 。）
6. 在 ONTAP 中安裝這些憑證。產生及安裝 `server` 和 `server-ca` ONTAP 上的憑證。

展開 **SAMPLE.Yaml**

```
# Copy the contents of ca_root.crt and use it here.

security certificate install -type server-ca

Please enter Certificate: Press <Enter> when done

-----BEGIN CERTIFICATE-----
<certificate details>
-----END CERTIFICATE-----


You should keep a copy of the CA-signed digital certificate for
future reference.

The installed certificate's CA and serial number for reference:

CA:
serial:

The certificate's generated name for reference:


===

# Copy the contents of ontap_signed_cert.crt and use it here. For
key, use the contents of ontap_cert_request.key file.
security certificate install -type server
Please enter Certificate: Press <Enter> when done

-----BEGIN CERTIFICATE-----
<certificate details>
-----END CERTIFICATE-----

Please enter Private Key: Press <Enter> when done

-----BEGIN PRIVATE KEY-----
<private key details>
-----END PRIVATE KEY-----

Enter certificates of certification authorities (CA) which form the
certificate chain of the server certificate. This starts with the
issuing CA certificate of the server certificate and can range up to
the root CA certificate.
Do you want to continue entering root and/or intermediate
```

```
certificates {y|n}: n

The provided certificate does not have a common name in the subject
field.
Enter a valid common name to continue installation of the
certificate: <ONTAP_CLUSTER_FQDN_NAME>

You should keep a copy of the private key and the CA-signed digital
certificate for future reference.
The installed certificate's CA and serial number for reference:
CA:
serial:
The certificate's generated name for reference:


==
# Modify the vserver settings to enable SSL for the installed
certificate

ssl modify -vserver <vserver_name> -ca <CA>  -server-enabled true
-serial <serial number>       (security ssl modify)

==
# Verify if the certificate works fine:

openssl s_client -CAfile ca_root.crt -showcerts -servername server
-connect <ONTAP_CLUSTER_FQDN_NAME>:443
CONNECTED(00000005)
depth=1 DC = local, DC = umca, CN = <CA>
verify return:1
depth=0
verify return:1
write W BLOCK
---
Certificate chain
0 s:
   i:/DC=local/DC=umca/<CA>

-----BEGIN CERTIFICATE-----
<Certificate details>
```

7. 為同一主機建立用戶端憑證、以進行無密碼通訊。Astra 控制中心使用此程序與 ONTAP 通訊。
8. 在 ONTAP 上產生及安裝用戶端憑證：

展開 **SAMPLE.Yaml**

```
# Use /CN=admin or use some other account which has privileges.
openssl req -x509 -nodes -days 1095 -newkey rsa:2048 -keyout
ontap_test_client.key -out ontap_test_client.pem -subj "/CN=admin"

Copy the content of ontap_test_client.pem file and use it in the
below command:
security certificate install -type client-ca -vserver <vserver_name>

Please enter Certificate: Press <Enter> when done

-----BEGIN CERTIFICATE-----
<Certificate details>
-----END CERTIFICATE-----

You should keep a copy of the CA-signed digital certificate for
future reference.
The installed certificate’s CA and serial number for reference:

CA:
serial:
The certificate’s generated name for reference:


==

ssl modify -vserver <vserver_name> -client-enabled true
(security ssl modify)

# Setting permissions for certificates
security login create -user-or-group-name admin -application ontapi
-authentication-method cert -role admin -vserver <vserver_name>

security login create -user-or-group-name admin -application http
-authentication-method cert -role admin -vserver <vserver_name>

==

#Verify passwordless communication works fine with the use of only
certificates:

curl --cacert ontap_signed_cert.crt  --key ontap_test_client.key
--cert ontap_test_client.pem
https://<ONTAP_CLUSTER_FQDN_NAME>/api/storage/aggregates
{
```

```
"records": [
{
"uuid": "f84e0a9b-e72f-4431-88c4-4bf5378b41bd",
"name": "<aggr_name>",
"node": {
"uuid": "7835876c-3484-11ed-97bb-d039ea50375c",
"name": "<node_name>",
"_links": {
"self": {
"href": "/api/cluster/nodes/7835876c-3484-11ed-97bb-d039ea50375c"
}
}
},
"_links": {
"self": {
"href": "/api/storage/aggregates/f84e0a9b-e72f-4431-88c4-
4bf5378b41bd"
}
}
}
],
"num_records": 1,
"_links": {
"self": {
"href": "/api/storage/aggregates"
}
}
}%
```


9. 在 Astra Control Center UI 中新增儲存後端、並提供下列值：
   - * 用戶端憑證 * ： ONTAP 測試用戶端 .pem
   - * 私密金鑰 * ： ontap_test_client.key
   - * 可信 CA 證書 * ： ONTAP 簽署的 _cert.crt

新增儲存後端

您可以將現有ONTAP 的不支援儲存後端新增至Astra Control Center、以管理其資源。

將Astra Control中的儲存叢集管理為儲存後端、可讓您在持續磁碟區（PV）與儲存後端之間建立連結、以及取得額外的儲存指標。

設定認證或憑證驗證資訊之後、您可以將現有的 ONTAP 儲存後端新增至 Astra 控制中心、以管理其資源。

步驟

1. 從左側導覽區域的儀表板中、選取*後端*。

2. 選取*「Add*」。
3. 在「新增儲存設備後端」頁面的「使用現有的」區段中、選取 * ONTAP * 。
4. 選取下列其中一項：
   - * 使用管理員認證 * ：輸入 ONTAP 叢集管理 IP 位址和管理認證。認證資料必須是整個叢集的認證資料。

     您在此處輸入認證的使用者必須擁有 `ontapi` 使用者登入存取方法已在ONTAP 支援的叢集上的「支援系統管理程式」中啟用ONTAP 。如果您打算使用SnapMirror複寫、請套用具有「admin」角色的使用者認證、該角色具有存取方法 `ontapi` 和 `http`、在來源ONTAP 和目的地等叢集上。請參閱 "管理ONTAP 使用者帳戶、請參閱本文檔" 以取得更多資訊。

   - * 使用憑證 * ：上傳憑證 `.pem` 檔案、憑證金鑰 `.key` 檔案、以及選擇性的憑證授權單位檔案。
5. 選擇*下一步*。
6. 確認後端詳細資料、然後選取*管理*。

結果

後端隨即出現在中 `online` 列出摘要資訊。

您可能需要重新整理頁面、以便顯示後端。

新增儲存庫

您可以使用Astra Control UI或來新增儲存區 "Astra Control API"。如果您想要備份應用程式和持續儲存設備、或是想要跨叢集複製應用程式、則必須新增物件存放區資源庫供應商。Astra Control會將這些備份或複製儲存在您定義的物件存放區中。

如果您要將應用程式組態和持續儲存設備複製到同一個叢集、則無需使用Astra Control中的儲存庫。應用程式快照功能不需要儲存庫。

開始之前

- 可從由Astra Control Center管理的叢集存取的儲存庫。
- 庫位認證資料。
- 下列類型的儲存桶：
  - NetApp ONTAP 產品S3
  - NetApp StorageGRID 產品S3
  - Microsoft Azure
  - 一般S3

Amazon Web Services（AWS）和Google Cloud Platform（GCP）使用通用S3儲存區類型。

雖然Astra Control Center支援Amazon S3做為通用S3儲存區供應商、但Astra Control Center可能不支援所有聲稱Amazon S3支援的物件儲存區廠商。

步驟

1. 在左側導覽區域中、選取*鏟斗*。
2. 選取*「Add*」。
3. 選取貯體類型。

   新增儲存庫時、請選擇正確的儲存庫供應商、並提供該供應商的適當認證資料。例如、UI接受NetApp ONTAP S3作為類型並接受StorageGRID 驗證、但這將導致所有未來使用此儲存庫的應用程式備份與還原失敗。

4. 輸入現有的庫位名稱和選用說明。

   庫位名稱和說明會顯示為備份位置、您可以在建立備份時稍後選擇。此名稱也會在保護原則組態期間顯示。

5. 輸入S3端點的名稱或IP位址。
6. 在「選取認證」下、選擇「新增」或「使用現有」索引標籤。
   - 如果您選擇*新增*：
     i. 在Astra Control中輸入認證與其他認證不同的名稱。
     ii. 從剪貼簿貼上內容、輸入存取ID和秘密金鑰。
   - 如果您選擇*使用現有*：
     i. 選取您要搭配儲存區使用的現有認證資料。
7. 選取 `Add`。

   當您新增貯體時、Astra Control會使用預設的貯體指標來標記一個貯體。您建立的第一個儲存區會成為預設儲存區。當您新增儲存庫時、可以稍後決定 "設定另一個預設儲存區"。

接下來呢？

現在您已經登入Astra Control Center並新增叢集、就能開始使用Astra Control Center的應用程式資料管理功能。

- "管理本機使用者和角色"
- "開始管理應用程式"
- "保護應用程式"
- "管理通知"
- "連線Cloud Insights 至"
- "新增自訂TLS憑證"
- "變更預設儲存類別"

如需詳細資訊、請參閱

- "使用Astra Control API"

- "已知問題"

Astra Control Center的常見問題集

如果您只是想要快速回答問題、這個常見問題集就能幫上忙。

總覽

以下各節提供使用Astra Control Center時可能會遇到的其他問題解答。如需進一步的說明、請聯絡astra.feedback@netapp.com

存取Astra Control Center

什麼是**Astra Control URL**？

Astra Control Center使用本機驗證和每個環境的專屬URL。

對於URL、請在瀏覽器中輸入您在安裝Astra Control Center時、於Astra_control_center.yaml自訂資源（CR）檔案的SPEC.astraAddress欄位中所設定的完整網域名稱（FQDN）。電子郵件是您在Astra_control_center.yaml CR的spec.email*欄位中設定的值。

授權

- 我正在使用試用版授權。如何變更為完整授權？ *

您可以從 NetApp 取得 NetApp 授權檔案（ NLF ）、輕鬆變更為完整授權。

步驟

1. 從左側導覽中、選取*帳戶*>*授權*。
2. 在授權總覽中、於授權資訊右側、選取選項功能表。
3. 選取 * 取代 * 。
4. 瀏覽至您下載的授權檔案、然後選取*「Add*（新增*）」。
    - 我正在使用試用版授權。我還能管理應用程式嗎？*

是的、您可以使用評估授權（包括預設安裝的內嵌評估授權）來測試管理應用程式功能。試用版授權與完整版授權之間的功能或功能並無差異；試用版授權的使用壽命更短。請參閱 "授權" 以取得更多資訊。

正在登錄Kubernetes叢集

新增**Astra Control**之後、我需要將工作節點新增至**Kubernetes**叢集。我該怎麼辦？

新的工作者節點可新增至現有的資源池。Astra Control會自動探索這些功能。如果在Astra Control中看不到新節點、請檢查新的工作節點是否執行支援的映像類型。您也可以使用驗證新工作節點的健全狀況 `kubectl get nodes` 命令。

如何正確地取消管理叢集？

1. "從Astra Control取消應用程式管理"。
2. "從Astra Control取消管理叢集"。

從**Astra Control**移除**Kubernetes**叢集之後、應用程式和資料會發生什麼變化？

從Astra Control移除叢集不會對叢集的組態（應用程式和持續儲存）進行任何變更。在該叢集上執行的任何Astra Control快照或應用程式備份都無法還原。由Astra Control所建立的持續儲存備份仍在Astra Control之內、但無法還原。

> 透過任何其他方法刪除叢集之前、請務必先從Astra Control移除叢集。使用另一個工具刪除叢集時、如果叢集仍由Astra Control進行管理、可能會對Astra Control帳戶造成問題。

- 當我取消管理叢集時、 NetApp Astra Trident 是否會自動從叢集解除安裝？ *
  當您從 Astra Control Center 取消管理叢集時、 Astra Trident 不會自動從叢集解除安裝。若要解除安裝Astra Trident、您需要 "請遵循Astra Trident文件中的下列步驟"。

管理應用程式

- Astra Control是否能部署應用程式？*

Astra Control不會部署應用程式。應用程式必須部署在Astra Control之外。

停止從**Astra Control**管理應用程式之後、應用程式會發生什麼事？

將刪除任何現有的備份或快照。應用程式與資料仍可繼續使用。資料管理作業無法用於未受管理的應用程式、或屬於它的任何備份或快照。

- Astra Control能否管理非NetApp儲存設備上的應用程式？*

不可以雖然Astra Control可以探索使用非NetApp儲存設備的應用程式、但它無法管理使用非NetApp儲存設備的應用程式。

- 我應該自行管理 Astra Control 嗎？ *

Astra Control Center 預設不會顯示為您可以管理的應用程式、但您可以使用其他 Astra Control Center 執行個體來備份和還原 Astra Control Center 執行個體。

不健康的 Pod 會影響應用程式管理嗎？ *

否、 Pod 的健全狀況不會影響應用程式管理。

資料管理作業

我的應用程式使用數個**PV**。**Astra Control**是否會擷取這些**PV**的快照與備份？

是的。Astra Control在應用程式上執行的快照作業包括繫結至應用程式PVCS的所有PV快照。

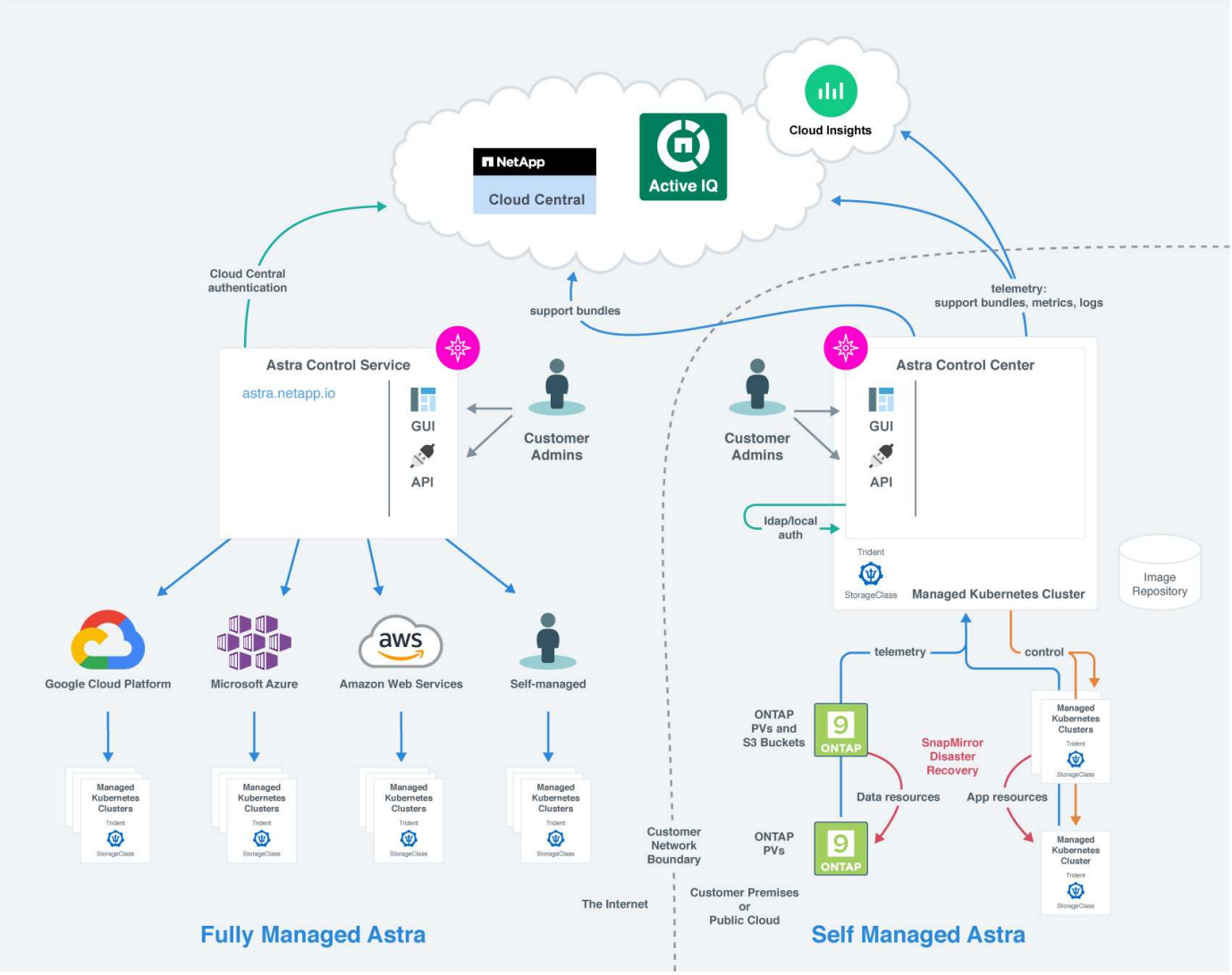
我可以直接透過不同的介面或物件儲存設備來管理**Astra Control**所拍攝的快照嗎？

不可以Astra Control所拍攝的快照與備份、只能透過Astra Control進行管理。

概念

架構與元件

以下概述Astra Control環境的各個元件。



Astra控制元件

- * Kubernetes叢集*：Kubernetes是可攜式、可擴充的開放原始碼平台、可用於管理容器化工作負載與服務、同時促進宣告式組態與自動化。Astra為Kubernetes叢集中託管的應用程式提供管理服務。
- * Astra Trident *：Astra Trident *是完全受支援的開放原始碼儲存資源配置程式、由NetApp維護、可讓您為Docker和Kubernetes所管理的容器化應用程式建立儲存磁碟區。Astra Trident部署於Astra Control Center時、包含已設定ONTAP 的整套儲存後端。
- 儲存後端：
  - Astra Control Service使用下列儲存後端：
    - "適用於Cloud Volumes Service Google Cloud的NetApp解決方案" 或Google持續磁碟做為GKE叢集

的儲存後端

- "Azure NetApp Files" 或Azure託管磁碟做為高效能叢集的儲存後端。
- "Amazon彈性區塊儲存區（EBS）" 或 "Amazon FSX for NetApp ONTAP 產品" 作為EKS叢集的後端儲存選項。

◦ Astra Control Center使用下列儲存後端：

- 不只是部分、不只是部分、更是部分ASA FAS ONTAP AFF 。作為儲存軟體與硬體平台ONTAP 、支援核心儲存服務、支援多種儲存存取傳輸協定、以及快照與鏡射等儲存管理功能。
- Cloud Volumes ONTAP

- * Cloud Insights * ： Cloud Insights 是 NetApp 雲端基礎架構監控工具、可讓您監控 Astra 控制中心所管理的 Kubernetes 叢集的效能與使用率。可將儲存使用量與工作負載建立關聯。Cloud Insights當您在Cloud Insights Astra控制中心啟用「支援不中斷連線」時、遙測資訊會顯示在Astra控制中心UI頁面中。

Astra控制介面

您可以使用不同的介面來完成工作：

- 網路使用者介面（**UI**）：Astra Control Service和Astra Control Center都使用相同的網路型UI來管理、移轉及保護應用程式。也可以使用UI來管理使用者帳戶和組態設定。
- * API*：Astra Control Service和Astra Control Center都使用相同的Astra Control API。使用API、您可以執行與使用UI相同的工作。

Astra Control Center也能讓您管理、移轉及保護在VM環境中執行的Kubernetes叢集。

以取得更多資訊

- "Astra Control Service文件"
- "Astra Control Center文件"
- "Astra Trident文件"
- "使用Astra Control API"
- "本文檔 Cloud Insights"
- "本文檔 ONTAP"

資料保護

瞭解Astra Control Center中可用的資料保護類型、以及如何以最佳方式使用這些類型來保護應用程式。

快照、備份及保護原則

快照和備份都能保護下列類型的資料：

- 應用程式本身
- 與應用程式相關的任何持續資料磁碟區

- 屬於應用程式的任何資源成品

_snapshot是應用程式的時間點複本、儲存在與應用程式相同的已配置磁碟區上。通常速度很快。您可以使用本機快照、將應用程式還原至較早的時間點。快照對快速複製非常實用；快照包括應用程式的所有Kubernetes物件、包括組態檔案。快照可用於複製或還原同一個叢集內的應用程式。

備份_是以快照為基礎。它儲存在外部物件存放區中、因此相較於本機快照、拍攝速度可能較慢。您可以將應用程式備份還原至同一個叢集、也可以將應用程式備份還原至不同的叢集、藉此移轉應用程式。您也可以選擇較長的備份保留期間。由於備份儲存在外部物件存放區中、因此在伺服器故障或資料遺失的情況下、備份通常比快照提供更好的保護。

_protection policy_is是一種保護應用程式的方法、可根據您為該應用程式定義的排程、自動建立快照、備份或兩者。保護原則也可讓您選擇要在排程中保留多少個快照和備份、並設定不同的排程精細度層級。使用保護原則將備份與快照自動化、是確保每個應用程式都能根據組織和服務層級協議（SLA）需求來保護的最佳方式。

> 您必須等到最近進行備份之後、才能獲得完整保護。這很重要、因為備份儲存在遠離持續磁碟區的物件存放區中。如果發生故障或意外、會清除叢集及其相關的持續儲存設備、則需要備份才能恢復。快照無法讓您恢復。

複製

_clon__是應用程式、其組態及其持續資料磁碟區的完全複製。您可以在相同的Kubernetes叢集或其他叢集上手動建立複本。如果您需要將應用程式和儲存設備從一個Kubernetes叢集移至另一個叢集、複製應用程式就很有用。

儲存後端之間的複寫

使用Astra Control、您可以利用NetApp SnapMirror技術的非同步複寫功能、利用低RPO（恢復點目標）和低RTO（恢復時間目標）、為應用程式建立營運不中斷。設定完成後、您的應用程式就能將資料和應用程式變更從一個儲存後端複寫到另一個儲存後端、在同一個叢集或不同叢集之間複寫。

您可以在同一個 ONTAP 叢集或不同 ONTAP 叢集上的兩個 ONTAP VM 之間進行複寫。

Astra Control 會以非同步方式將應用程式快照複本複寫到目的地叢集。複寫程序包括SnapMirror複寫之持續磁碟區中的資料、以及由Astra Control保護的應用程式中繼資料。

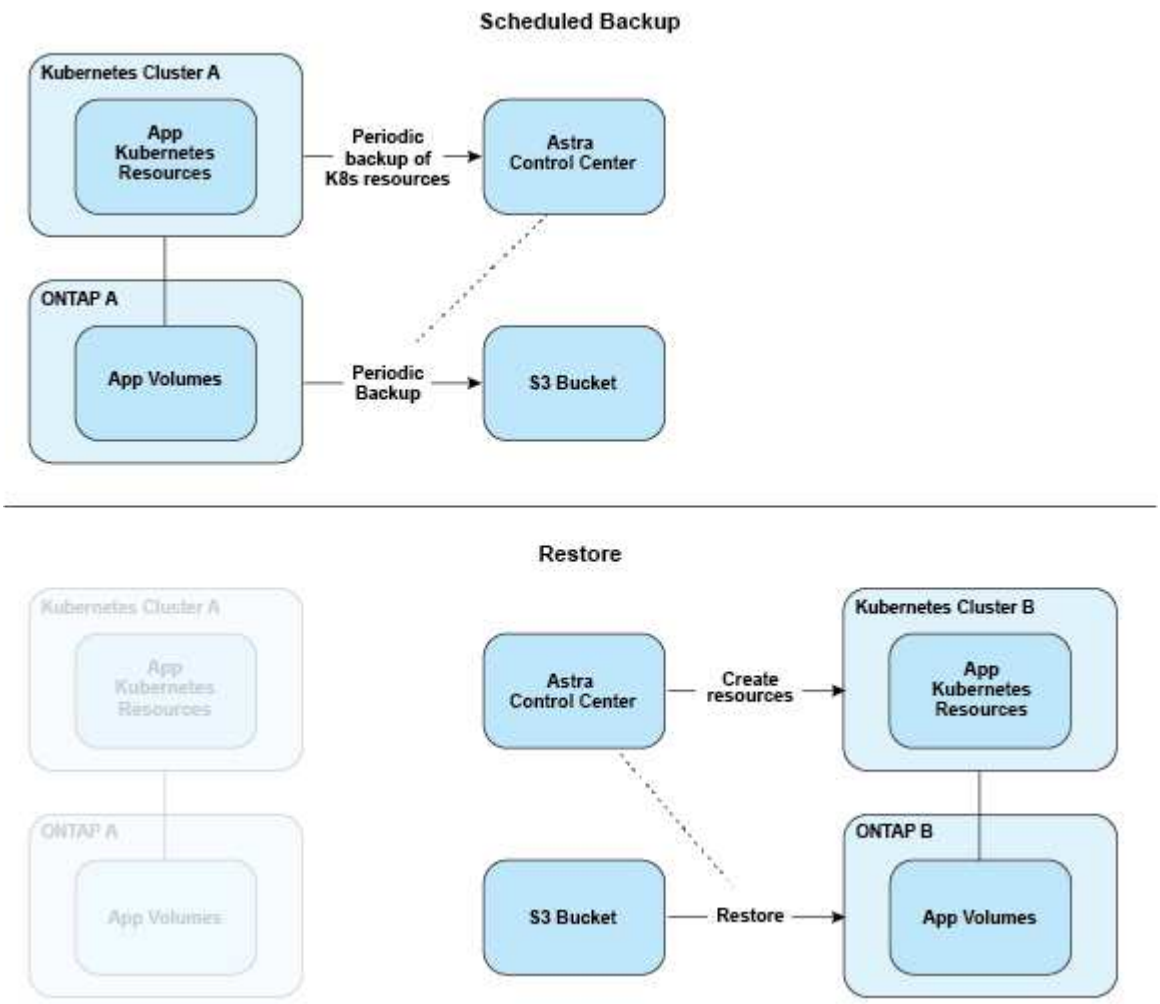
應用程式複寫不同於應用程式備份與還原、方法如下：

- * 應用程式複寫 * ： Astra Control 需要來源和目的地 Kubernetes 叢集（可以是同一個叢集）、才能使用和管理各自的 ONTAP 儲存後端、並將其設定為啟用 NetApp SnapMirror 。Astra Control 會擷取原則導向的應用程式快照、並將其複寫到目的地儲存後端。NetApp SnapMirror 技術用於複寫持續性 Volume 資料。若要容錯移轉、Astra Control可以在目的地Kubernetes叢集上重新建立應用程式物件、並在目的地ONTAP 叢集上重新建立複寫的磁碟區、使複寫的應用程式上線。由於目的地 ONTAP 叢集上已存在持續磁碟區資料、因此 Astra Control 可提供快速的容錯移轉恢復時間。
- * 應用程式備份與還原 * ：當備份應用程式時、 Astra Control 會建立應用程式資料的快照、並將其儲存在物件儲存貯體中。需要還原時、必須將儲存庫中的資料複製到ONTAP 位在該叢集上的持續磁碟區。備份/還原作業不需要次要Kubernetes/ONTAP叢集可供使用和管理、但額外的資料複本可能會導致較長的還原時間。

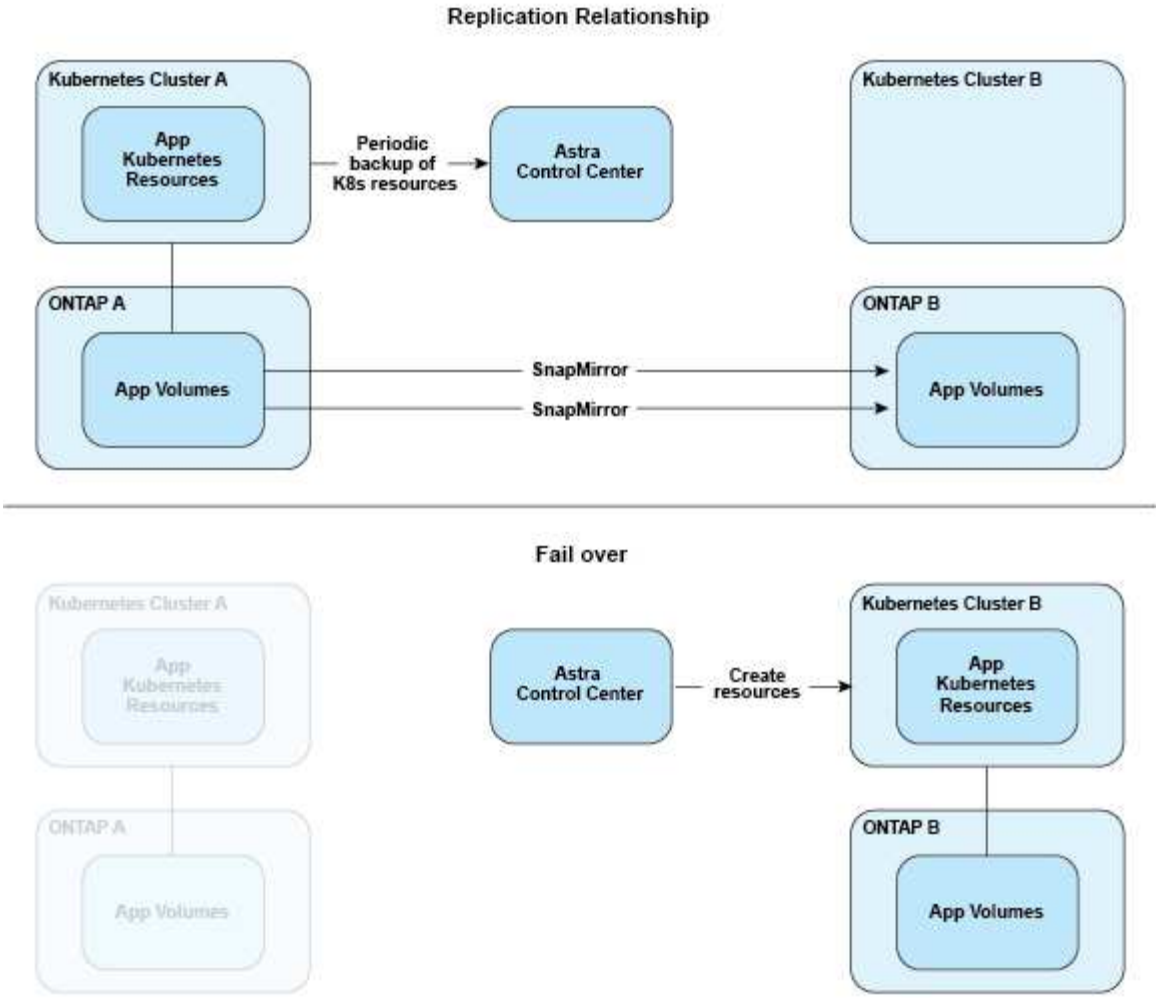
若要瞭解如何複寫應用程式、請參閱 "使用SnapMirror技術將應用程式複寫到遠端系統"。

下列影像顯示排程的備份與還原程序、與複寫程序比較。

備份程序會將資料複製到S3儲存區、並從S3儲存區還原：



另一方面、複寫是透過複寫到 ONTAP 、然後容錯移轉會建立 Kubernetes 資源：



具有過期授權的備份、快照和複本

如果授權過期、您只能在新增或保護的應用程式是另一個 Astra Control Center 執行個體時、新增應用程式或執行應用程式保護作業（例如快照、備份、複本和還原作業）。

授權

當您部署 Astra Control Center 時、系統會安裝內嵌式 90 天試用版授權、適用於 4 、 800 個 CPU 單元。如果您需要更多容量或更長的評估期、或想要升級至完整授權、您可以向 NetApp 取得不同的評估授權或完整授權。

您可以使用下列其中一種方式取得授權：

- 如果您正在評估 Astra Control Center 、並需要不同於內嵌評估授權所含的評估條款、請聯絡 NetApp 以申請不同的評估授權檔案。
- "如果您已購買Astra Control Center、請產生您的NetApp授權檔案（NLF）" 登入 NetApp 支援網站 並瀏覽至系統功能表下的軟體授權。

如需ONTAP 有關支援不支援的詳細資訊、請參閱 "支援的儲存後端"。

請確定您的授權至少能啟用所需的 CPU 單位。如果 Astra Control Center 目前所管理的 CPU 單位數量超過所套用新授權中的可用 CPU 單位、您將無法套用新授權。

評估授權與完整授權

內嵌評估授權隨附全新的 Astra Control Center 安裝。評估授權可在有限（ 90 天）期間內、提供與完整授權相同的功能與功能。評估期結束後、必須取得完整授權才能繼續使用完整功能。

授權過期

如果作用中的 Astra Control Center 授權過期、下列功能的 UI 和 API 功能將無法使用：

- 手動本機快照與備份
- 排程的本機快照與備份
- 從快照或備份還原
- 從快照或目前狀態複製
- 管理新應用程式
- 設定複寫原則

如何計算授權使用量

當您將新叢集新增至Astra Control Center時、除非至少有一個執行於叢集上的應用程式由Astra Control Center管理、否則它不會將使用的授權列入計算。

當您開始管理叢集上的應用程式時、 Astra Control Center 的所有 CPU 單元都會包含在 Astra Control Center 授權使用量中、但 Red Hat OpenShift 叢集節點 CPU 單元則會使用標籤回報 `node-role.kubernetes.io/infra: ""`。

Red Hat OpenShift 基礎架構節點不會使用 Astra Control Center 中的授權。若要將節點標記為基礎架構節點、請套用標籤 `node-role.kubernetes.io/infra: ""` 至節點。

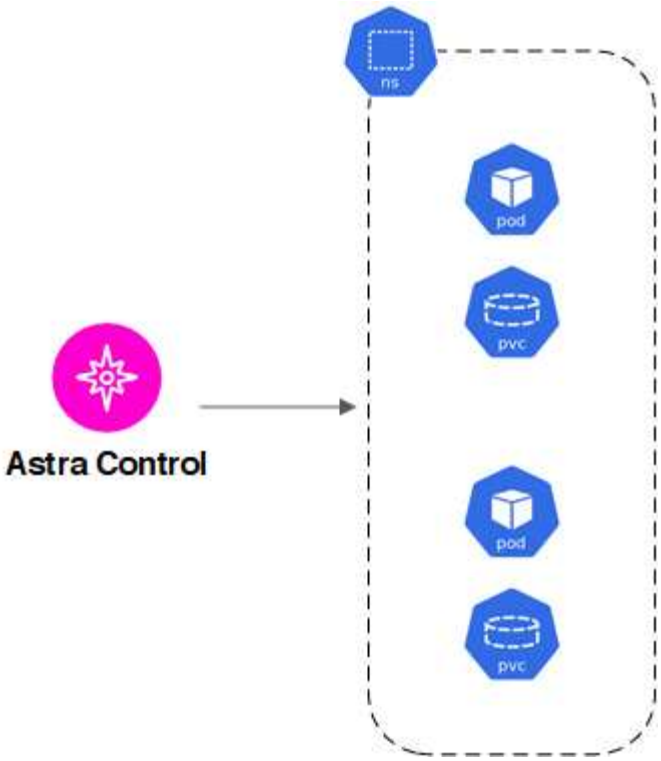
如需詳細資訊、請參閱

- "第一次設定Astra Control Center時、請新增授權"
- "更新現有授權"

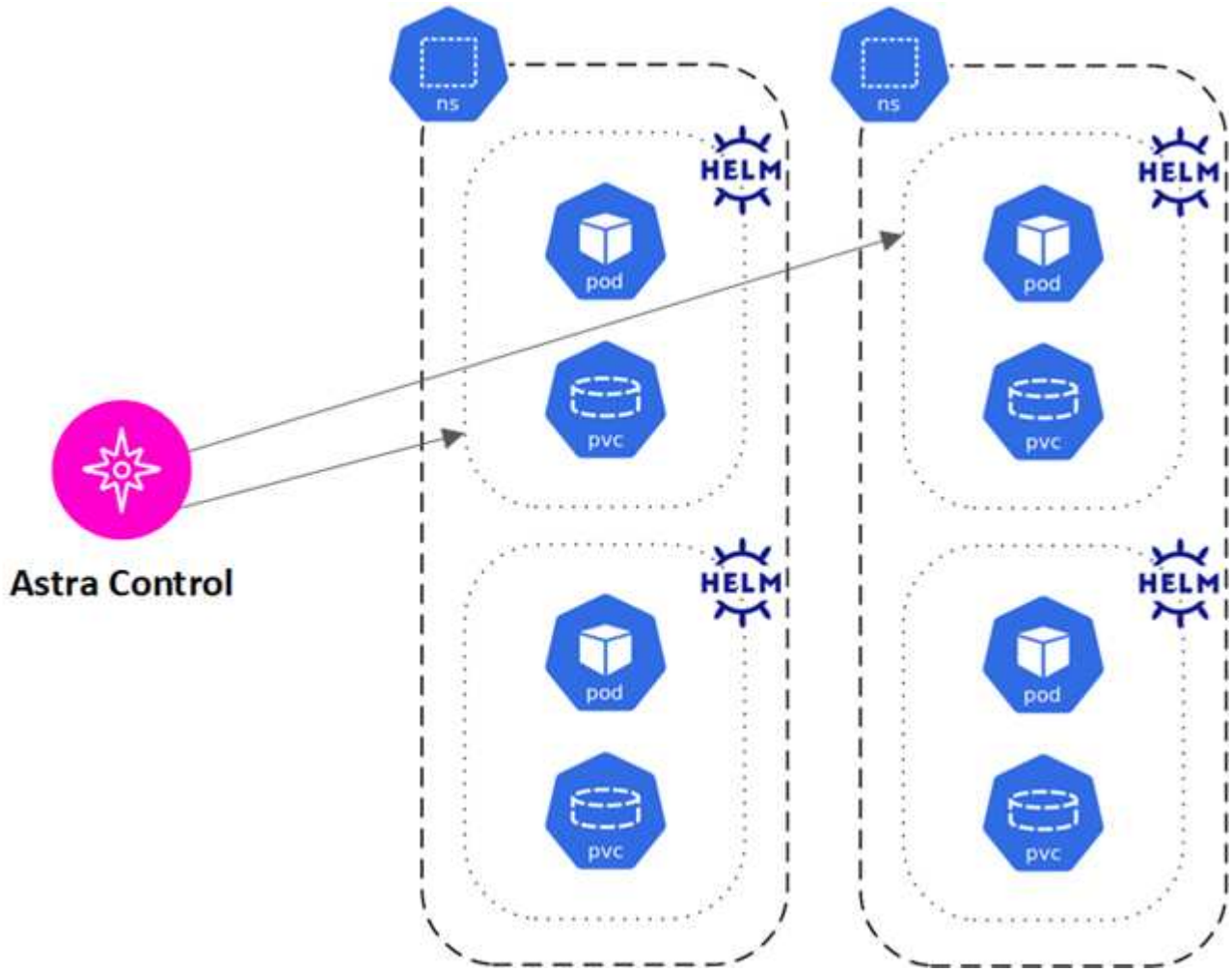
應用程式管理

當Astra Control探索叢集時、這些叢集上的應用程式將無法管理、直到您選擇要管理的方式為止。Astra Control中的託管應用程式可以是下列任一項：

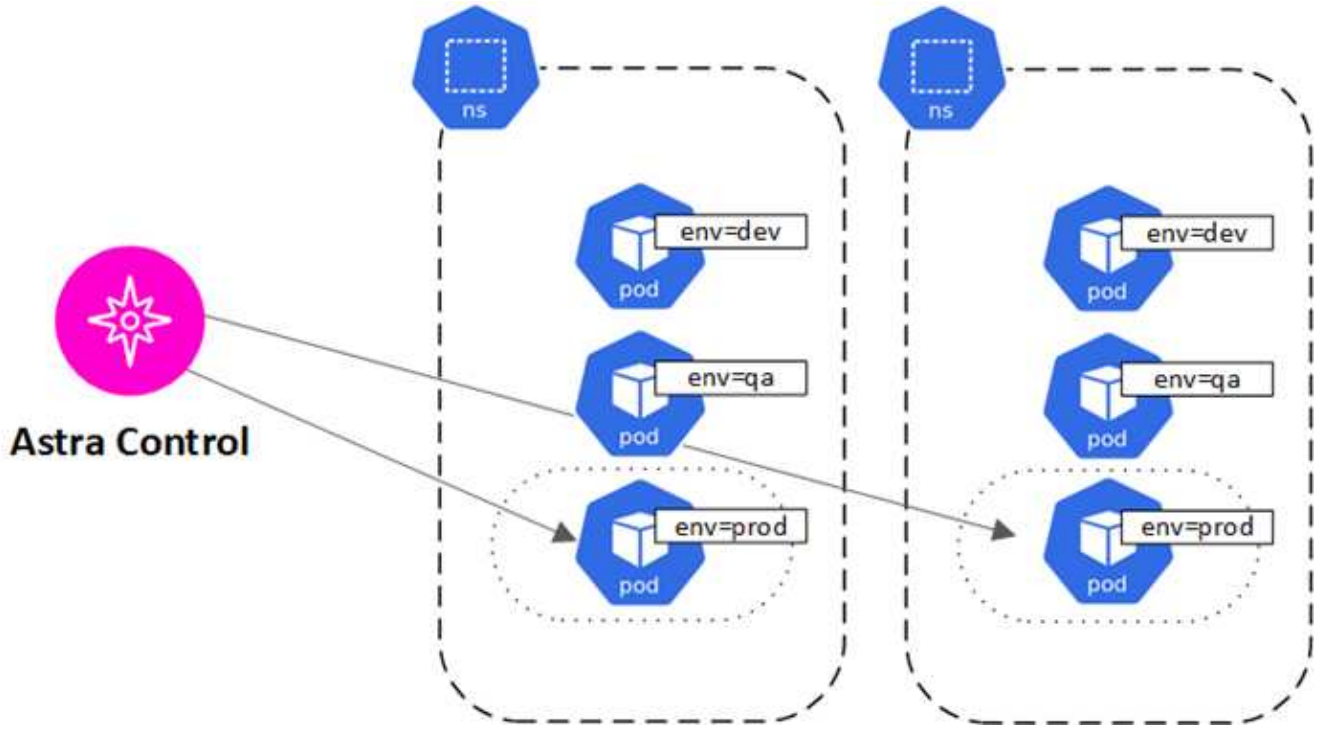
- 命名空間、包括該命名空間中的所有資源



- 部署在一個或多個命名空間內的個別應用程式（本範例使用helm3）



- 一組資源、由一個或多個命名空間內的Kubernetes標籤識別



儲存類別和持續Volume大小

Astra Control Center 支援 NetApp ONTAP 和 Longhorn 作為儲存後端。

總覽

Astra Control Center支援下列項目：

- * Astra Trident 儲存類別以 ONTAP 儲存 * 為後盾：如果您使用的是 ONTAP 後端、 Astra 控制中心可以匯入 ONTAP 後端、以報告各種監控資訊。
- * 以 Longhorn* 為後盾的 CSI 型儲存類別：您可以搭配 Longhorn Container Storage Interface （ CSI ）驅動程式使用 Longhorn 。

Astra Trident 儲存類別應在 Astra Control Center 之外預先設定。

儲存類別

當您將叢集新增至Astra Control Center時、系統會提示您在該叢集上選取先前設定的儲存類別作為預設儲存類別。當持續磁碟區宣告（PVc）中未指定任何儲存類別時、就會使用此儲存類別。預設儲存類別可隨時在Astra Control Center內變更、而任何儲存類別都可隨時在PVc或Helm圖表中指定儲存類別名稱、以供使用。請確定您只為Kubernetes叢集定義單一預設儲存類別。

以取得更多資訊

- "Astra Trident文件"

使用者角色和命名空間

瞭解Astra Control中的使用者角色和命名空間、以及如何使用這些角色和命名空間來控制組織中的資源存取。

使用者角色

您可以使用角色來控制使用者對Astra Control資源或功能的存取。以下是Astra Control的使用者角色：

- *檢視器*可以檢視資源。
- *成員*具有檢視者角色權限、可管理應用程式與叢集、取消管理應用程式、以及刪除快照與備份。
- 「管理」具有「成員角色」權限、可新增及移除擁有者以外的任何其他使用者。
- *擁有者*具有管理員角色權限、可新增及移除任何使用者帳戶。

您可以新增限制給成員或檢視者使用者、將使用者限制為一或多個 [命名空間]。

命名空間

命名空間是可指派給由Astra Control管理之叢集內特定資源的範圍。當您將叢集新增至Astra Control時、Astra Control會探索叢集的命名空間。一旦發現命名空間、就能將其指派為限制給使用者。只有具有該命名空間存取權的成員才能使用該資源。您可以使用命名空間來控制對資源的存取、這種模式對您的組織而言很合理、例如依實體區域或公司內部的部門而定。當您新增限制給使用者時、可以將該使用者設定為只能存取所有命名空間或特定的命名空間集合。您也可以使用命名空間標籤指派命名空間限制。

如需詳細資訊、請參閱

"管理本機使用者和角色"

Pod安全性

Astra Control Center透過pod安全性原則（ASP）和pod安全性許可（Ps）來支援權限限制。這些架構可讓您限制哪些使用者或群組能夠執行容器、以及這些容器可以擁有哪些權限。

部分Kubernetes發佈版本的預設Pod安全性組態可能過於嚴加限制、因此在安裝Astra Control Center時會發生問題。

您可以使用此處提供的資訊和範例來瞭解Astra Control Center所做的Pod安全性變更、並使用pod安全方法來提供所需的保護、而不會干擾Astra Control Center功能。

由Astra Control Center執行的SSA

Astra Control Center 可將下列標籤新增至安裝 Astra 的命名空間（ NetApp-acc 或自訂命名空間）、以及為備份建立的命名空間、藉此強制執行 Pod 安全許可。

```
pod-security.kubernetes.io/enforce: privileged
```

由Astra Control Center安裝的PSPS

當您在Kubernetes 1.23或1.24上安裝Astra Control Center時、會在安裝期間建立數個Pod安全性原則。其中有些是永久性的、有些是在特定作業期間建立、一旦作業完成、就會移除。當主機叢集執行Kubernetes 1.25或更新版本時、Astra Control Center不會嘗試安裝ASP、因為這些版本不支援這些應用程式。

在安裝期間建立PSPS

在Astra Control Center安裝期間、Astra Control Center營運者會安裝自訂的Pod安全性原則A `Role` 物件和 `RoleBinding` 用於支援Astra Control Center命名空間中Astra Control Center服務部署的物件。

新原則和物件具有下列屬性：

```
kubectl get psp

NAME                              PRIV    CAPS          SELINUX    RUNASUSER
FSGROUP     SUPGROUP    READONLYROOTFS   VOLUMES
netapp-astra-deployment-psp    false                 RunAsAny   RunAsAny
RunAsAny    RunAsAny    false            *

kubectl get role -n <namespace_name>

NAME                                        CREATED AT
netapp-astra-deployment-role                2022-06-27T19:34:58Z

kubectl get rolebinding -n <namespace_name>

NAME                                        ROLE
AGE
netapp-astra-deployment-rb                  Role/netapp-astra-deployment-role
32m
```

在備份作業期間建立PSPS

在備份作業期間、Astra Control Center會建立動態Pod安全性原則 `ClusterRole` 物件和 `RoleBinding` 物件：這些支援在個別命名空間中執行的備份程序。

新原則和物件具有下列屬性：

```
kubectl get psp

NAME                             PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-astra-backup             false   DAC_READ_SEARCH
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false            *

kubectl get role -n <namespace_name>

NAME                   CREATED AT
netapp-astra-backup   2022-07-21T00:00:00Z

kubectl get rolebinding -n <namespace_name>

NAME                   ROLE                         AGE
netapp-astra-backup   Role/netapp-astra-backup   62s
```

叢集管理期間建立的**PSPS**

當您管理叢集時、Astra Control Center會在託管叢集中安裝NetApp監控操作員。這位營運者會建立一個Pod安全性原則、a `ClusterRole` 物件和 `RoleBinding` 在Astra Control Center命名空間中部署遙測服務的物件。

新原則和物件具有下列屬性：

```
kubectl get psp

NAME                             PRIV    CAPS
SELINUX    RUNASUSER          FSGROUP     SUPGROUP    READONLYROOTFS
VOLUMES
netapp-monitoring-psp-nkmo      true    AUDIT_WRITE,NET_ADMIN,NET_RAW
RunAsAny   RunAsAny           RunAsAny    RunAsAny    false            *

kubectl get role -n <namespace_name>

NAME                                          CREATED AT
netapp-monitoring-role-privileged             2022-07-21T00:00:00Z

kubectl get rolebinding -n <namespace_name>

NAME                                                ROLE
AGE
netapp-monitoring-role-binding-privileged           Role/netapp-
monitoring-role-privileged             2m5s
```

使用Astra控制中心

開始管理應用程式

您先請 "將叢集新增至Astra Control管理"、您可以在叢集上安裝應用程式（Astra Control之外）、然後前往Astra Control的「應用程式」頁面、定義應用程式及其資源。

應用程式管理需求

Astra Control具備下列應用程式管理需求：

- * 授權 * ：若要使用 Astra Control Center 管理應用程式、您需要內嵌 Astra Control Center 評估授權或完整授權。
- 命名空間：應用程式可以使用Astra Control在單一叢集的一個或多個指定命名空間內定義。應用程式可包含橫跨同一個叢集內多個命名空間的資源。Astra Control不支援跨多個叢集定義應用程式的功能。
- 儲存類別：如果您安裝的應用程式已明確設定儲存類別、而且需要複製應用程式、則複製作業的目標叢集必須具有原本指定的儲存類別。將具有明確設定儲存類別的應用程式複製到沒有相同儲存類別的叢集、將會失敗。
- * Kubernetes資源*：使用未由Astra Control收集之Kubernetes資源的應用程式、可能沒有完整的應用程式資料管理功能。Astra Control會收集下列Kubernetes資源：

ClusterRole	ClusterRoleBinding	ConfigMap
CronJob	CustomResourceDefinition	CustomResource
DaemonSet	DeploymentConfig	HorizontalPodAutoscaler
Ingress	MutatingWebhook	NetworkPolicy
PersistentVolumeClaim	Pod	PodDisruptionBudget
PodTemplate	ReplicaSet	Role
RoleBinding	Route	Secret
Service	ServiceAccount	StatefulSet
ValidatingWebhook		

支援的應用程式安裝方法

Astra Control支援下列應用程式安裝方法：

- 資訊清單檔案：Astra Control支援使用KUbectl從資訊清單檔案安裝的應用程式。例如：

```
kubectl apply -f myapp.yaml
```

- * Helm 3*：如果您使用Helm來安裝應用程式、Astra Control需要Helm版本3。完全支援使用Helm 3（或從Helm 2升級至Helm 3）來管理及複製安裝的應用程式。不支援管理以Helm 2安裝的應用程式。

- * 由操作員部署的應用程式 * ： Astra Control 支援以命名空間範圍運算子安裝的應用程式、這些應用程式通常是以「依值傳遞」而非「依參照傳遞」架構設計。營運者及其安裝的應用程式必須使用相同的命名空間；您可能需要修改部署 YAML 檔案、讓營運者確保這種情況發生。

  以下是一些遵循這些模式的營運者應用程式：

  - "Apache K8ssandra"

    K8ssandra 支援原位還原作業。若要還原新命名空間或叢集的作業、必須先關閉應用程式的原始執行個體。這是為了確保傳遞的對等群組資訊不會導致跨執行個體通訊。不支援複製應用程式。

  - "Jenkins CI"
  - "Percona XtraDB叢集"

  Astra Control可能無法複製以「傳遞參考」架構設計的操作員（例如CockroachDB操作員）。在這些類型的複製作業中、複製的操作員會嘗試從來源操作員參考Kubernetes機密、儘管在複製程序中有自己的新秘密。由於Astra Control不知道來源營運者的Kubernetes機密資料、因此複製作業可能會失敗。

在叢集上安裝應用程式

您先請 "新增叢集" 若要使用Astra Control、您可以在叢集上安裝應用程式或管理現有的應用程式。範圍為一或多個命名空間的任何應用程式都可以管理。

定義應用程式

Astra Control在叢集上探索命名空間之後、您可以定義要管理的應用程式。您可以選擇 管理橫跨一或多個命名空間的應用程式 或 將整個命名空間當作單一應用程式來管理。所有這些都達到資料保護作業所需的精細度。

雖然Astra Control可讓您分別管理階層的兩個層級（命名空間和該命名空間或擴充命名空間中的應用程式）、但最佳實務做法是選擇其中一個。如果在命名空間和應用程式層級同時執行動作、則Astra Control中所採取的動作可能會失敗。

舉例來說、您可能想要為每週有節奏的「Maria」設定備份原則、但您可能需要比這更頻繁地備份「MariaDB」（位於同一個命名空間中）。根據這些需求、您需要分別管理應用程式、而非單一命名空間應用程式。

開始之前

- 將Kubernetes叢集新增至Astra Control。
- 叢集上已安裝一或多個應用程式。 深入瞭解支援的應用程式安裝方法。
- 您新增至Astra Control的Kubernetes叢集上現有的命名空間。
- （選用）任何產品上都有Kubernetes標籤 "支援的Kubernetes資源"。

標籤是可指派給Kubernetes物件以供識別的金鑰/值配對。標籤可讓您更輕鬆地排序、組織及尋找Kubernetes物件。若要深入瞭解Kubernetes標籤、 "請參閱Kubernetes官方文件"。

關於這項工作

- 在開始之前、您也應該瞭解 "管理標準和系統命名空間"。
- 如果您打算在Astra Control中使用多個命名空間搭配應用程式、 "修改具有命名空間限制的使用者角色" 升級至Astra Control Center版本之後、即可支援多個命名空間。
- 如需如何使用Astra Control API管理應用程式的指示、請參閱 "Astra Automation和API資訊"。

應用程式管理選項

- [定義要以應用程式形式管理的資源]
- [定義要以應用程式形式管理的命名空間]

定義要以應用程式形式管理的資源

您可以指定 "Kubernetes是組成應用程式的資源" 您想要使用Astra Control進行管理。定義應用程式可讓您將Kubernetes叢集的元素群組成單一應用程式。此Kubernetes資源集合是根據命名空間和標籤選取器準則來組織。

定義應用程式可讓您更精細地控制要納入Astra Control作業的內容、包括複製、快照和備份。

在定義應用程式時、請確保不將Kubernetes資源納入具有保護原則的多個應用程式中。Kubernetes資源上的保護原則重疊、可能會造成資料衝突。 請參閱範例以瞭解更多資訊。

展開以深入瞭解如何將叢集範圍的資源新增至應用程式命名空間。

除了自動包含的Astra Control之外、您也可以匯入與命名空間資源相關聯的叢集資源。您可以新增規則、其中包含特定群組的資源、種類、版本及選擇性的標籤。如果Astra Control沒有自動包含資源、您可能會想要這麼做。

您無法排除由Astra Control自動包含的任何叢集範圍資源。

您可以新增下列項目 `apiVersions` （與API版本結合的群組）：

資源種類	每個版本（群組+版本）
`ClusterRole`	rbac.authorization.k8s.io/v1
`ClusterRoleBinding`	rbac.authorization.k8s.io/v1
`CustomResource`	apiextensions.k8s.io/v1、apiextensions.k8s.io/v1bet1
`CustomResourceDefinition`	apiextensions.k8s.io/v1、apiextensions.k8s.io/v1bet1
`MutatingWebhookConfiguration`	可受理的registration.k8s.io/v1
`ValidatingWebhookConfiguration`	可受理的registration.k8s.io/v1

步驟

1. 從「應用程式」頁面選取*定義*。
2. 在*定義應用程式*視窗中、輸入應用程式名稱。
3. 在*叢集*下拉式清單中選擇應用程式執行所在的叢集。

4. 從「命名空間」下拉式清單中選擇應用程式的命名空間。

   應用程式可以使用Astra Control在單一叢集上的一個或多個指定命名空間內定義。應用程式可包含橫跨同一個叢集內多個命名空間的資源。Astra Control不支援跨多個叢集定義應用程式的功能。

5. （選用）在每個命名空間中輸入Kubernetes資源的標籤。您可以指定單一標籤或標籤選取器準則（查詢）。

   若要深入瞭解Kubernetes標籤、 "請參閱Kubernetes官方文件"。

6. （選用）選取*新增命名空間*並從下拉式清單中選擇命名空間、即可新增應用程式的其他命名空間。
7. （選用）針對您新增的任何其他命名空間、輸入單一標籤或標籤選取器條件。
8. （可選）要包括除Astra Control自動包含的資源之外的叢集範圍資源、請勾選*包含其他叢集範圍資源*、然後完成下列步驟：
   a. 選取*新增包含規則*。
   b. 群組：從下拉式清單中、選取API資源群組。
   c. 種類：從下拉式清單中、選取物件架構的名稱。
   d. 版本：輸入API版本。
   e. 標籤選取器：選擇性地加入要新增至規則的標籤。此標籤僅用於擷取符合此標籤的資源。如果您未提供標籤、Astra Control會收集為該叢集指定之資源種類的所有執行個體。
   f. 根據您的輸入項目來檢閱建立的規則。
   g. 選取*「Add*」。

      您可以根據需要建立任意數量的叢集範圍資源規則。這些規則會出現在「定義應用程式摘要」中。

9. 選擇*定義*。
10. 選取*定義*之後、視需要為其他應用程式重複此程序。

定義完應用程式之後、應用程式會出現在中 `Healthy` 請在應用程式頁面的應用程式清單中說明。您現在可以複製並建立備份與快照。

您剛新增的應用程式可能會在「受保護的」欄下顯示警告圖示、表示尚未備份且尚未排程備份。

若要查看特定應用程式的詳細資料、請選取應用程式名稱。

若要查看新增至此應用程式的資源、請選取*資源*索引標籤。在「資源」欄中選取資源名稱後的數字、或在「搜尋」中輸入資源名稱、以查看所包含的其他叢集範圍資源。

定義要以應用程式形式管理的命名空間

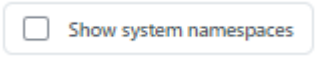
您可以將命名空間中的所有Kubernetes資源新增至Astra Control管理、方法是將該命名空間的資源定義為應用程式。如果您打算以類似的方式、以相同的時間間隔來管理及保護特定命名空間中的所有資源、則此方法較適合個別定義應用程式。

步驟

1. 從「叢集」頁面中選取叢集。
2. 選取「命名空間」索引標籤。
3. 選取包含您要管理之應用程式資源的命名空間「動作」功能表、然後選取*「定義為應用程式*」。

如果要定義多個應用程式、請從命名空間清單中選取、然後選取左上角的*「Actions」（動作）按鈕、然後選取「define as application*」（定義為應用程式*）。這會在個別命名空間中定義多個個別應用程式。如需多命名空間應用程式、請參閱 [定義要以應用程式形式管理的資源]。

選取「顯示系統命名空間」核取方塊、顯示預設不會用於應用程式管理的系統命名空間。



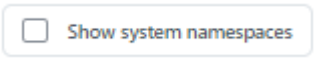
"瞭解更多資訊"。

程序完成後、與命名空間相關聯的應用程式會出現在中 `Associated applications` 欄位。

系統命名空間如何？

Astra Control也會探索Kubernetes叢集上的系統命名空間。我們預設不會顯示這些系統命名空間、因為您很少需要備份系統應用程式資源。

您可以選取「顯示系統命名空間」核取方塊、從「命名空間」索引標籤顯示所選叢集的系統命名空間。



Astra Control Center 預設不會顯示為您可以管理的應用程式、但您可以使用其他 Astra Control Center 執行個體來備份和還原 Astra Control Center 執行個體。

範例：不同版本的個別保護原則

在此範例中、DevOps團隊正在管理「一元化」版本部署。該團隊的叢集有三個執行Ngin像 的Pod。其中兩個Pod專用於穩定版本。第三個pod是用於金箱版本。

DevOps團隊的Kubernetes管理員新增標籤 `deployment=stable` 穩定的釋放Pod。團隊會新增標籤 `deployment=canary` 至準則發行Pod。

該團隊的穩定版本包括每小時快照和每日備份的需求。該準備金版本更為短暫、因此他們想要針對任何標示的項目、建立更具競爭力的短期保護原則 `deployment=canary`。

為了避免可能的資料衝突、管理員將建立兩個應用程式：一個用於「資料」版本、另一個用於「穩定」版本。如此可將兩個Kubernetes物件群組的備份、快照和複製作業分開進行。

如需詳細資訊、請參閱

- "使用Astra Control API"
- "取消管理應用程式"

保護應用程式

保護總覽

您可以使用Astra Control Center為應用程式建立備份、複製、快照及保護原則。備份應用程式有助於您的服務和相關資料盡可能可用；在災難案例中、從備份還原可確保應用程式及其相關資料的完整還原、並將中斷時間降至最低。備份、複製和快照有助於防範勒索軟體、意外資料遺失和環境災難等常見威脅。 "瞭解Astra Control Center中可用的資料保護類型、以及使用時間"。

此外、您也可以將應用程式複寫到遠端叢集、以便做好災難恢復的準備。

應用程式保護工作流程

您可以使用下列範例工作流程、開始保護應用程式。

[一] 保護所有應用程式

為了確保應用程式立即受到保護、 "建立所有應用程式的手動備份"。

[二] 為每個應用程式設定保護原則

若要自動化未來的備份與快照、 "為每個應用程式設定保護原則"。舉例來說、您可以從每週備份和每日快照開始著手、兩個快照均保留一個月。強烈建議使用保護原則來自動化備份與快照、而不要手動備份與快照。

[三] 調整保護原則

隨著應用程式及其使用模式的改變、請視需要調整保護原則、以提供最佳保護。

[四] 將應用程式複寫到遠端叢集

"複寫應用程式" 使用 NetApp SnapMirror 技術將其移至遠端叢集。Astra Control會將Snapshot複寫到遠端叢集、提供非同步的災難恢復功能。

[五] 發生災難時、請使用最新的備份或複寫功能、將應用程式還原至遠端系統

如果發生資料遺失、您可以透過進行恢復 "還原最新的備份" 每個應用程式的第一名。然後您可以還原最新的快照（如果有）。或者、您也可以使用複寫功能來複寫到遠端系統。

利用快照與備份來保護應用程式

使用自動保護原則或以臨機操作的方式、擷取快照與備份資料、以保護所有應用程式。您可以使用Astra Control Center UI或 "Astra Control API" 保護應用程式。

關於這項工作

- * Helm已部署應用程式*：如果您使用Helm來部署應用程式、Astra Control Center需要Helm版本3。完全支援使用Helm 3部署的應用程式管理與複製（或從Helm 2升級至Helm 3）。不支援以Helm 2部署的應用程式。
- （僅限**Openshift**叢集）新增原則：當您建立專案以在OpenShift叢集上裝載應用程式時、專案（

或Kubernetes命名空間）會被指派一個安全性轉換唯一碼。若要啟用Astra Control Center來保護應用程式、並將應用程式移至OpenShift中的其他叢集或專案、您必須新增原則、讓應用程式以任何唯一識別碼的形式執行。例如、下列OpenShift CLI命令會將適當的原則授予WordPress應用程式。

```
oc new-project wordpress
oc adm policy add-scc-to-group anyuid system:serviceaccounts:wordpress
oc adm policy add-scc-to-user privileged -z default -n wordpress
```

您可以執行下列與保護應用程式資料相關的工作：

- [設定保護原則]
- [建立快照]
- [建立備份]
- [檢視快照與備份]
- [刪除快照]
- [取消備份]
- [刪除備份]

設定保護原則

保護原則可在已定義的排程中建立快照、備份或兩者、以保護應用程式。您可以選擇每小時、每天、每週和每月建立快照和備份、也可以指定要保留的複本數量。

如果您需要每小時執行一次以上的備份或快照、您可以 "使用Astra Control REST API建立快照與備份"。

偏移備份和複寫排程、以避免排程重疊。例如、在每小時的最長時間執行備份、並排程複寫以 5 分鐘偏移和 10 分鐘間隔開始。

如果您的應用程式使用以作為後盾的儲存類別 `ontap-nas-economy` 驅動程式、無法使用保護原則。如果您想要排程備份和快照、請移轉至 Astra Control 所支援的儲存類別。

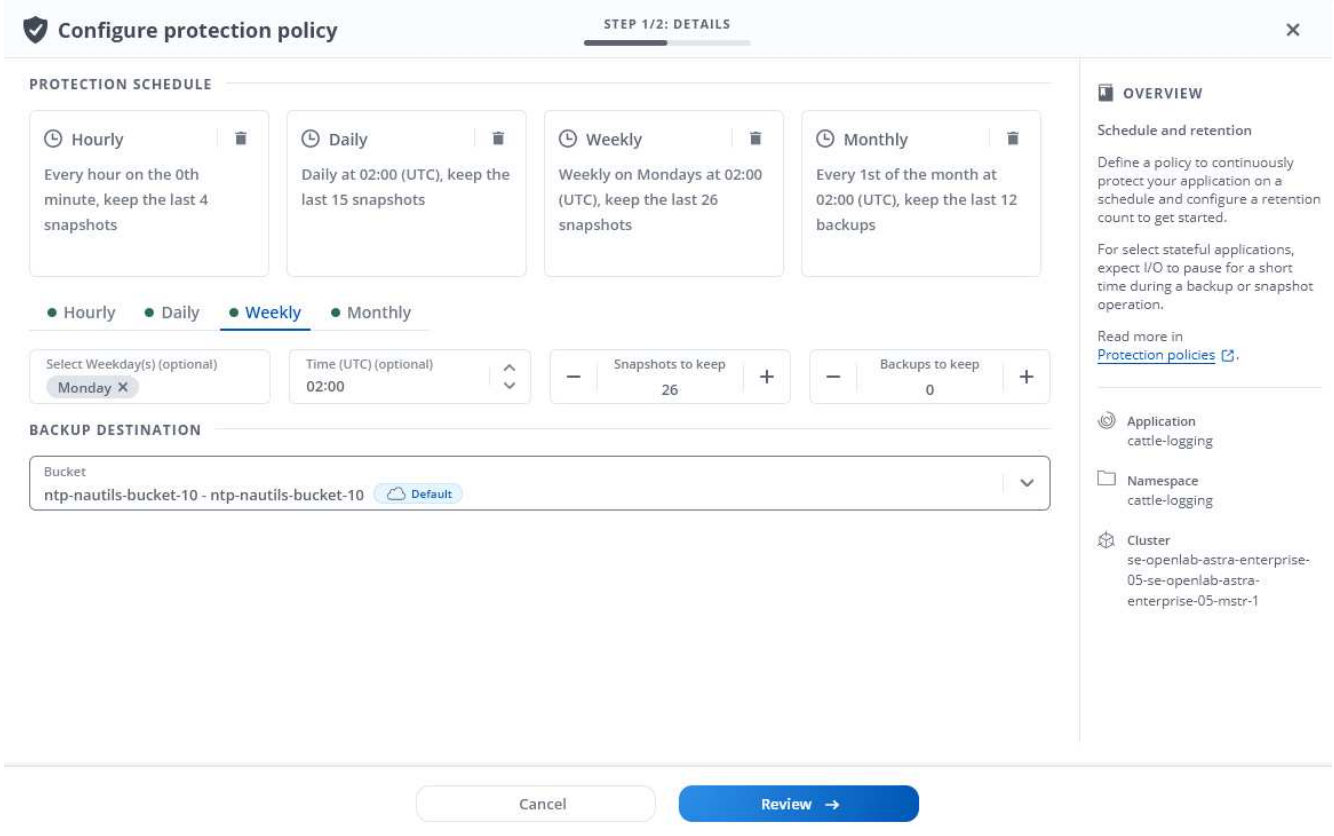
步驟

1. 選取*應用程式*、然後選取應用程式名稱。
2. 選擇*資料保護*。
3. 選取*設定保護原則*。
4. 選擇每小時、每天、每週和每月保留的快照和備份數量、以定義保護排程。

   您可以同時定義每小時、每日、每週及每月排程。在您設定保留層級之前、排程不會變成作用中。

   當您設定備份的保留層級時、可以選擇要儲存備份的儲存區。

   下列範例設定四種保護排程：每小時、每日、每週及每月提供快照與備份。



5. 選擇* Review *。
6. 選取*設定保護原則。*

結果

Astra Control會使用您定義的排程和保留原則來建立和保留快照和備份、以實作資料保護原則。

建立快照

您可以隨時建立隨需快照。

> 如果您的應用程式使用以作為後盾的儲存類別 `ontap-nas-economy` 驅動程式、無法建立快照。使用替代的儲存類別來執行快照。

步驟

1. 選擇*應用程式*。
2. 在所需應用程式*「Actions」（動作）欄的「Options」（選項）功能表中、選取*「Snapshot」（快照）*。
3. 自訂快照的名稱、然後選取*下一步*。
4. 檢閱快照摘要、然後選取* Snapshot *。

結果

快照程序隨即開始。當「資料保護>*快照*」頁面的「狀態」欄中的狀態為「健全」時、快照就會成功。

建立備份

您也可以隨時備份應用程式。

> Astra Control Center中的S3鏟斗未報告可用容量。在備份或複製由Astra Control Center管理的應用程式之前、請先查看ONTAP 資訊庫（英文）或StorageGRID 資訊庫（英文）管理系統中的庫位資訊。

> 如果您的應用程式使用以作為後盾的儲存類別 `ontap-nas-economy` 驅動程式、請確定您已定義 `backendType` 中的參數 "Kubernetes 儲存物件" 值為 `ontap-nas-economy` 執行任何保護作業之前。備份以支援的應用程式 `ontap-nas-economy` 在備份作業完成之前、應用程式會中斷運作、且無法使用。

步驟

1. 選擇*應用程式*。
2. 在所需應用程式*「Actions」（動作）欄的「Options」（選項）功能表中、選取*「Back up」（備份）*。
3. 自訂備份名稱。
4. 選擇是否要從現有的快照備份應用程式。如果選取此選項、您可以從現有快照清單中進行選擇。
5. 從儲存貯體清單中選擇要備份的目的地儲存桶。
6. 選擇*下一步*。
7. 檢閱備份摘要、然後選取*備份*。

結果

Astra Control會建立應用程式的備份。

> 如果您的網路中斷或異常緩慢、備份作業可能會逾時。這會導致備份失敗。

> 如果您需要取消執行中的備份、請依照中的指示操作 [取消備份]。若要刪除備份、請等到備份完成後再依照中的指示進行 [刪除備份]。

> 資料保護作業（複製、備份、還原）及後續持續調整磁碟區大小之後、UI中會顯示新的磁碟區大小、延遲最多20分鐘。資料保護作業只需幾分鐘就能成功完成、您可以使用儲存後端的管理軟體來確認磁碟區大小的變更。

檢視快照與備份

您可以從「資料保護」索引標籤檢視應用程式的快照與備份。

步驟

1. 選取*應用程式*、然後選取應用程式名稱。
2. 選擇*資料保護*。

   快照預設會顯示。

3. 選取*備份*以查看備份清單。

刪除快照

刪除不再需要的排程或隨需快照。

您無法刪除目前正在複寫的快照。

步驟

1. 選取*應用程式*、然後選取託管應用程式的名稱。
2. 選擇*資料保護*。
3. 在所需快照*「Actions」（動作）欄的「Options」（選項）功能表中、選取*「Delete snapshot」（刪除快照）*。
4. 輸入「DELETE」一詞以確認刪除、然後選取*「Yes、Delete snapshot （是、刪除快照）」。

結果

Astra Control會刪除快照。

取消備份

您可以取消進行中的備份。

若要取消備份、備份必須在中 `Running` 州/省。您無法取消中的備份 `Pending` 州/省。

步驟

1. 選取*應用程式*、然後選取應用程式名稱。
2. 選擇*資料保護*。
3. 選擇*備份*。
4. 在所需備份*「Actions」（動作）欄的「**Options**」（選項）功能表中、選取「Cancel*」（取消*）。
5. 輸入「cancel」一詞以確認操作、然後選擇「* Yes、cancel backup*（是、取消備份*）」。

刪除備份

刪除不再需要的排程或隨需備份。

如果您需要取消執行中的備份、請依照中的指示操作 [取消備份]。若要刪除備份、請等到備份完成後再使用這些指示。

步驟

1. 選取*應用程式*、然後選取應用程式名稱。
2. 選擇*資料保護*。
3. 選擇*備份*。
4. 在所需備份*「Actions」（動作）欄的「**Options**」（選項）功能表中、選取「Delete backup*」（刪除備份*）。
5. 輸入「DELETE」一詞以確認刪除、然後選取*「Yes、Delete backup*（是、刪除備份*）」。

結果

Astra Control會刪除備份。

還原應用程式

Astra Control可以從快照或備份還原應用程式。將應用程式還原至同一個叢集時、從現有的快照還原速度會更快。您可以使用Astra Control UI或 "Astra Control API" 以還原應用程式。

關於這項工作

- * 請先保護應用程式 * ：強烈建議您在還原應用程式之前、先拍攝快照或備份應用程式。這可讓您在還原失敗時、從快照或備份進行複製。
- * 檢查目的地 Volume * ：如果您還原至不同的儲存類別、請確定儲存類別使用相同的持續磁碟區存取模式（例如 ReadWriteMany ）。如果目的地持續磁碟區存取模式不同、還原作業將會失敗。例如、如果來源持續性磁碟區使用 rwx 存取模式、請選取無法提供 rwx 的目的地儲存類別、例如 Azure Managed Disks 、 AWS EBS 、 Google Persistent Disk 或 `ontap-san`將導致還原作業失敗。如需持續磁碟區存取模式的詳細資訊、請參閱 "Kubernetes" 文件。
- * 空間需求規劃 * ：當您對使用 NetApp ONTAP 儲存設備的應用程式執行原位還原時、還原的應用程式所使用的空間可能加倍。執行就地還原之後、請從還原的應用程式中移除任何不想要的快照、以釋放儲存空間。
- （僅限**Openshift**叢集）新增原則：當您建立專案以在OpenShift叢集上裝載應用程式時、專案（或Kubernetes命名空間）會被指派一個安全性轉換唯一碼。若要啟用Astra Control Center來保護應用程式、並將應用程式移至OpenShift中的其他叢集或專案、您必須新增原則、讓應用程式以任何唯一識別碼的形式執行。例如、下列OpenShift CLI命令會將適當的原則授予WordPress應用程式。

```
oc new-project wordpress
oc adm policy add-scc-to-group anyuid system:serviceaccounts:wordpress
oc adm policy add-scc-to-user privileged -z default -n wordpress
```

- * Helm 部署的應用程式 * ：完全支援使用 Helm 3 部署的應用程式（或從 Helm 2 升級至 Helm 3 ）。不支援以Helm 2部署的應用程式。

> 在與其他應用程式共用資源的應用程式上執行就地還原作業、可能會產生非預期的結果。在其中一個應用程式上執行就地還原時、應用程式之間共享的任何資源都會被取代。如需詳細資訊、請參閱 此範例。

步驟

1. 選取*應用程式*、然後選取應用程式名稱。
2. 從「動作」欄的「選項」功能表中、選取 * 還原 * 。
3. 選擇還原類型：
   - 還原至原始命名空間：使用此程序可將應用程式就地還原至原始叢集。

     > 如果您的應用程式使用以作為後盾的儲存類別 `ontap-nas-economy` 驅動程式、您必須使用原始儲存類別還原應用程式。如果您要將應用程式還原至相同的命名空間、則無法指定不同的儲存類別。

     i. 選取要用來還原應用程式的快照或備份、此應用程式會將應用程式還原為其舊版。

ii. 選擇*下一步*。

> 如果還原至先前刪除的命名空間、則會在還原程序中建立名稱相同的新命名空間。任何在先前刪除命名空間中擁有管理應用程式權限的使用者、都必須手動還原新重新建立命名空間的權限。

- 還原至新命名空間：使用此程序可將應用程式還原至其他叢集、或從來源還原具有不同命名空間的叢集。

> 您可以使用此程序來執行其中一項 以作為後盾的儲存類別 `ontap-nas` 在同一個叢集 * 或 * 上、將應用程式複製到另一個叢集、並以儲存類別為後盾 `ontap-nas-economy` 驅動程式：

i. 指定還原的應用程式名稱。

ii. 針對您要還原的應用程式、選擇目的地叢集。

iii. 為每個與應用程式相關聯的來源命名空間輸入目的地命名空間。

> Astra Control會在此還原選項中建立新的目的地命名空間。您指定的目的地命名空間不得已存在於目的地叢集上。

iv. 選擇*下一步*。

v. 選取要用來還原應用程式的快照或備份。

vi. 選擇*下一步*。

vii. 請選擇下列其中一項：

- * 使用原始儲存類別還原 * ：除非目標叢集上不存在、否則應用程式會使用原本關聯的儲存類別。在這種情況下、將會使用叢集的預設儲存類別。
- * 使用不同的儲存類別還原 * ：選取目標叢集上存在的儲存類別。所有應用程式磁碟區、無論其最初關聯的儲存類別為何、都會移轉到這個不同的儲存類別、作為還原的一部分。

viii. 選擇*下一步*。

4. 選擇要篩選的任何資源：
   - * 還原所有資源 * ：還原與原始應用程式相關的所有資源。
   - * 篩選資源 * ：指定還原原始應用程式資源子集的規則：
     i. 選擇從還原的應用程式中包含或排除資源。
     ii. 選取 * 新增包含規則 * 或 * 新增排除規則 * 、然後設定規則、在應用程式還原期間篩選正確的資源。您可以編輯或移除規則、然後再次建立規則、直到組態正確為止。

> 若要瞭解如何設定 INCLUDE 及 EXCLUDE 規則、請參閱 [在應用程式還原期間篩選資源]。

5. 選擇*下一步*。
6. 仔細檢閱還原動作的詳細資料、輸入「還原」（如有提示）、然後選取 * 還原 * 。

結果

Astra Control會根據您提供的資訊還原應用程式。如果您就地還原應用程式、現有持續磁碟區的內容會由還原應用程式的持續磁碟區內容取代。

在資料保護作業（複製、備份或還原）及後續持續調整磁碟區大小之後、新的磁碟區大小會在網路UI中顯示、延遲最多20分鐘。資料保護作業只需幾分鐘就能成功完成、您可以使用儲存後端的管理軟體來確認磁碟區大小的變更。

任何具有命名空間限制的成員使用者、都可以使用命名空間名稱/ ID或命名空間標籤、將應用程式複製或還原到同一個叢集上的新命名空間、或是組織帳戶中的任何其他叢集。不過、相同的使用者無法存取新命名空間中的複製或還原應用程式。在複製或還原作業建立新命名空間之後、帳戶管理員/擁有者可以編輯成員使用者帳戶、並更新受影響使用者的角色限制、以便授予新命名空間的存取權。

在應用程式還原期間篩選資源

您可以將篩選規則新增至 "還原" 將指定要從還原的應用程式中包含或排除的現有應用程式資源的作業。您可以根據指定的命名空間、標籤或 GVK （ GroupVersionKind ）來包含或排除資源。

展開以深入瞭解納入和排除案例

- * 您選擇包含原始命名空間的 INCLUDE 規則（原地還原） * ：您在規則中定義的現有應用程式資源將會刪除、並由您用於還原的選定快照或備份中的資源取代。您未在「包括」規則中指定的任何資源將保持不變。
- * 您選擇包含新命名空間的 INCLUDE 規則 * ：使用該規則在還原的應用程式中選取所需的特定資源。您未在「包括」規則中指定的任何資源將不會包含在還原的應用程式中。
- * 您選擇具有原始命名空間的排除規則（就地還原） * ：您指定要排除的資源將不會還原、並保持不變。您未指定排除的資源將會從快照或備份還原。如果對應的 StateSetSet 是篩選資源的一部分、則持續磁碟區上的所有資料都會被刪除並重新建立。
- * 您選取含有新命名空間的排除規則 * ：使用規則選取您要從還原的應用程式中移除的特定資源。您未指定排除的資源將會從快照或備份還原。

規則可以是「包含」或「排除」類型。合併資源包容與排除的規則無法使用。

步驟

1. 在您選擇篩選資源並在「還原應用程式」精靈中選取「包含」或「排除」選項之後、請選取 * 新增「包括」規則 * 或 * 新增排除規則 * 。

   您無法排除 Astra Control 自動包含的任何叢集範圍資源。

2. 設定篩選規則：

   您必須指定至少一個命名空間、標籤或 GVK 。請確保套用篩選規則後保留的任何資源、足以讓還原的應用程式保持正常狀態。

   a. 選取規則的特定命名空間。如果您沒有進行選擇、篩選器將會使用所有命名空間。

如果您的應用程式原本包含多個命名空間、而您將其還原至新命名空間、則即使所有命名空間不包含資源、也會建立這些命名空間。

b. （選用）輸入資源名稱。

c. （選用） * 標籤選取器 * ：包含 A "標籤選取器" 新增至規則。標籤選取器僅用於篩選符合所選標籤的資源。

d. （選用）選取 * 使用設定為篩選資源 * 的 GVK （ GroupVersionKind ）、以取得其他篩選選項。

如果您使用的是 GVK 篩選器、則必須指定版本和種類。

i. （選用） * 群組 * ：從下拉式清單中選取 Kubernetes API 群組。

ii. * 種類 * ：從下拉式清單中、選取要在篩選器中使用的 Kubernetes 資源類型的物件架構。

iii. * 版本 * ：選取 Kubernetes API 版本。

3. 根據您的輸入項目來檢閱建立的規則。

4. 選取*「Add*」。

您可以根據需要建立任意數量的資源、包括和排除規則。這些規則會在您開始作業之前顯示在還原應用程式摘要中。

從 **ONTAP NAS** 經濟型儲存設備移轉至 **ONTAP NAS** 儲存設備

您可以使用 Astra Control "應用程式還原" 或 "應用程式複製" 從以作為後盾的儲存類別移轉應用程式磁碟區的作業 `ontap-nas-economy`（允許有限的應用程式保護選項）、以作為後盾的儲存類別 `ontap-nas` 提供完整的 Astra Control 保護選項。複製或還原作業會移轉使用的 Qtree 型磁碟區 `ontap-nas-economy` 後端到標準磁碟區的備份 `ontap-nas`。Volume 、無論它們是否存在 `ontap-nas-economy` 僅備份或混合、將移轉至目標儲存類別。移轉完成後、保護選項不再受到限制。

應用程式與其他應用程式共用資源的就地還原複雜度

您可以在與其他應用程式共用資源的應用程式上執行就地還原作業、並產生非預期的結果。在其中一個應用程式上執行就地還原時、應用程式之間共享的任何資源都會被取代。

以下是使用 NetApp SnapMirror 複寫進行還原時、造成不必要情況的範例案例：

1. 您可以定義應用程式 `app1` 使用命名空間 `ns1`。
2. 您可以設定的複寫關係 `app1`。
3. 您可以定義應用程式 `app2` （在同一個叢集上）使用命名空間 `ns1` 和 `ns2`。
4. 您可以設定的複寫關係 `app2`。
5. 您可以針對進行反轉複寫 `app2`。這會導致 `app1` 要停用的來源叢集上的應用程式。

使用 **SnapMirror** 技術在儲存設備後端之間複寫應用程式

使用Astra Control、您可以利用NetApp SnapMirror技術的非同步複寫功能、利用低RPO（恢復點目標）和低RTO（恢復時間目標）、為應用程式建立營運不中斷。設定完成後、您

的應用程式就能將資料和應用程式變更從一個儲存後端複寫到另一個儲存後端、在同一個叢集或不同叢集之間複寫。

如需備份 / 還原與複寫之間的比較、請參閱 "資料保護概念"。

您可以在不同的案例中複寫應用程式、例如下列僅限內部部署、混合式和多雲端的案例：

- 內部站台 A 到內部站台 A
- 內部部署站台A到內部部署站台B
- 內部部署至雲端、Cloud Volumes ONTAP 使用不整合技術
- 將Cloud Volumes ONTAP 雲端技術整合至內部部署
- 雲端搭配從功能到雲端（在同一個雲端供應商的不同地區或不同的雲端供應商之間）Cloud Volumes ONTAP

Astra Control可在內部部署叢集、內部部署到雲端（使用Cloud Volumes ONTAP 原地功能）或在雲端之間（Cloud Volumes ONTAP 從地到Cloud Volumes ONTAP 地）複寫應用程式。

> 您可以在相反方向同時複寫不同的應用程式。例如、應用程式A、B、C可以從資料中心1複寫到資料中心2、而應用程式X、Y、Z可以從資料中心2複寫到資料中心1。

使用Astra Control、您可以執行下列與複寫應用程式相關的工作：

- [設定複寫關係]
- [將複寫的應用程式上線至目的地叢集（容錯移轉）]
- [重新同步複寫失敗的情況]
- [反轉應用程式複寫]
- [將應用程式容錯移轉至原始來源叢集]
- [刪除應用程式複寫關係]

複寫先決條件

Astra Control 應用程式複寫需要先滿足下列先決條件、才能開始進行：

- * ONTAP 叢集 * ：
  - * Astra Trident * ： Astra Trident 版本 22.10 或更新版本必須同時存在於使用 ONTAP 作為後端的來源叢集和目的地 Kubernetes 叢集上。
  - * 授權 * ：使用資料保護套件的 ONTAP SnapMirror 非同步授權必須同時在來源和目的地 ONTAP 叢集上啟用。請參閱 "SnapMirror授權概述ONTAP" 以取得更多資訊。
- * 對等 * ：
  - * 叢集與 SVM* ：必須對 ONTAP 儲存設備的後端進行對等處理。請參閱 "叢集與SVM對等概觀" 以取得更多資訊。

    > 確保兩個 ONTAP 叢集之間複寫關係中使用的 SVM 名稱是唯一的。

  - **Astra Trident** 與 **SVM** ：對等的遠端 SVM 必須可用於目的地叢集上的 Astra Trident 。

- * Astra 控制中心 * ：

  "部署 Astra Control Center" 在第三個故障網域或次要站台中進行無縫災難恢復。

  - * 託管叢集 * ：下列叢集必須新增至 Astra Control 、並由 Astra Control 加以管理、最適合在不同的故障網域或站台：
    - 來源 Kubernetes 叢集
    - 目的地 Kubernetes 叢集
    - 相關的 ONTAP 叢集
  - * 使用者帳戶 * ：當您將 ONTAP 儲存後端新增至 Astra 控制中心時、請套用具有「 admin 」角色的使用者認證。此角色具有存取方法 `http` 和 `ontapi` 同時在ONTAP 來源叢集和目的地叢集上啟用。請參閱 "管理ONTAP 使用者帳戶、請參閱本文檔" 以取得更多資訊。
- *Astra Trident / ONTAP 組態 * ： Astra 控制中心要求您至少設定一個儲存後端、以支援來源叢集和目的地叢集的複寫。如果來源叢集和目的地叢集相同、則目的地應用程式應使用不同於來源應用程式的儲存後端、以獲得最佳恢復能力。

Astra Control 複寫支援使用單一儲存類別的應用程式。當您將應用程式新增至命名空間時、請確定該應用程式與命名空間中的其他應用程式具有相同的儲存類別。將 PVC 新增至複寫的應用程式時、請確定新的 PVC 與命名空間中的其他 PVC 具有相同的儲存類別。

設定複寫關係

設定複寫關係涉及下列事項：

- 選擇 Astra Control 拍攝應用程式快照的頻率（包括應用程式的 Kubernetes 資源、以及每個應用程式磁碟區的磁碟區快照）
- 選擇複寫排程（包括Kubernetes資源及持續磁碟區資料）
- 設定拍攝快照的時間

步驟

1. 從Astra Control左側導覽中、選取* Applications*。
2. 選擇 * 資料保護 * > * 複寫 * 標籤。
3. 選取 * 設定複寫原則 * 。或者、從「應用程式保護」方塊中選取「動作」選項、然後選取「設定複寫原則 *」 。
4. 輸入或選取下列資訊：
   - * 目的地叢集 * ：輸入目的地叢集（可以與來源叢集相同） 。
   - * 目的地儲存類別 * ：選取或輸入在目的地 ONTAP 叢集上使用對等 SVM 的儲存類別。最佳實務做法是、目的地儲存類別應指向不同於來源儲存類別的儲存後端。
   - * 複寫類型 * ： `Asynchronous` 目前是唯一可用的複寫類型。
   - 目的地命名空間：為目的地叢集輸入新的或現有的目的地命名空間。
   - （可選）通過選擇* Add namespace*並從下拉列表中選擇命名空間來添加其他命名空間。
   - * 複寫頻率 * ：設定您希望 Astra Control 多久拍攝一次快照並複寫到目的地。

。* 偏移 * ：設定您想要 Astra Control 拍攝快照的小時數頂端的分鐘數。您可能想要使用偏移、使其不與其他排程作業一致。

偏移備份和複寫排程、以避免排程重疊。例如、在每小時的最長時間執行備份、並排程複寫以 5 分鐘偏移和 10 分鐘間隔開始。

5. 選取*下一步*、檢閱摘要、然後選取*儲存*。

一開始、狀態會在第一個排程發生之前顯示「app-mirror」（應用程式鏡射）。

Astra Control 會建立用於複寫的應用程式快照。

6. 若要查看應用程式快照狀態、請選取 * 應用程式 * > * 快照 * 索引標籤。

快照名稱使用的格式 `replication-schedule-<string>`。Astra Control 會保留上次用於複寫的快照。成功完成複寫後、任何較舊的複寫快照都會刪除。

結果

這會建立複寫關係。

Astra Control在建立關係後完成下列行動：

- 在目的地上建立命名空間（如果不存在）
- 在目的地命名空間上建立一個與來源應用程式PVCS對應的PVc。
- 擷取應用程式一致的初始快照。
- 使用初始快照建立持續磁碟區的 SnapMirror 關係。

「 * 資料保護 * 」頁面會顯示複寫關係的狀態和狀態：
<Health status> | <Relationship life cycle state>

例如：
正常 | 已建立

深入瞭解本主題結尾的複寫狀態和狀態。

將複寫的應用程式上線至目的地叢集（容錯移轉）

使用 Astra Control 、您可以將複寫的應用程式容錯移轉至目的地叢集。此程序會停止複寫關係、並在目的地叢集上使應用程式上線。此程序不會停止來源叢集上的應用程式（如果運作正常）。

步驟

1. 從Astra Control左側導覽中、選取* Applications*。
2. 選擇 * 資料保護 * > * 複寫 * 標籤。
3. 從「動作」功能表中、選取 * 容錯移轉 * 。
4. 在「容錯移轉」頁面中、檢閱資訊並選取*容錯移轉*。

結果

容錯移轉程序會執行下列動作：

- 目的地應用程式是根據最新的複寫快照來啟動。
- 來源叢集和應用程式（如果運作正常）不會停止、將會繼續執行。
- 複寫狀態會變更為「容錯移轉」、並在完成後變更為「容錯移轉」。
- 來源應用程式的保護原則會根據容錯移轉時來源應用程式上的排程、複製到目的地應用程式。
- 如果來源應用程式已啟用一或多個還原後執行掛勾、則會為目的地應用程式執行這些執行掛勾。
- Astra Control會在來源叢集和目的地叢集上顯示應用程式及其各自的健全狀況。

重新同步複寫失敗的情況

重新同步作業會重新建立複寫關係。您可以選擇關聯的來源、以保留來源或目的地叢集上的資料。此作業會重新建立SnapMirror關係、以便在選擇的方向開始磁碟區複寫。

此程序會在重新建立複寫之前、停止新目的地叢集上的應用程式。

在重新同步程序期間、生命週期狀態會顯示為「Establishing」。

步驟

1. 從Astra Control左側導覽中、選取* Applications*。
2. 選擇 * 資料保護 * > * 複寫 * 標籤。
3. 從「動作」功能表中、選取 * 重新同步 * 。
4. 在「ResSync（重新同步）」頁面中、選取包含您要保留之資料的來源或目的地應用程式執行個體。

   請謹慎選擇重新同步來源、因為目的地上的資料將被覆寫。

5. 選擇*重新同步*以繼續。
6. 輸入「resSync」以確認。
7. 選取*是、重新同步*以完成。

結果

- 「複寫」頁面會顯示「建立」作為複寫狀態。
- Astra Control會在新的目的地叢集上停止應用程式。
- Astra Control會使用SnapMirror重新同步、在所選方向重新建立持續Volume複寫。
- 「複寫」頁面會顯示更新的關係。

反轉應用程式複寫

這是將應用程式移至目的地儲存後端、同時繼續複寫回原始來源儲存後端的計畫作業。Astra Control 會停止來源應用程式、並在容錯移轉至目的地應用程式之前、將資料複寫到目的地。

在這種情況下、您要交換來源和目的地。

步驟

1. 從Astra Control左側導覽中、選取* Applications*。
2. 選擇 * 資料保護 * > * 複寫 * 標籤。
3. 從「動作」功能表中、選取 * 「反向複寫」 * 。
4. 在「Reverse Replication」（反轉複寫）頁面中、檢閱資訊、然後選取* Reverse Replication*繼續。

結果

下列動作是因為反轉複寫而發生：

- 原始來源應用程式的 Kubernetes 資源會擷取快照。
- 刪除應用程式的Kubernetes資源（保留PVCS和PVs）、即可順利停止原始來源應用程式的Pod。
- 當 Pod 關機之後、應用程式的磁碟區快照就會被擷取和複寫。
- SnapMirror關係中斷、使目的地磁碟區準備好進行讀寫。
- 應用程式的 Kubernetes 資源會從關機前快照還原、並使用原始來源應用程式關機後複寫的 Volume 資料。
- 複寫會以相反方向重新建立。

將應用程式容錯移轉至原始來源叢集

使用 Astra Control 、您可以在容錯移轉作業之後、使用下列作業順序來達成「容錯回復」。在此工作流程中、Astra Control 會先複寫（重新同步）任何應用程式變更回原始來源應用程式、然後再反轉複寫方向。

此程序從已完成容錯移轉至目的地的關係開始、並涉及下列步驟：

- 從容錯移轉狀態開始。
- 重新同步關係。
- 反轉複寫。

步驟

1. 從Astra Control左側導覽中、選取* Applications*。
2. 選擇 * 資料保護 * > * 複寫 * 標籤。
3. 從「動作」功能表中、選取 * 重新同步 * 。
4. 針對容錯回復作業、請選擇容錯移轉應用程式做為重新同步作業的來源（保留任何在容錯移轉後寫入的資料）。
5. 輸入「resSync」以確認。
6. 選取*是、重新同步*以完成。
7. 重新同步完成後、請在「Data Protection（資料保護）」>「Replication（複寫）」索引標籤的「Actions（動作）」功能表中、選取* Reverse replection*（反轉複寫）。
8. 在「Reverse Replication」（反轉複寫）頁面中、檢閱資訊並選取* Reverse Replication*。

結果

這將「重新同步」和「反轉關係」作業的結果結合在一起、以便在原始來源叢集上使應用程式上線、並將複寫恢復至原始目的地叢集。

刪除應用程式複寫關係

刪除關係會產生兩個獨立的應用程式、兩者之間沒有任何關係。

步驟

1. 從Astra Control左側導覽中、選取* Applications*。
2. 選擇 * 資料保護 * > * 複寫 * 標籤。
3. 從「應用程式保護」方塊或關係圖中、選取 * 刪除複寫關係 * 。

結果

刪除複寫關係之後會發生下列動作：

- 如果建立關係、但應用程式尚未在目的地叢集上上線（容錯移轉）、Astra Control會保留初始化期間建立的PVCS、並在目的地叢集上留下「空白」的託管應用程式、並保留目的地應用程式、以保留可能建立的任何備份。
- 如果應用程式已在目的地叢集上線（容錯移轉）、Astra Control會保留PVCS和目的地應用程式。來源和目的地應用程式現在被視為獨立的應用程式。備份排程會保留在兩個應用程式上、但不會彼此關聯。

複寫關係健全狀況狀態和關係生命週期狀態

Astra Control會顯示複寫關係的關係健全狀況、以及複寫關係的生命週期狀態。

複寫關係健全狀況狀態

下列狀態表示複寫關係的健全狀況：

- * 正常 * ：關係正在建立或已建立、最近的快照已成功傳輸。
- 警告：關係可能是容錯移轉或容錯移轉（因此不再保護來源應用程式）。
- 重大
  - 關係正在建立或容錯移轉、最後一次的協調嘗試失敗。
  - 建立關係、最後一次嘗試協調新增的永久虛擬基礎虛擬基礎虛擬基礎虛擬基礎虛擬基礎虛擬基礎虛擬基礎層面時、就會失敗。
  - 這種關係已建立（因此已複寫成功的快照、並可能進行容錯移轉）、但最近的快照無法複寫或無法複寫。

複寫生命週期狀態

下列狀態反映複寫生命週期的不同階段：

- 正在建立：正在建立新的複寫關係。Astra Control會視需要建立命名空間、在目的地叢集的新磁碟區上建立持續磁碟區宣告（PVCS）、並建立SnapMirror關係。此狀態也表示複寫正在重新同步或反轉複寫。
- 已建立：存在複寫關係。Astra Control 會定期檢查 PVC 是否可用、檢查複寫關係、定期建立應用程式快照、並在應用程式中識別任何新的來源 PVC 。如果是、Astra Control會建立資源以將其納入複寫中。
- * 容錯移轉 * ： Astra Control 會中斷 SnapMirror 關係、並從上次成功複寫的應用程式快照中還原應用程式的 Kubernetes 資源。
- * 故障轉移 * ： Astra Control 停止從來源叢集複寫、在目的地上使用最近（成功）複寫的應用程式快照、並還原 Kubernetes 資源。

- 重新同步：Astra Control使用SnapMirror重新同步、將重新同步來源上的新資料重新同步至重新同步目的地。此作業可能會根據同步方向覆寫目的地上的部分資料。Astra Control會停止在目的地命名空間上執行的應用程式、並移除Kubernetes應用程式。在重新同步程序期間、狀態會顯示為「Establing（正在建立）」。
- 反轉：是將應用程式移至目的地叢集、同時繼續複寫回原始來源叢集的計畫性作業。Astra Control會停止來源叢集上的應用程式、將資料複寫到目的地、然後再將應用程式容錯移轉到目的地叢集。在反向複寫期間、狀態會顯示為「Establing（正在建立）」。
- 刪除：
  - 如果複寫關係已建立但尚未容錯移轉、Astra Control會移除複寫期間建立的PVCS、並刪除目的地託管應用程式。
  - 如果複寫已失敗、Astra Control會保留PVCS和目的地應用程式。

複製及移轉應用程式

您可以複製現有的應用程式、在相同的Kubernetes叢集或其他叢集上建立複製的應用程式。當Astra Control複製應用程式時、會建立應用程式組態和持續儲存的複本。

如果您需要將應用程式和儲存設備從一個Kubernetes叢集移至另一個叢集、複製作業將有助於您。例如、您可能想要透過CI/CD傳輸途徑和Kubernetes命名空間來移動工作負載。您可以使用Astra Control Center UI或 "Astra Control API" 複製及移轉應用程式。

開始之前

- * 檢查目的地 Volume * ：如果您複製到不同的儲存類別、請確定儲存類別使用相同的持續磁碟區存取模式（例如 ReadWriteMany ）。如果目的地持續磁碟區存取模式不同、則複製作業將會失敗。例如、如果來源持續性磁碟區使用 rwx 存取模式、請選取無法提供 rwx 的目的地儲存類別、例如 Azure Managed Disks 、 AWS EBS 、 Google Persistent Disk 或 `ontap-san`，將導致克隆操作失敗。如需持續磁碟區存取模式的詳細資訊、請參閱 "Kubernetes" 文件。
- 若要將應用程式複製到不同的叢集、您必須確保包含來源和目的地叢集（如果它們不同）的雲端執行個體具有預設的儲存區。您必須為每個雲端執行個體指派預設儲存區。
- 在複製作業期間、需要IngresClass資源或Webhooks才能正常運作的應用程式、不得在目的地叢集上定義這些資源。

> 在OpenShift環境中進行應用程式複製時、Astra Control Center需要允許OpenShift掛載磁碟區並變更檔案的擁有權。因此、您必須設定ONTAP 一個不中斷的Volume匯出原則、才能執行這些作業。您可以使用下列命令來執行此作業：
>
> 1. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -superuser sys`
> 2. `export-policy rule modify -vserver <storage virtual machine name> -policyname <policy name> -ruleindex 1 -anon 65534`

複製限制

- 明確的儲存類別：如果您部署的應用程式已明確設定儲存類別、而且需要複製應用程式、則目標叢集必須具有原本指定的儲存類別。將具有明確設定儲存類別的應用程式複製到沒有相同儲存類別的叢集、將會失敗。
- * ONTAP NAS 經濟型儲存等級 * ：如果您的應用程式使用的是以為後盾的儲存等級 `ontap-nas-economy` 驅動程式、複製作業的備份部分會中斷運作。在備份完成之前、來源應用程式無法使用。複製作業的還原部分不會中斷營運。

- * Clone與使用者限制*：任何具有命名空間名稱/ ID或命名空間標籤限制的成員使用者、都可以將應用程式複製或還原至同一叢集上的新命名空間、或是組織帳戶中的任何其他叢集。不過、相同的使用者無法存取新命名空間中的複製或還原應用程式。在複製或還原作業建立新命名空間之後、帳戶管理員/擁有者可以編輯成員使用者帳戶、並更新受影響使用者的角色限制、以便授予新命名空間的存取權。
- * Clones使用預設值區段*：在應用程式備份或應用程式還原期間、您可以選擇性地指定區段ID。不過、應用程式複製作業一律會使用已定義的預設儲存區。沒有選項可變更實體複本的儲存區。如果您想要控制所使用的儲存桶、您也可以選擇 "變更庫位預設值" 或執行 "備份" 接著是A "還原" 獨立提供。
- 使用**Jenkins CI**：如果您複製由操作人員部署的Jenkins CI執行個體、則必須手動還原持續性資料。這是應用程式部署模式的限制。
- 使用**S3**鏟斗：Astra Control Center中的S3鏟斗不會報告可用容量。在備份或複製由Astra Control Center管理的應用程式之前、請先查看ONTAP 資訊庫（英文）或StorageGRID 資訊庫（英文）管理系統中的庫位資訊。
- * 使用特定版本的 PostgreSQL * ：同一個叢集中的應用程式複製作業、會以 Bitnami PostgreSQL 11.5.0 圖表持續失敗。若要成功複製、請使用舊版或更新版本的圖表。

OpenShift考量

- 叢集與**OpenShift**版本：如果您在叢集之間複製應用程式、來源與目的地叢集必須是OpenShift的相同發佈版本。例如、如果您從OpenShift 4.7叢集複製應用程式、請使用同樣為OpenShift 4.7的目的地叢集。
- 專案與**UID**：當您建立專案以在OpenShift叢集上裝載應用程式時、專案（或Kubernetes命名空間）會被指派安全性轉換唯一碼。若要啟用Astra Control Center來保護應用程式、並將應用程式移至OpenShift中的其他叢集或專案、您必須新增原則、讓應用程式以任何唯一識別碼的形式執行。例如、下列OpenShift CLI命令會將適當的原則授予WordPress應用程式。

```
oc new-project wordpress
oc adm policy add-scc-to-group anyuid system:serviceaccounts:wordpress
oc adm policy add-scc-to-user privileged -z default -n wordpress
```

步驟

1. 選擇*應用程式*。
2. 執行下列其中一項：
   - 在所需應用程式的*「Actions」（動作）欄中、選取「Options」（選項）功能表。
   - 選取所需應用程式的名稱、然後選取頁面右上角的狀態下拉式清單。
3. 選擇* Clone（克隆）*。
4. 指定實體複本的詳細資料：
   - 輸入名稱。
   - 選擇要複製的目的地叢集。
   - 輸入複本的目的地命名空間。與應用程式相關聯的每個來源命名空間都會對應至您所定義的目的地命名空間。

     Astra Control會在複製作業中建立新的目的地命名空間。您指定的目的地命名空間不得已存在於目的地叢集上。

   - 選擇*下一步*。

。選擇保留與應用程式相關的原始儲存類別、或選擇不同的儲存類別。

您可以將應用程式的儲存類別移轉至原生雲端供應商儲存類別或其他支援的儲存類別、以作為後盾的儲存類別 `ontap-nas` 在同一個叢集上、或是將應用程式複製到另一個叢集、並以儲存類別為後盾 `ontap-nas-economy` 驅動程式：

如果您選取不同的儲存類別、而此儲存類別在還原時並不存在、則會傳回錯誤。

5. 選擇*下一步*。
6. 檢閱有關複本的資訊、然後選取* Clone（複製）*。

結果

Astra Control會根據您提供的資訊來複製應用程式。當有新的應用程式複製時、複製作業會成功完成 `Healthy` 請在「應用程式」頁面上說明。

在複製或還原作業建立新命名空間之後、帳戶管理員/擁有者可以編輯成員使用者帳戶、並更新受影響使用者的角色限制、以便授予新命名空間的存取權。

資料保護作業（複製、備份或還原）及後續持續調整磁碟區大小之後、UI中會顯示新的磁碟區大小、延遲最多20分鐘。資料保護作業只需幾分鐘就能成功完成、您可以使用儲存後端的管理軟體來確認磁碟區大小的變更。

管理應用程式執行掛勾

執行攔截是一種自訂動作、可設定搭配託管應用程式的資料保護作業一起執行。例如、如果您有資料庫應用程式、您可以使用執行掛勾來暫停快照之前的所有資料庫交易、並在快照完成後繼續交易。如此可確保應用程式一致的快照。

執行掛勾的類型

Astra Control支援下列類型的執行掛勾、視執行時間而定：

- 快照前
- 快照後
- 預先備份
- 備份後
- 還原後
- 容錯移轉後

執行攔截篩選器

當您新增或編輯應用程式的執行掛鉤時、您可以將篩選器新增至執行掛鉤、以管理掛鉤將符合的容器。篩選器對於在所有容器上使用相同容器映像的應用程式來說非常實用、但可能會將每個映像用於不同的用途（例如Elasticsearch）。篩選器可讓您建立執行攔截器在某些容器上執行的案例、但不一定所有容器都相同。如果您為單一執行掛勾建立多個篩選器、這些篩選器會與邏輯和運算子結合使用。每個執行掛機最多可有10個作用中篩選器。

您新增至執行掛勾的每個篩選器都會使用規則運算式來比對叢集中的容器。當掛機符合容器時、掛機會在該容器上執行其相關的指令碼。篩選器的規則運算式使用規則運算式2（RE2）語法、不支援建立篩選器、將容器從相符項目清單中排除。如需Astra Control在執行攔截篩選器中支援規則運算式的語法資訊、請參閱 "規則運算式2（RE2）語法支援"。

如果您將命名空間篩選器新增至執行掛勾、而執行還原或複製作業之後執行、且還原或複製來源與目的地位於不同的命名空間、則命名空間篩選器只會套用至目的地命名空間。

關於自訂執行掛勾的重要注意事項

規劃應用程式的執行掛勾時、請考量下列事項。

由於執行掛勾通常會減少或完全停用執行中應用程式的功能、因此您應該一律盡量縮短自訂執行掛勾執行所需的時間。
如果您以相關的執行掛勾開始備份或快照作業、但隨後取消它、則如果備份或快照作業已經開始、仍允許掛勾執行。這表示備份後執行掛勾中使用的邏輯無法假設備份已完成。

- 執行攔截必須使用指令碼來執行動作。許多執行掛勾可以參照相同的指令碼。
- Astra Control需要執行掛勾所使用的指令碼、以執行Shell指令碼的格式寫入。
- 指令碼大小上限為96KB。
- Astra Control使用執行掛勾設定及任何符合條件、來判斷哪些掛勾適用於快照、備份或還原作業。
- 所有執行掛機故障都是軟性故障、即使掛機故障、仍會嘗試其他掛機和資料保護作業。但是、當掛機失敗時、會在*活動*頁面事件記錄中記錄警告事件。
- 若要建立、編輯或刪除執行掛勾、您必須是擁有擁有者、管理員或成員權限的使用者。
- 如果執行掛機執行時間超過25分鐘、掛機將會失敗、並建立傳回代碼為「N/A」的事件記錄項目。任何受影響的快照都會逾時並標示為故障、並會出現一個事件記錄項目、指出逾時時間。
- 對於特殊資料保護作業、所有攔截事件都會產生並儲存在 * 活動 * 頁面事件記錄中。不過、對於排程的資料保護作業、事件記錄中只會記錄攔截故障事件（排程資料保護作業本身所產生的事件仍會記錄下來）。
- 如果 Astra Control Center 將複寫的來源應用程式容錯移轉至目的地應用程式、則在容錯移轉完成後、會針對目的地應用程式執行啟用的任何容錯移轉後執行攔截。

如果您在 Astra Control Center 23.04 上執行還原後掛勾、並將 Astra Control Center 升級至 23.07 、則容錯移轉複寫後將不再執行還原後執行掛勾。您需要為應用程式建立新的容錯移轉後執行掛鉤。或者、您也可以將用於容錯移轉的現有還原後掛鉤作業類型、從「還原後」變更為「容錯移轉後」。

執行順序

執行資料保護作業時、執行掛機事件會依照下列順序發生：

1. 任何適用的自訂操作前執行掛勾都會在適當的容器上執行。您可以視需要建立及執行任意數量的自訂操作前掛勾、但在作業之前執行這些掛勾的順序既不保證也無法設定。
2. 執行資料保護作業。
3. 任何適用的自訂操作後執行掛勾都會在適當的容器上執行。您可以視需要建立及執行任意數量的自訂後置作業掛勾、但在作業後執行這些掛勾的順序並不保證也無法設定。

如果您建立同一類型的多個執行掛勾（例如預先快照）、則無法保證這些掛勾的執行順序。不過、不同類型的掛勾的執行順序也有保證。例如、具有所有不同類型勾點的組態執行順序如下：

1. 執行備份前掛勾
2. 執行快照前掛勾
3. 快照後掛勾已執行
4. 執行備份後掛勾
5. 執行還原後的掛勾

如需此組態的範例、請參閱中表格的案例編號2 [確定掛機是否會執行]。

在正式作業環境中啟用執行攔截指令碼之前、請務必先進行測試。您可以使用'kubecl exec'命令來方便地測試指令碼。在正式作業環境中啟用執行掛勾之後、請測試所產生的快照和備份、以確保它們一致。您可以將應用程式複製到暫用命名空間、還原快照或備份、然後測試應用程式、藉此完成此作業。

確定掛機是否會執行

請使用下表協助判斷您的應用程式是否會執行自訂執行掛勾。

請注意、所有的高階應用程式作業都是執行快照、備份或還原等基本作業之一。視案例而定、複製作業可能由這些作業的各種組合組成、因此複製作業執行的執行掛勾內容會有所不同。

就地還原作業需要現有的快照或備份、因此這些作業不會執行快照或備份掛勾。

如果您先開始、然後取消包含快照的備份、並有相關的執行掛勾、有些掛勾可能會執行、有些則不會執行。這表示備份後執行掛勾無法假設備份已完成。請謹記以下幾點、以相關的執行掛勾來取消備份：

- 備份前和備份後的掛勾一律會執行。
- 如果備份包含新的快照、而且快照已啟動、則會執行快照前和快照後的掛勾。
- 如果在快照開始之前取消備份、則不會執行快照前和快照後掛勾。

案例	營運	現有快照	現有備份	命名空間	叢集	**Snapshot hooks**會執行	備份掛勾運轉	執行還原掛勾	容錯移轉攔截器執行中
1.	複製	n	n	新功能	相同	是	n	是	n
2.	複製	n	n	新功能	與眾不同	是	是	是	n
3.	複製或還原	是	n	新功能	相同	n	n	是	n
4.	複製或還原	n	是	新功能	相同	n	n	是	n
5.	複製或還原	是	n	新功能	與眾不同	n	n	是	n

案例	營運	現有快照	現有備份	命名空間	叢集	**Snapshot hooks**會執行	備份掛勾運轉	執行還原掛勾	容錯移轉攔截器執行中
6.	複製或還原	n	是	新功能	與眾不同	n	n	是	n
7.	還原	是	n	現有的	相同	n	n	是	n
8.	還原	n	是	現有的	相同	n	n	是	n
9.	Snapshot	不適用	不適用	不適用	不適用	是	不適用	不適用	n
10.	備份	n	不適用	不適用	不適用	是	是	不適用	n
11.	備份	是	不適用	不適用	不適用	n	n	不適用	n
12.	容錯移轉	是	不適用	由複寫所建立	與眾不同	n	n	n	是
13.	容錯移轉	是	不適用	由複寫所建立	相同	n	n	n	是

執行攔截範例

請造訪 "NetApp Verda GitHub專案" 可下載熱門應用程式的實際執行掛勾、例如Apache Cassandra和Elasticsearch。您也可以查看範例、瞭解如何建構您自己的自訂執行掛勾。

檢視現有的執行掛勾

您可以檢視應用程式的現有自訂執行掛勾。

步驟

1. 移至*應用程式*、然後選取託管應用程式的名稱。
2. 選取*執行掛勾*索引標籤。

   您可以在結果清單中檢視所有已啟用或已停用的執行掛勾。您可以查看某個掛機的狀態、相符的容器數量、建立時間、以及何時執行（作業前或作業後）。您可以選取 + 勾號名稱旁的圖示、可展開要執行的容器清單。若要檢視與此應用程式執行掛勾相關的事件記錄、請前往*活動*索引標籤。

檢視現有的指令碼

您可以檢視現有上傳的指令碼。您也可以在此頁面上查看使用中的指令碼、以及使用這些指令碼的攔截器。

步驟

1. 前往*帳戶*。
2. 選取*指令碼*索引標籤。

   您可以在此頁面上看到現有上傳指令碼的清單。「使用者」欄會顯示每個指令碼使用的執行掛勾。

新增指令碼

每個執行攔截都必須使用指令碼來執行動作。您可以新增一個或多個執行掛勾可以參考的指令碼。許多執行攔截器都可以參照相同的指令碼、只要變更一個指令碼、就能更新許多執行攔截器。

步驟

1. 前往*帳戶*。
2. 選取*指令碼*索引標籤。
3. 選取*「Add*」。
4. 執行下列其中一項：
   - 上傳自訂指令碼。
     i. 選取*上傳檔案*選項。
     ii. 瀏覽至檔案並上傳。
     iii. 為指令碼指定唯一名稱。
     iv. （選用）輸入其他系統管理員應該知道的任何指令碼附註。
     v. 選取*儲存指令碼*。
   - 從剪貼簿貼入自訂指令碼。
     i. 選取*貼上或類型*選項。
     ii. 選取文字欄位、然後將指令碼文字貼到欄位中。
     iii. 為指令碼指定唯一名稱。
     iv. （選用）輸入其他系統管理員應該知道的任何指令碼附註。
5. 選取*儲存指令碼*。

結果

新指令碼會出現在「指令碼」索引標籤的清單中。

刪除指令碼

如果指令碼不再需要、也不被任何執行掛勾使用、您可以從系統中移除指令碼。

步驟

1. 前往*帳戶*。
2. 選取*指令碼*索引標籤。
3. 選擇要移除的指令碼、然後在*「Actions」（動作）*欄中選取功能表。
4. 選擇*刪除*。

如果指令碼與一個或多個執行掛勾相關聯、則無法使用*刪除*動作。若要刪除指令碼、請先編輯相關的執行掛勾、然後將其與其他指令碼建立關聯。

建立自訂執行掛勾

您可以為應用程式建立自訂執行掛鉤、並將其新增至 Astra Control 。請參閱 [執行攔截範例] 如需攔截範例、您需要擁有擁有擁有者、管理員或成員權限、才能建立執行掛勾。

> 當您建立自訂Shell指令碼作為執行掛勾時、請記得在檔案開頭指定適當的Shell、除非您執行特定命令或提供執行檔的完整路徑。

步驟

1. 選取*應用程式*、然後選取託管應用程式的名稱。
2. 選取*執行掛勾*索引標籤。
3. 選取*「Add*」。
4. 在「勾號詳細資料」區域中：
   a. 從「作業」下拉式功能表中選取作業類型、以判斷掛機應在何時執行。
   b. 輸入掛機的唯一名稱。
   c. （選用）輸入執行期間要傳遞至掛機的任何引數、並在您輸入的每個引數之後按Enter鍵以記錄每個引數。
5. （可選）在*勾選篩選器詳細資料*區域中、您可以新增篩選器來控制執行勾點所在的容器：
   a. 選取*新增篩選器*。
   b. 在*勾選篩選類型*欄中、從下拉式功能表中選擇要篩選的屬性。
   c. 在* Regex*欄中、輸入要做為篩選器的規則運算式。Astra Control使用 "規則運算式2（RE2）regex語法"。

      > 如果您在規則運算式欄位中沒有其他文字的情況下、根據屬性的確切名稱（例如 Pod 名稱）進行篩選、則會執行子字串比對。若要完全符合名稱及名稱、請使用確切的字串相符語法（例如、 `^exact_podname$`）。

   d. 若要新增更多篩選條件、請選取*新增篩選條件*。

      > 執行掛勾的多個篩選器會與邏輯和運算子結合使用。每個執行掛機最多可有10個作用中篩選器。

6. 完成後、選取*下一步*。
7. 在*指令碼*區域中、執行下列其中一項：
   - 新增指令碼。
     i. 選取*「Add*」。
     ii. 執行下列其中一項：
     - 上傳自訂指令碼。
       I. 選取*上傳檔案*選項。
       II. 瀏覽至檔案並上傳。
       III. 為指令碼指定唯一名稱。

IV. （選用）輸入其他系統管理員應該知道的任何指令碼附註。

V. 選取*儲存指令碼*。

- 從剪貼簿貼入自訂指令碼。
  I. 選取*貼上或類型*選項。
  II. 選取文字欄位、然後將指令碼文字貼到欄位中。
  III. 為指令碼指定唯一名稱。
  IV. （選用）輸入其他系統管理員應該知道的任何指令碼附註。

◦ 從清單中選取現有的指令碼。

這會指示執行掛勾使用此指令碼。

8. 選擇*下一步*。
9. 檢閱執行掛機組態。
10. 選取*「Add*」。

檢查執行掛勾的狀態

在快照、備份或還原作業完成執行之後、您可以檢查執行掛勾的狀態、該掛勾是執行作業的一部分。您可以使用此狀態資訊來判斷是否要保留執行掛勾、修改或刪除它。

步驟

1. 選取*應用程式*、然後選取託管應用程式的名稱。
2. 選取*資料保護*索引標籤。
3. 選取* Snapshot*以查看執行中的快照、或選取*備份*以查看執行中的備份。

「掛機狀態」會顯示執行掛機在作業完成後執行的狀態。您可以將游標暫留在狀態上、以取得更多詳細資料。例如、如果快照期間發生執行掛機故障、則將游標移到該快照的掛機狀態上會顯示故障執行掛勾的清單。若要查看每次失敗的原因、您可以查看左側導覽區域的*活動*頁面。

檢視指令碼使用量

您可以在Astra Control Web UI中查看哪些執行掛勾使用特定指令碼。

步驟

1. 選擇*帳戶*。
2. 選取*指令碼*索引標籤。

指令碼清單中的「使用者」欄位包含清單中每個指令碼所使用之掛勾的詳細資料。

3. 在「使用者」欄中選取您感興趣的指令碼資訊。

此時會出現更詳細的清單、其中包含使用指令碼的掛勾名稱、以及設定用來執行的作業類型。

編輯執行掛勾

如果您想要變更執行掛勾的屬性、篩選器或所使用的指令碼、您可以編輯執行掛勾。您需要擁有擁有擁有者、管理員或成員權限、才能編輯執行掛勾。

步驟

1. 選取*應用程式*、然後選取託管應用程式的名稱。
2. 選取*執行掛勾*索引標籤。
3. 在「動作」欄中選取「選項」功能表、以選取您要編輯的掛勾。
4. 選擇*編輯*。
5. 完成每個區段後、請選擇*下一步*進行任何必要的變更。
6. 選擇*保存*。

停用執行掛勾

如果您想要暫時避免在應用程式快照之前或之後執行、可以停用執行掛勾。您需要擁有擁有擁有者、管理員或成員權限、才能停用執行掛勾。

步驟

1. 選取*應用程式*、然後選取託管應用程式的名稱。
2. 選取*執行掛勾*索引標籤。
3. 在「動作」欄中選取「選項」功能表、以顯示您要停用的掛勾。
4. 選擇*停用*。

刪除執行掛勾

如果不再需要執行掛勾、您可以完全移除該掛勾。您需要擁有擁有擁有者、管理員或成員權限、才能刪除執行掛勾。

步驟

1. 選取*應用程式*、然後選取託管應用程式的名稱。
2. 選取*執行掛勾*索引標籤。
3. 在「動作」欄中選取「選項」功能表、以選取您要刪除的掛勾。
4. 選擇*刪除*。
5. 在產生的對話方塊中、輸入「DELETE」進行確認。
6. 選擇*是、刪除執行勾點*。

以取得更多資訊

- "NetApp Verda GitHub專案"

使用 **Astra Control Center** 保護 **Astra Control Center**

為了更有效地確保在執行 Astra Control Center 的 Kubernetes 叢集上的恢復能力、請保護

Astra Control Center 應用程式本身。您可以使用次要 Astra Control Center 執行個體來備份和還原 Astra Control Center 、或是在基礎儲存設備使用 ONTAP 時使用 Astra 複寫。

在這些案例中、 Astra Control Center 的第二個執行個體會部署並設定在不同的故障網域中、並在不同於主要 Astra Control Center 執行個體的第二個 Kubernetes 叢集上執行。第二個 Astra Control 執行個體用於備份主要 Astra Control Center 執行個體、並可能還原主要 Astra Control Center 執行個體。還原或複寫的 Astra Control Center 執行個體將繼續為應用程式叢集應用程式提供應用程式資料管理、並還原這些應用程式的備份和快照存取能力。

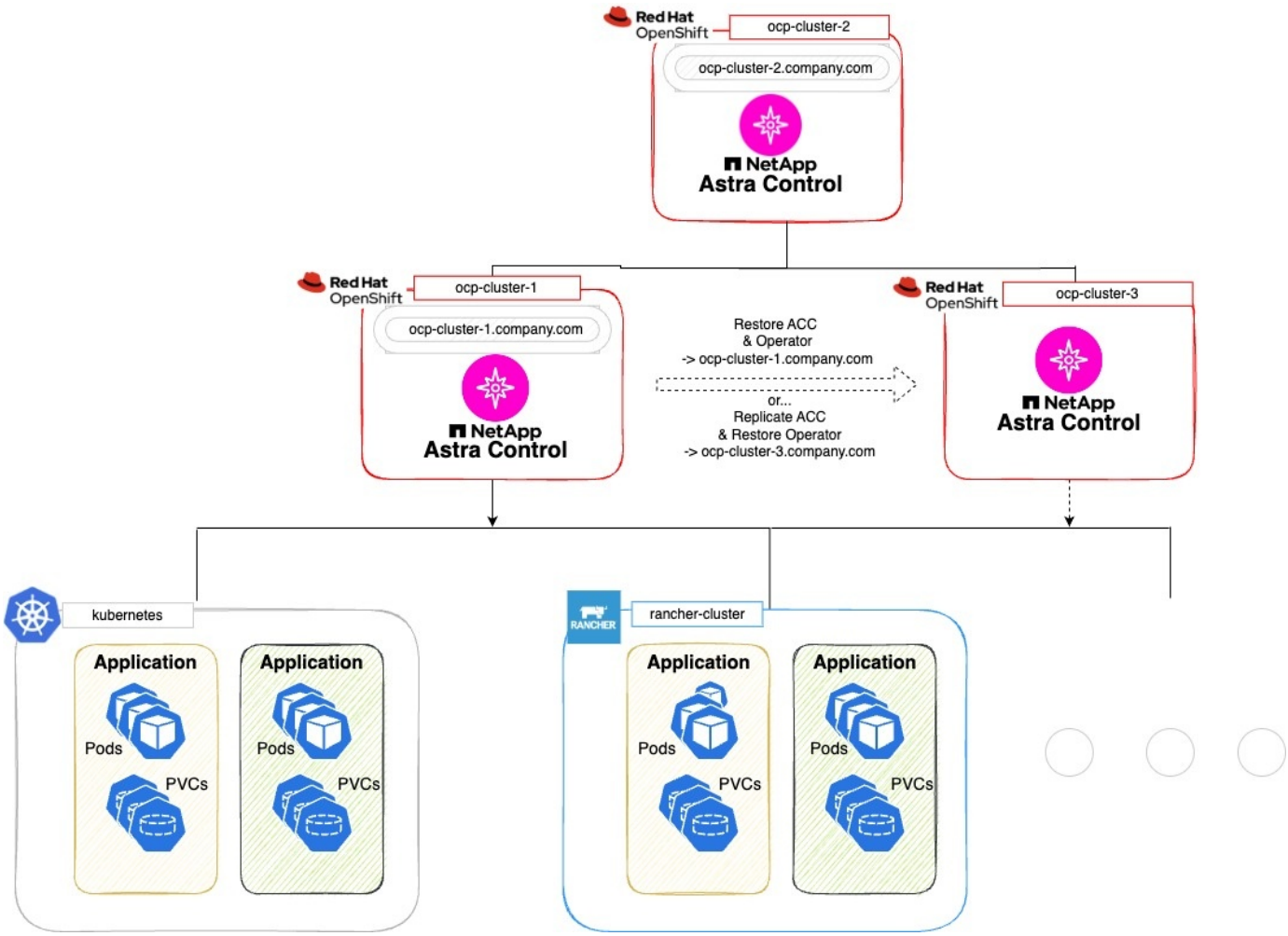
開始之前

在設定 Astra Control Center 的保護方案之前、請務必先執行下列步驟：

- * 執行主要 Astra Control Center 執行個體 * 的 Kubernetes 叢集：此叢集主控管理應用程式叢集的主要 Astra Control Center 執行個體。
- * 第二個 Kubernetes 叢集、其與執行次要 Astra Control Center 執行個體的主叢集具有相同的 Kubernetes 發佈類型 * ：此叢集主控管理主要 Astra Control Center 執行個體的 Astra Control Center 執行個體。
- * 第三個 Kubernetes 叢集、其 Kubernetes 發佈類型與主叢集相同 * ：此叢集將主控已還原或複寫的 Astra Control Center 執行個體。它必須具有目前部署在主要主機上的相同 Astra Control Center 命名空間。例如、如果 Astra Control Center 部署在命名空間中 `netapp-acc` 在來源叢集上、命名空間 `netapp-acc` 目的地 Kubernetes 叢集上的任何應用程式都必須可用、且不得使用。
- * 相容 S3 的儲存庫 * ：每個 Astra Control Center 執行個體都有可存取的 S3 相容物件儲存貯體。
- * 已設定的負載平衡器 * ：負載平衡器為 Astra 提供 IP 位址、而且必須與應用程式叢集和兩個 S3 儲存區建立網路連線。
- * 叢集符合 Astra Control Center 的需求 * ： Astra Control Center 保護所使用的每個叢集都符合 "Astra Control Center 的一般需求"。

關於這項工作

這些程序說明使用 Astra Control Center 還原至新叢集的必要步驟 備份與還原 或 複寫。步驟是根據以下所示的範例組態：



在此範例組態中、會顯示下列內容：

- * 執行主要 Astra Control Center 執行個體 * 的 Kubernetes 叢集：
  - OpenShift 叢集： `ocp-cluster-1`
  - Astra Control Center 主要執行個體： `ocp-cluster-1.company.com`
  - 此叢集可管理應用程式叢集。
- * 第二個 Kubernetes 叢集與執行次要 Astra Control Center 執行個體的主要伺服器具有相同的 Kubernetes 發佈類型 * ：
  - OpenShift 叢集： `ocp-cluster-2`
  - Astra Control Center 次要執行個體： `ocp-cluster-2.company.com`
  - 此叢集將用於備份主要 Astra Control Center 執行個體、或將複寫設定至不同的叢集（在此範例中為 `ocp-cluster-3` 叢集）。
- * 第三個 Kubernetes 叢集、其 Kubernetes 發佈類型與用於還原作業的主要叢集相同 * ：
  - OpenShift 叢集： `ocp-cluster-3`
  - Astra Control Center 第三個執行個體： `ocp-cluster-3.company.com`
  - 此叢集將用於 Astra Control Center 還原或複寫容錯移轉。

理想情況下、應用程式叢集應位於上述影像中 Kubernetes 和 rancher 叢集所描述的三個 Astra Control Center 叢集之外。

圖中未說明：

- 所有叢集都有安裝 Trident 的 ONTAP 後端。
- 在此組態中、 Openshift 叢集使用 MetalLB 做為負載平衡器。
- Snapshot 控制器和 Volume SnapshotClass 也會安裝在所有叢集上、如中所述 "先決條件"。

步驟 **1** 選項：備份與還原 **Astra Control Center**

本程序說明使用備份與還原將 Astra Control Center 還原至新叢集的必要步驟。

在此範例中、 Astra Control Center 一律安裝在 `netapp-acc` 命名空間和運算子會安裝在 `netapp-acc-operator` 命名空間。

雖然未說明、 Astra Control Center 營運者也可以部署在與 Astra CR 相同的命名空間中。

開始之前

- 您已在叢集上安裝主要 Astra Control Center 。
- 您已在不同的叢集上安裝次要 Astra Control Center 。

步驟

1. 從次要 Astra Control Center 執行個體（在上執行）管理主要 Astra Control Center 應用程式和目的地叢集 `ocp-cluster-2` 叢集）：
   a. 登入次要 Astra Control Center 執行個體。
   b. "新增主要 Astra Control Center 叢集" (`ocp-cluster-1`) 。
   c. "新增目的地第三叢集" (`ocp-cluster-3`）用於還原。
2. 在次要 Astra Control Center 上管理 Astra Control Center 和 Astra Control Center 營運者：
   a. 從「應用程式」頁面選取*定義*。
   b. 在 * 定義應用程式 * 視窗中、輸入新的應用程式名稱 (`netapp-acc`) 。
   c. 選擇執行主要 Astra Control Center 的叢集 (`ocp-cluster-1`）從 * 叢集 * 下拉式清單。
   d. 選擇 `netapp-acc` 從 * 命名空間 * 下拉式清單中的 Astra Control Center 命名空間。
   e. 在「叢集資源」頁面上、勾選 * 包括其他叢集範圍的資源 * 。
   f. 選取*新增包含規則*。
   g. 選取這些項目、然後選取 * 新增 * ：
      - 標籤選擇器： ACC-crds
      - 群組： apiextensions.k8s.io
      - 版本： V1,
      - 種類： CustomResourceDefinition

h. 確認應用程式資訊。

i. 選擇*定義*。

選取 * 定義 * 後、請重複操作員的定義應用程式程序程序 `netapp-acc-operator`）、然後選取 `netapp-acc-operator` 定義應用程式精靈中的命名空間。

3. 備份 Astra Control Center 和駕駛員：

a. 在次要 Astra Control Center 上、選取應用程式索引標籤、瀏覽至應用程式頁面。

b. "備份" Astra Control Center 應用程式 (`netapp-acc`）。

c. "備份" 營運者 (`netapp-acc-operator`）。

4. 在您備份 Astra Control Center 和營運者之後、請透過模擬災難恢復（ DR ）案例 "解除安裝 Astra Control Center" 從主叢集。

您將將 Astra Control Center 還原至新叢集（本程序所述的第三個 Kubernetes 叢集）、並將相同的 DNS 作為新安裝 Astra Control Center 的主要叢集。

5. 使用次要 Astra Control Center 、 "還原" Astra Control Center 應用程式從其備份中的主要執行個體：

a. 選取 * 應用程式 * 、然後選取 Astra Control Center 應用程式的名稱。

b. 從「動作」欄的「選項」功能表中、選取 * 還原 * 。

c. 選擇 * 還原至新命名空間 * 作為還原類型。

d. 輸入還原名稱 (`netapp-acc`）。

e. 選擇目的地第三叢集 (`ocp-cluster-3`）。

f. 更新目的地命名空間、使其與原始命名空間相同。

g. 在「還原來源」頁面上、選取將用作還原來源的應用程式備份。

h. 選取 * 使用原始儲存類別還原 * 。

i. 選取 * 還原所有資源 * 。

j. 檢閱還原資訊、然後選取 * 還原 * 以開始還原程序、將 Astra Control Center 還原至目的地叢集 (`ocp-cluster-3`）。應用程式進入時即完成還原 `available` 州/省。

6. 在目的地叢集上設定 Astra Control Center ：

a. 開啟終端機、並使用 kubeconfig 連線至目的地叢集 (`ocp-cluster-3`）、其中包含已還原的 Astra Control Center 。

b. 確認 `ADDRESS` Astra Control Center 組態中的欄會參照主要系統的 DNS 名稱：

```
kubectl get acc -n netapp-acc
```

回應：

```
NAME  UUID                                   VERSION    ADDRESS
READY
astra 89f4fd47-0cf0-4c7a-a44e-43353dc96ba8 23.07.0-24 ocp-cluster-
1.company.com           True
```

a. 如果是 `ADDRESS` 上述回應中的欄位沒有主要 Astra Control Center 執行個體的 FQDN 、請更新組態以參考 Astra Control Center DNS ：

```
kubectl edit acc -n netapp-acc
```

i. 變更 `astraAddress` 低於 `spec:` 至 FQDN (`ocp-cluster-1.company.com` 在此範例中）的主要 Astra Control Center 執行個體。

ii. 儲存組態。

iii. 確認地址已更新：

```
kubectl get acc -n netapp-acc
```

b. 前往 還原 Astra Control Center 操作員 本文件的一節、以完成還原程序。

步驟 **1** 選項：使用複寫保護 **Astra Control Center**

本程序說明設定所需的步驟 "Astra Control Center 複寫" 保護主要 Astra Control Center 執行個體。

在此範例中、 Astra Control Center 一律安裝在 `netapp-acc` 命名空間和運算子會安裝在 `netapp-acc-operator` 命名空間。

開始之前

- 您已在叢集上安裝主要 Astra Control Center 。
- 您已在不同的叢集上安裝次要 Astra Control Center 。

步驟

1. 從次要 Astra Control Center 執行個體管理主要 Astra Control Center 應用程式和目的地叢集：
   a. 登入次要 Astra Control Center 執行個體。
   b. "新增主要 Astra Control Center 叢集" (`ocp-cluster-1`) 。
   c. "新增目的地第三叢集" (`ocp-cluster-3`）用於複寫。
2. 在次要 Astra Control Center 上管理 Astra Control Center 和 Astra Control Center 營運者：
   a. 選取 * 叢集 * 、然後選取包含主要 Astra Control Center 的叢集 (`ocp-cluster-1`) 。
   b. 選取「命名空間」索引標籤。
   c. 選取 `netapp-acc` 和 `netapp-acc-operator` 命名空間：
   d. 選取「動作」功能表、然後選取 * 「定義為應用程式」 * 。

e. 選取 * 在應用程式中檢視 * 以查看定義的應用程式。

3. 設定複寫的後端：

> 複寫需要主要 Astra Control Center 叢集和目的地叢集 (`ocp-cluster-3`）使用不同的對等 ONTAP 儲存設備後端。
> 在每個後端被逐一偵測並新增至 Astra Control 之後、後端會出現在「後端」頁面的 * 探索 * 標籤中。

a. "新增對等後端" 至主叢集上的 Astra Control Center 。

b. "新增對等後端" 至目的地叢集上的 Astra Control Center 。

4. 設定複寫：

a. 在應用程式畫面上、選取 `netapp-acc` 應用程式：

b. 選取 * 設定複寫原則 * 。

c. 選取 `ocp-cluster-3` 作為目的地叢集。

d. 選取儲存類別。

e. 輸入 `netapp-acc` 作為目的地命名空間。

f. 視需要變更複寫頻率。

g. 選擇*下一步*。

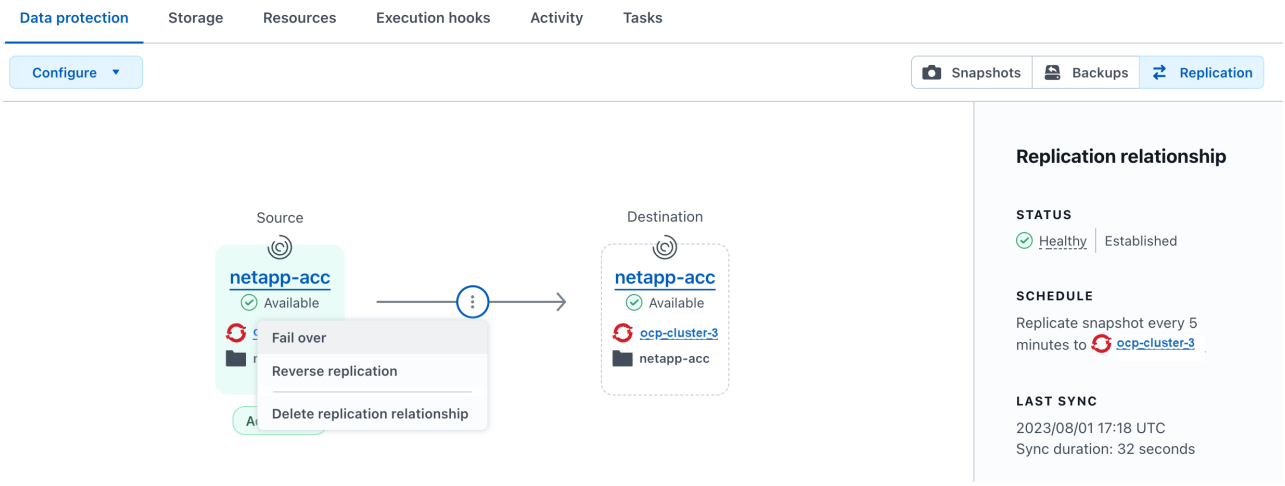
h. 確認組態正確、然後選取 * 儲存 * 。

複寫關係會從轉換 `Establishing` 至 `Established`。啟用時、此複寫會每五分鐘進行一次、直到刪除複寫組態為止。

5. 如果主系統毀損或無法再存取、請將複寫容錯移轉至其他叢集：

> 請確定目的地叢集未安裝 Astra Control Center 、以確保容錯移轉成功。

a. 選取垂直省略符號圖示、然後選取 * 容錯移轉 * 。



b. 確認詳細資料、然後選取 * 容錯移轉 * 以開始容錯移轉程序。

複寫關係狀態會變更為 `Failing over` 然後 `Failed over` 完成時。

6. 完成容錯移轉組態：

   a. 開啟終端機、並使用第三個叢集的 kubeconfig 進行連線 (`ocp-cluster-3`）。此叢集現在已安裝 Astra Control Center 。

   b. 確定第三個叢集上的 Astra Control Center FQDN (`ocp-cluster-3`）。

   c. 更新組態以參考 Astra Control Center DNS ：

   ```
   kubectl edit acc -n netapp-acc
   ```

      i. 變更 `astraAddress` 低於 `spec:` 使用 FQDN (`ocp-cluster-3.company.com`）。

      ii. 儲存組態。

      iii. 確認地址已更新：

      ```
      kubectl get acc -n netapp-acc
      ```

   d. 確認所有必要的傳輸 CRD 都存在：

   ```
   kubectl get crds | grep traefik
   ```

   必要的傳輸 CRD ：

```
ingressroutes.traefik.containo.us
ingressroutes.traefik.io
ingressroutetcps.traefik.containo.us
ingressroutetcps.traefik.io
ingressrouteudps.traefik.containo.us
ingressrouteudps.traefik.io
middlewares.traefik.containo.us
middlewares.traefik.io
middlewaretcps.traefik.containo.us
middlewaretcps.traefik.io
serverstransports.traefik.containo.us
serverstransports.traefik.io
tlsoptions.traefik.containo.us
tlsoptions.traefik.io
tIsstores.traefik.containo.us
tIsstores.traefik.io
traefikservices.traefik.containo.us
traefikservices.traefik.io
```

a. 如果上述部分客戶需求日遺失：

   i. 前往 "傳輸文件"。

   ii. 將「定義」區域複製到檔案中。

   iii. 套用變更：

```
kubectl apply -f <file name>
```

   iv. 重新啟動傳輸：

```
kubectl get pods -n netapp-acc | grep -e "traefik" | awk '{print
$1}' | xargs kubectl delete pod -n netapp-acc"
```

b. 前往 還原 Astra Control Center 操作員 本文件的一節、以完成還原程序。

步驟 **2** ：還原 **Astra Control Center** 操作員

使用次要 Astra Control Center 、從備份還原主要 Astra Control Center 營運者。目的地命名空間必須與來源命名空間相同。在從主要來源叢集刪除 Astra Control Center 的情況下、仍會存在備份以執行相同的還原步驟。

步驟

1. 選取 * 應用程式 * 、然後選取運算子應用程式的名稱 (`netapp-acc-operator`) 。
2. 從「動作」欄的「選項」功能表中、選取 * 還原 *
3. 選擇 * 還原至新命名空間 * 作為還原類型。
4. 選擇目的地第三叢集 (`ocp-cluster-3`) 。
5. 將命名空間變更為與主要來源叢集相關聯的命名空間 (`netapp-acc-operator`) 。
6. 選取先前採取的備份做為還原來源。
7. 選取 * 使用原始儲存類別還原 * 。
8. 選取 * 還原所有資源 * 。
9. 查看詳細資料、然後按一下 * 還原 * 以開始還原程序。

   「應用程式」頁面會顯示正在還原至目的地第三叢集的 Astra Control Center 操作員 (`ocp-cluster-3`) 。程序完成時、狀態會顯示為 `Available`。10 分鐘內、網頁上的 DNS 位址應該會解析。

結果

Astra Control Center 、其註冊叢集、以及具有快照和備份的託管應用程式、現在可在目的地第三叢集上使用 (`ocp-cluster-3`) 。您在原始執行個體上所擁有的任何保護原則、也會出現在新執行個體上。您可以繼續執行排程或隨需備份和快照。

疑難排解

判斷系統健全狀況、以及保護程序是否成功。

- * Pod 未執行 * ：確認所有 Pod 均已啟動並執行：

```
kubectl get pods -n netapp-acc
```

如果中有部分 Pod `CrashLookBackOff` 請重新啟動、然後將其轉換至 `Running` 州/省。

- * 確認系統狀態 * ：確認 Astra Control Center 系統已進入 `ready` 州：

```
kubectl get acc -n netapp-acc
```

回應：

```
NAME  UUID                                   VERSION    ADDRESS
READY
astra 89f4fd47-0cf0-4c7a-a44e-43353dc96ba8 23.07.0-24 ocp-cluster-
1.company.com           True
```

- * 確認部署狀態 * ：顯示 Astra Control Center 部署資訊以確認 `Deployment State` 是 `Deployed`。

```
kubectl describe acc astra -n netapp-acc
```

- * 已還原的 Astra Control Center UI 會傳回 404 錯誤 * ：如果您已選取此選項、則會傳回此錯誤 * `AccTraefik` 作為入口選項、請檢查 TRAefik 客戶需求日 確保全部安裝完畢。

監控應用程式和叢集健全狀況

檢視應用程式與叢集健全狀況的摘要

選取*儀表板*以查看應用程式、叢集、儲存後端及其健全狀況的高層級檢視。

這些不只是靜態數字或狀態、您可以逐一深入瞭解。例如、如果應用程式未受到完整保護、您可以將游標停留在圖示上、以識別哪些應用程式未受到完整保護、這也是原因之一。

應用程式並排顯示

「應用程式」方塊可協助您識別下列項目：

- 您目前使用Astra管理的應用程式數量。
- 這些託管應用程式是否健全。
- 應用程式是否受到完整保護（如果有最近的備份可用、則會受到保護）。
- 已探索但尚未管理的應用程式數量。

理想情況下、這個數字會是零、因為您會在發現應用程式之後管理或忽略這些應用程式。然後、您可以監控儀表板上探索到的應用程式數量、以識別開發人員何時將新應用程式新增至叢集。

叢集並排顯示

「叢集」方塊提供類似的詳細資料、說明您使用Astra Control Center管理的叢集健全狀況、您也可以深入瞭解更多詳細資料、就像使用應用程式一樣。

儲存後端並排顯示

「儲存後端」方塊提供資訊、協助您識別儲存後端的健全狀況、包括：

- 管理多少個儲存後端
- 這些託管後端是否健全
- 後端是否受到完整保護
- 已探索但尚未管理的後端數目。

檢視叢集健全狀況並管理儲存類別

新增要由Astra Control Center管理的叢集之後、您可以檢視叢集的詳細資料、例如其位置、工作節點、持續磁碟區和儲存類別。您也可以變更受管理叢集的預設儲存類別。

檢視叢集健全狀況和詳細資料

您可以檢視叢集的詳細資料、例如其位置、工作節點、持續磁碟區和儲存類別。

步驟

1. 在Astra Control Center UI中、選取* Clusters*。
2. 在「叢集」頁面上、選取您要檢視其詳細資料的叢集。

> 如果叢集位於 `removed` 狀態但叢集和網路連線似乎正常（外部使用Kubernetes API存取叢集的嘗試成功）、您提供給Astra Control的Kubeconfig可能不再有效。這可能是因為叢集上的憑證輪替或過期。若要修正此問題、請使用更新Astra Control中與叢集相關的認證資料 "Astra Control API"。

3. 查看*概述*、*儲存設備*和*活動*索引標籤上的資訊、以尋找您要尋找的資訊。
   - 總覽：工作節點的詳細資料、包括其狀態。
   - * Storage *：與運算相關的持續磁碟區、包括儲存類別和狀態。
   - 活動：顯示與叢集相關的活動。

> 您也可以從Astra控制中心*儀表板*開始檢視叢集資訊。在*叢集*索引標籤的*資源摘要*下、您可以選取受管理的叢集、然後前往*叢集*頁面。進入「叢集」頁面之後、請依照上述步驟操作。

變更預設儲存類別

您可以變更叢集的預設儲存類別。當Astra Control管理叢集時、它會追蹤叢集的預設儲存類別。

請勿使用kubectl命令變更儲存類別。請改用此程序。若使用KECBECVL、Astra Control將會回復變更。

步驟

1. 在Astra Control Center Web UI中、選取* Clusters*。
2. 在「叢集」頁面上、選取您要變更的叢集。
3. 選擇* Storage*（儲存設備）選項卡。
4. 選擇*儲存類別*類別。
5. 針對您要設為預設的儲存類別、選取「動作」功能表。
6. 選擇*設為預設*。

檢視應用程式的健全狀況和詳細資料

在您開始管理應用程式之後、Astra Control會提供應用程式的詳細資料、讓您識別應用程式的狀態（是否健全）、保護狀態（是否在故障時受到完整保護）、Pod、持續儲存設備等。

步驟

1. 在Astra Control Center UI中、選取* Applications*、然後選取應用程式名稱。
2. 檢閱資訊。
   - 應用程式狀態：提供反映Kubernetes應用程式狀態的狀態。例如、Pod和持續磁碟區是否在線上？如果某個應用程式不健全、您必須查看Kubernetes記錄檔、在叢集上進行疑難排解。Astra並未提供資訊來協助您修正毀損的應用程式。
   - 應用程式保護狀態：提供應用程式受保護程度的狀態：
     - 完全保護：應用程式有作用中的備份排程、而且備份成功的時間不到一週
     - 部分保護：應用程式有作用中的備份排程、作用中的快照排程、或成功的備份或快照
     - 未受保護：未受到完整保護或部分保護的應用程式。

       您必須等到最近進行備份之後、才能獲得完整保護。這很重要、因為備份儲存在遠離持續磁碟區的物件存放區中。如果故障或意外將叢集及其持續儲存設備擦除、則需要備份才能恢復。快照無法讓您恢復。
   - 總覽：與應用程式相關聯的Pod狀態資訊。
   - 資料保護：可讓您設定資料保護原則、並檢視現有的快照與備份。
   - 儲存設備：顯示應用程式層級的持續磁碟區。持續磁碟區的狀態是從Kubernetes叢集的觀點來看。
   - 資源：可讓您驗證要備份和管理的資源。
   - 活動：顯示與應用程式相關的活動。

您也可以從Astra Control Center * Dashboard 開始檢視應用程式資訊。在*應用程式*索引標籤的*資源摘要*下、您可以選取託管應用程式、以前往*應用程式*頁面。進入「*應用程式」頁面之後、請依照上述步驟操作。

管理您的帳戶

管理本機使用者和角色

您可以使用Astra Control UI來新增、移除及編輯Astra Control Center安裝的使用者。您可以使用Astra Control UI或 "Astra Control API" 管理使用者：

您也可以使用LDAP為選取的使用者執行驗證。

使用**LDAP**

LDAP是存取分散式目錄資訊的業界標準傳輸協定、也是企業驗證的熱門選擇。您可以將Astra Control Center連線至LDAP伺服器、為選取的Astra Control使用者執行驗證。在高層級上、組態包括將Astra與LDAP整合、以及定義與LDAP定義對應的Astra Control使用者和群組。您可以使用Astra Control API或Web UI來設定LDAP驗證、以及LDAP使用者和群組。如需詳細資訊、請參閱下列文件：

- "使用Astra Control API來管理遠端驗證和使用者"
- "使用Astra Control UI來管理遠端使用者和群組"
- "使用Astra Control UI來管理遠端驗證"

新增使用者

帳戶擁有者和系統管理員可以新增更多使用者至Astra Control Center安裝。

步驟

1. 在*管理您的帳戶*導覽區域中、選取*帳戶*。
2. 選取*使用者*索引標籤。
3. 選取*新增使用者*。
4. 輸入使用者的名稱、電子郵件地址和暫用密碼。

   使用者必須在第一次登入時變更密碼。

5. 選取具有適當系統權限的使用者角色。

   每個角色都提供下列權限：

   - *檢視器*可以檢視資源。
   - *成員*具有檢視者角色權限、可管理應用程式與叢集、取消管理應用程式、以及刪除快照與備份。
   - 「管理」具有「成員角色」權限、可新增及移除擁有者以外的任何其他使用者。
   - *擁有者*具有管理員角色權限、可新增及移除任何使用者帳戶。

6. 若要新增限制給具有成員或檢視者角色的使用者、請啟用*限制角色限制*核取方塊。

   如需新增限制的詳細資訊、請參閱 "管理本機使用者和角色"。

7. 選取*「Add*」。

管理密碼

您可以在Astra Control Center中管理使用者帳戶的密碼。

變更您的密碼

您可以隨時變更使用者帳戶的密碼。

步驟

1. 選取畫面右上角的使用者圖示。
2. 選擇*設定檔*。
3. 從「動作」欄的「選項」功能表中選取「變更密碼」。
4. 輸入符合密碼需求的密碼。
5. 再次輸入密碼進行確認。
6. 選擇*變更密碼*。

重設其他使用者的密碼

如果您的帳戶具有「管理員」或「擁有者」角色權限、您可以重設其他使用者帳戶和您自己的密碼。當您重設密碼時、您會設定使用者登入時必須變更的暫用密碼。

步驟

1. 在*管理您的帳戶*導覽區域中、選取*帳戶*。
2. 選取「動作」下拉式清單。
3. 選擇*重設密碼*。
4. 輸入符合密碼需求的暫用密碼。
5. 再次輸入密碼進行確認。

   下次使用者登入時、系統會提示使用者變更密碼。

6. 選擇*重設密碼*。

移除使用者

擁有擁有者或管理員角色的使用者可以隨時從帳戶中移除其他使用者。

步驟

1. 在*管理您的帳戶*導覽區域中、選取*帳戶*。
2. 在「使用者」索引標籤中、選取您要移除之每個使用者列中的核取方塊。
3. 從「動作」欄的「選項」功能表中、選取「移除使用者」。
4. 出現提示時、請輸入「移除」一詞、然後選取「是、移除使用者*」、確認刪除。

結果

Astra Control Center會將使用者從帳戶中移除。

管理角色

您可以新增命名空間限制、並將使用者角色限制在這些限制中、藉此管理角色。這可讓您控制組織內資源的存取。您可以使用Astra Control UI或 "Astra Control API" 以管理角色。

將命名空間限制新增至角色

管理員或擁有者使用者可以將命名空間限制新增至「成員」或「檢視者」角色。

步驟

1. 在*管理您的帳戶*導覽區域中、選取*帳戶*。
2. 選取*使用者*索引標籤。
3. 在「動作」欄中、選取具有「成員」或「檢視者」角色的使用者功能表按鈕。
4. 選擇*編輯角色*。
5. 啟用「限制角色*」核取方塊。

   此核取方塊僅適用於「成員」或「檢視者」角色。您可以從*角色*下拉式清單中選取不同的角色。

6. 選取*新增限制*。

   您可以依命名空間或命名空間標籤檢視可用限制清單。

7. 在*限制類型*下拉式清單中、視命名空間的設定方式而定、選取* Kubernetes命名空間*或* Kubernetes命名空間標籤*。
8. 從清單中選取一或多個命名空間或標籤、以構成限制、限制角色只能使用這些命名空間。
9. 選擇* Confirm（確認）*。

   「編輯角色」頁面會顯示您為此角色選擇的限制清單。

10. 選擇* Confirm（確認）*。

    在「帳戶」頁面上、您可以在「角色」欄中檢視任何成員或檢視者角色的限制條件。

> 如果您啟用角色的限制、並選取* Confirm（確認）*而不新增任何限制、則該角色會被視為具有完整限制（該角色無法存取指派給命名空間的任何資源）。

從角色移除命名空間限制

管理員或擁有者使用者可以從角色移除命名空間限制。

步驟

1. 在*管理您的帳戶*導覽區域中、選取*帳戶*。
2. 選取*使用者*索引標籤。
3. 在「動作」欄中、選取具有作用中限制之「成員」或「檢視者」角色的使用者功能表按鈕。
4. 選擇*編輯角色*。

「編輯角色」對話方塊會顯示角色的作用中限制。

5. 選取您需要移除之限制右側的* X*。
6. 選擇* Confirm（確認）*。

以取得更多資訊

- "使用者角色和命名空間"

管理遠端驗證

LDAP是存取分散式目錄資訊的業界標準傳輸協定、也是企業驗證的熱門選擇。您可以將Astra Control Center連線至LDAP伺服器、為選取的Astra Control使用者執行驗證。

在高層級上、組態包括將Astra與LDAP整合、以及定義與LDAP定義對應的Astra Control使用者和群組。您可以使用Astra Control API或Web UI來設定LDAP驗證、以及LDAP使用者和群組。

> Astra Control Center 會使用啟用遠端驗證時所設定的使用者登入屬性來搜尋和追蹤遠端使用者。此欄位中必須存在電子郵件地址（「郵件」）或使用者主體名稱（「 userPrincipalName 」）的屬性、您想要在 Astra Control Center 中顯示的任何遠端使用者都必須存在。此屬性在 Astra Control Center 中作為驗證的使用者名稱、並在搜尋遠端使用者時使用。

新增**LDAPS**驗證的憑證

新增LDAP伺服器的私有TLS憑證、以便Astra Control Center在您使用LDAPS連線時、能夠與LDAP伺服器進行驗證。您只需要執行一次、或是安裝的憑證過期時。

步驟

1. 前往*帳戶*。
2. 選取*憑證*索引標籤。
3. 選取*「Add*」。
4. 上傳 `.pem` 將檔案內容從剪貼簿中歸檔或貼上。
5. 選取「信任」核取方塊。
6. 選取*新增憑證*。

啟用遠端驗證

您可以啟用LDAP驗證、並設定Astra Control與遠端LDAP伺服器之間的連線。

開始之前

如果您打算使用LDAPS、請確定LDAP伺服器的私有TLS憑證已安裝在Astra控制中心、以便Astra控制中心能夠與LDAP伺服器進行驗證。請參閱 新增LDAPS驗證的憑證 以取得相關指示。

步驟

1. 前往*帳戶>連線*。
2. 在*遠端驗證*窗格中、選取組態功能表。

3. 選擇*連接*。
4. 輸入伺服器IP位址、連接埠及偏好的連線傳輸協定（LDAP或LDAPS）。

> 最佳實務做法是在連接LDAP伺服器時使用LDAPS。您必須先在Astra Control Center中安裝LDAP伺服器的私有TLS憑證、才能連線至LDAPS。

5. 以電子郵件格式輸入服務帳戶認證（administrator@example.com）。Astra Control會在連線至LDAP伺服器時使用這些認證資料。
6. 在 * 使用者比對 * 區段中、執行下列步驟：
   a. 輸入從 LDAP 伺服器擷取使用者資訊時要使用的基礎 DN 和適當的使用者搜尋篩選器。
   b. （選用）如果您的目錄使用使用者登入屬性 `userPrincipalName` 而非 `mail`、輸入 `userPrincipalName` 在 * 使用者登入屬性 * 欄位的正確屬性中。
7. 在「群組比對」區段中、輸入群組搜尋基礎DN和適當的自訂群組搜尋篩選器。

> 請務必使用正確的基礎辨別名稱（DN）和適當的搜尋篩選器來搜尋*使用者比對*和*群組比對*。基礎DN會告知Astra Control在目錄樹狀結構的哪個層級開始搜尋、而搜尋篩選器則會限制Astra Control從目錄樹狀結構中搜尋的部分。

8. 選擇*提交*。

結果

「遠端驗證」窗格狀態會移至*「擱置中」、並在建立與**LDAP**伺服器的連線時移至「已連線」*。

停用遠端驗證

您可以暫時停用與LDAP伺服器的作用中連線。

> 停用LDAP伺服器連線時、會儲存所有設定、並保留從該LDAP伺服器新增至Astra Control的所有遠端使用者和群組。您可以隨時重新連線至此LDAP伺服器。

步驟

1. 前往*帳戶>連線*。
2. 在*遠端驗證*窗格中、選取組態功能表。
3. 選擇*停用*。

結果

「遠端驗證」窗格狀態會移至「停用」。所有遠端驗證設定、遠端使用者和遠端群組都會保留下來、您可以隨時重新啟用連線。

編輯遠端驗證設定

如果您已停用LDAP伺服器的連線、或*遠端驗證*窗格處於「連線錯誤」狀態、您可以編輯組態設定。

> 當「遠端驗證」窗格處於「已停用」狀態時、您無法編輯LDAP伺服器URL或IP位址。您需要 [中斷遠端驗證] 第一。

步驟

1. 前往*帳戶>連線*。
2. 在*遠端驗證*窗格中、選取組態功能表。
3. 選擇*編輯*。
4. 進行必要的變更、然後選取*編輯*。

中斷遠端驗證

您可以中斷與LDAP伺服器的連線、並從Astra Control移除組態設定。

如果您是 LDAP 使用者且中斷連線、工作階段將立即結束當您中斷與LDAP伺服器的連線時、該LDAP伺服器的所有組態設定都會從Astra Control中移除、以及從該LDAP伺服器新增的任何遠端使用者和群組。

步驟

1. 前往*帳戶>連線*。
2. 在*遠端驗證*窗格中、選取組態功能表。
3. 選擇*中斷連線*。

結果

「遠端驗證」窗格狀態會移至「中斷連線」。遠端驗證設定、遠端使用者和遠端群組都會從Astra Control中移除。

管理遠端使用者和群組

如果您已在Astra Control系統上啟用LDAP驗證、您可以搜尋LDAP使用者和群組、並將其納入系統的核准使用者中。

新增遠端使用者

帳戶擁有者和管理員可以將遠端使用者新增至Astra Control。Astra Control Center 最多支援 10 、 000 名 LDAP 遠端使用者。

Astra Control Center 會使用啟用遠端驗證時所設定的使用者登入屬性來搜尋和追蹤遠端使用者。此欄位中必須存在電子郵件地址（「郵件」）或使用者主體名稱（「 userPrincipalName 」）的屬性、您想要在 Astra Control Center 中顯示的任何遠端使用者都必須存在。此屬性在 Astra Control Center 中作為驗證的使用者名稱、並在搜尋遠端使用者時使用。

如果系統上已存在具有相同電子郵件地址的本機使用者（根據「 mail 」或「 user principal name 」屬性）、則無法新增遠端使用者。若要將使用者新增為遠端使用者、請先從系統中刪除本機使用者。

步驟

1. 前往*帳戶*區域。
2. 選取*使用者與群組*索引標籤。

3. 在頁面最右側、選取*遠端使用者*。
4. 選取*「Add*」。
5. 或者、您也可以在*依電子郵件篩選*欄位中輸入使用者的電子郵件地址、以搜尋LDAP使用者。
6. 從清單中選取一或多個使用者。
7. 指派角色給使用者。

   如果您指派不同的角色給使用者和使用者群組、則權限越高的角色優先。

8. 您也可以將一或多個命名空間限制指派給此使用者、然後選取*限制角色至限制*以強制執行限制。您可以選取*新增限制*來新增命名空間限制。

   當使用者透過LDAP群組成員資格指派多個角色時、最具權限角色的限制是唯一會生效的限制。例如、如果具有本機檢視器角色的使用者加入三個繫結至「成員」角色的群組、則「成員」角色的限制總和會生效、而且會忽略「檢視器」角色的任何限制。

9. 選取*「Add*」。

結果

新使用者會出現在遠端使用者清單中。在此清單中、您可以看到使用者的作用中限制、也可以從*動作*功能表管理使用者。

新增遠端群組

若要一次新增許多遠端使用者、帳戶擁有者和管理員可以將遠端群組新增至Astra Control。當您新增遠端群組時、該群組中的所有遠端使用者都可以登入 Astra Control 、並繼承與該群組相同的角色。

Astra Control Center 最多支援 5 、 000 個 LDAP 遠端群組。

步驟

1. 前往*帳戶*區域。
2. 選取*使用者與群組*索引標籤。
3. 在頁面最右側、選取*遠端群組*。
4. 選取*「Add*」。

   在此視窗中、您可以看到Astra Control從目錄擷取的LDAP群組一般名稱和辨別名稱清單。

5. 或者、您也可以在*依一般名稱篩選*欄位中輸入群組的一般名稱、以搜尋LDAP群組。
6. 從清單中選取一或多個群組。
7. 指派角色給群組。

   您選取的角色會指派給此群組中的所有使用者。如果您指派不同的角色給使用者和使用者群組、則權限越高的角色優先。

8. 您也可以將一或多個命名空間限制指派給此群組、然後選取*限制角色限制*以強制執行限制。您可以選取*新增限制*來新增命名空間限制。

當使用者透過LDAP群組成員資格指派多個角色時、最具權限角色的限制是唯一會生效的限制。例如、如果具有本機檢視器角色的使用者加入三個繫結至「成員」角色的群組、則「成員」角色的限制總和會生效、而且會忽略「檢視器」角色的任何限制。

9. 選取*「Add*」。

結果

新群組會出現在遠端群組清單中。此群組中的遠端使用者不會出現在遠端使用者清單中、直到每個遠端使用者登入為止。在此清單中、您可以查看群組的詳細資料、也可以從*「動作」*功能表管理群組。

檢視及管理通知

Astra會在行動完成或失敗時通知您。例如、如果成功完成應用程式的備份、您會看到通知。

您可以從介面右上角管理這些通知：

步驟

1. 選取右上角的未讀取通知數。
2. 檢閱通知、然後選取*標示為已讀取*或*顯示所有通知*。

   如果您選取*顯示所有通知*、則會載入「通知」頁面。

3. 在*通知*頁面上、檢視通知、選取您要標示為已讀的通知、選取*行動*、然後選取*標示為已讀*。

新增及移除認證資料

隨時從ONTAP 您的帳戶新增及移除本地私有雲端供應商的認證資料、例如用OpenShift管理的Kubernetes叢集、或Unmanaged Kubernetes叢集。Astra Control Center會使用這些認證資料來探索叢集和叢集上的應用程式、並代表您配置資源。

請注意、Astra Control Center中的所有使用者都共用相同的認證資料集。

新增認證資料

您可以在管理叢集時、將認證新增至Astra Control Center。若要透過新增叢集來新增認證、請參閱 "新增Kubernetes叢集"。

如果您建立自己的 kubeconfig 檔案、則應該只定義其中的 * 一 * 內容元素。請參閱 "Kubernetes 文件" 以取得建立 kubeconfig 檔案的相關資訊。

移除認證資料

隨時從帳戶移除認證資料。您只能在之後移除認證 "取消管理所有相關的叢集"。

您新增至Astra Control Center的第一組認證資料一律使用中、因為Astra Control Center使用認證資料來驗證備份儲存區。最好不要移除這些認證資料。

步驟

1. 選擇*帳戶*。
2. 選取*認證*索引標籤。
3. 在*狀態*欄中選取您要移除之認證的「選項」功能表。
4. 選擇*移除*。
5. 輸入「移除」一詞以確認刪除、然後選取*是、移除認證*。

結果

Astra Control Center會從帳戶移除認證資料。

監控帳戶活動

您可以檢視Astra Control帳戶中活動的詳細資料。例如、當邀請新使用者、新增叢集或擷取快照時。您也可以將帳戶活動匯出至CSV檔案。

如果您從Astra Control管理Kubernetes叢集、且Astra Control已連線Cloud Insights 至原地、Astra Control會將事件記錄傳送至Cloud Insights 原地。日誌資訊（包括Pod部署和PVc附件的相關資訊）會顯示在Astra Control活動記錄中。使用此資訊來識別您所管理的Kubernetes叢集上的任何問題。

檢視**Astra Control**中的所有帳戶活動

1. 選擇*活動*。
2. 使用篩選器縮小活動清單範圍、或使用搜尋方塊找到您想要的確切內容。
3. 選取*匯出至CSV*、將您的帳戶活動下載至CSV檔案。

檢視特定應用程式的帳戶活動

1. 選取*應用程式*、然後選取應用程式名稱。
2. 選擇*活動*。

檢視叢集的帳戶活動

1. 選取*叢集*、然後選取叢集名稱。
2. 選擇*活動*。

採取行動以解決需要注意的事件

1. 選擇*活動*。
2. 選取需要注意的事件。
3. 選取*「採取行動」*下拉式選項。

   您可在此清單中檢視可能採取的修正行動、檢視與問題相關的文件、並取得協助解決問題的支援。

更新現有授權

您可以將試用版授權轉換為完整授權、也可以使用新授權來更新現有的試用版或完整授權。如果您沒有完整授權、請與NetApp銷售聯絡人聯絡、以取得完整授權與序號。您可以使用Astra Control Center UI或 "Astra Control API" 以更新現有授權。

步驟

1. 登入 "NetApp 支援網站"。
2. 存取Astra Control Center下載頁面、輸入序號、然後下載完整的NetApp授權檔案（NLF）。
3. 登入Astra Control Center UI。
4. 從左側導覽中、選取*帳戶*>*授權*。
5. 在「帳戶>*授權*」頁面中、選取現有授權的狀態下拉式功能表、然後選取「取代」。
6. 瀏覽至您下載的授權檔案。
7. 選取*「Add*」。

「帳戶>*授權*」頁面會顯示授權資訊、到期日、授權序號、帳戶ID及使用的CPU單位。

以取得更多資訊

- "Astra Control Center授權"

管理儲存庫

如果您想要備份應用程式和持續儲存設備、或是想要跨叢集複製應用程式、物件存放區供應商是不可或缺的。使用Astra Control Center、新增物件存放區供應商做為您的應用程式離叢集備份目的地。

如果您要將應用程式組態和持續儲存設備複製到同一個叢集、則不需要儲存庫。

請使用下列其中一家Amazon Simple Storage Service（S3）資源庫供應商：

- NetApp ONTAP 產品S3
- NetApp StorageGRID 產品S3
- Microsoft Azure
- 一般S3

> Amazon Web Services（AWS）和Google Cloud Platform（GCP）使用通用S3儲存區類型。

> 雖然Astra Control Center支援Amazon S3做為通用S3儲存區供應商、但Astra Control Center可能不支援所有聲稱Amazon S3支援的物件儲存區廠商。

儲存庫可以位於下列其中一種狀態：

- 擱置中：已排定要探索的儲存區。

- 可用：鏟斗可供使用。
- 已移除：目前無法存取貯體。

如需如何使用Astra Control API管理儲存區的指示、請參閱 "Astra Automation和API資訊"。

您可以執行與管理儲存庫相關的工作：

- "新增儲存庫"
- [編輯儲存庫]
- [設定預設儲存區]
- [旋轉或移除庫位認證資料]
- [移除貯體]

Astra Control Center中的S3鏟斗未報告可用容量。在備份或複製由Astra Control Center管理的應用程式之前、請先查看ONTAP 資訊庫（英文）或StorageGRID 資訊庫（英文）管理系統中的庫位資訊。

編輯儲存庫

您可以變更儲存區的存取認證資訊、並變更所選儲存區是否為預設儲存區。

新增儲存庫時、請選擇正確的儲存庫供應商、並提供該供應商的適當認證資料。例如、UI接受NetApp ONTAP S3作為類型並接受StorageGRID 驗證、但這將導致所有未來使用此儲存庫的應用程式備份與還原失敗。請參閱 "版本資訊"。

步驟

1. 從左側導覽中、選取*鏟斗*。
2. 從「動作」欄的功能表中、選取*編輯*。
3. 變更儲存桶類型以外的任何資訊。

   您無法修改貯體類型。

4. 選擇*更新*。

設定預設儲存區

當您跨叢集執行實體複本時、Astra Control需要預設的儲存區。請依照下列步驟為所有叢集設定預設儲存區。

步驟

1. 轉至* Cloud Instances *。
2. 選取清單中雲端執行個體*「Actions」（動作）欄中的功能表。
3. 選擇*編輯*。
4. 在* Bucket *清單中、選取您要做為預設值的儲存區。
5. 選擇*保存*。

旋轉或移除庫位認證資料

Astra Control使用儲存區認證來取得S3儲存區的存取權、並提供密碼金鑰、以便Astra Control Center能夠與儲存區通訊。

旋轉儲存庫認證資料

如果您旋轉認證資料、請在維護期間（排程或隨需）無備份進行時、於維護期間旋轉認證資料。

編輯及旋轉認證的步驟

1. 從左側導覽中、選取*鏟斗*。
2. 從「動作」欄的「選項」功能表中、選取「編輯」。
3. 建立新認證資料。
4. 選擇*更新*。

移除庫位認證資料

只有在新認證已套用至庫位、或庫位已不再有效使用時、才應移除庫位認證。

您新增至Astra Control的第一組認證資料一律使用中、因為Astra Control使用認證資料來驗證備份儲存區。如果儲存區正在使用中、請勿移除這些認證資料、因為這會導致備份失敗和備份不可用。

如果您確實移除作用中的儲存區認證、請參閱 "移除庫位認證疑難排解"。

如需如何使用Astra Control API移除S3認證的指示、請參閱 "Astra Automation和API資訊"。

移除貯體

您可以移除不再使用或不健全的庫位。您可能會想要這麼做、讓物件存放區組態保持簡單且最新狀態。

您無法移除預設的儲存區。如果您要移除該儲存區、請先選取另一個儲存區做為預設值。

開始之前

- 開始之前、您應檢查以確保此儲存區沒有執行中或已完成的備份。
- 您應檢查以確保儲存庫未用於任何作用中的保護原則。

如果有、您將無法繼續。

步驟

1. 從左側導覽中選取*鏟斗*。
2. 從* Actions（操作）功能表中、選取*移除。

Astra Control會先確保不會有使用儲存庫進行備份的排程原則、而且您要移除的儲存庫中沒有作用中的備份。

3. 輸入「移除」以確認動作。
4. 選擇*是、移除桶*。

如需詳細資訊、請參閱

- "使用Astra Control API"

管理儲存後端

將Astra Control中的儲存叢集管理為儲存後端、可讓您在持續磁碟區（PV）與儲存後端之間建立連結、以及取得額外的儲存指標。如果Astra Control Center連接Cloud Insights到VMware、您可以監控儲存容量和健全狀況詳細資料、包括效能。

如需如何使用Astra Control API管理儲存後端的指示、請參閱 "Astra Automation和API資訊"。

您可以完成下列與管理儲存後端相關的工作：

- "新增儲存後端"
- [檢視儲存後端詳細資料]
- [編輯儲存後端驗證詳細資料]
- [管理探索到的儲存後端]
- [取消管理儲存後端]
- [移除儲存後端]

檢視儲存後端詳細資料

您可以從儀表板或後端選項檢視儲存後端資訊。

從儀表板檢視儲存後端詳細資料

步驟

1. 從左側導覽中選取*儀表板*。
2. 檢閱儀表板的儲存後端面板、其中會顯示狀態：
   - 不健全：儲存設備未處於最佳狀態。這可能是因為延遲問題、或是應用程式因為容器問題而降級。
   - 一切正常：儲存設備已經過管理、並處於最佳狀態。
   - 探索：儲存設備已被探索、但未由Astra Control管理。

從後端選項檢視儲存後端詳細資料

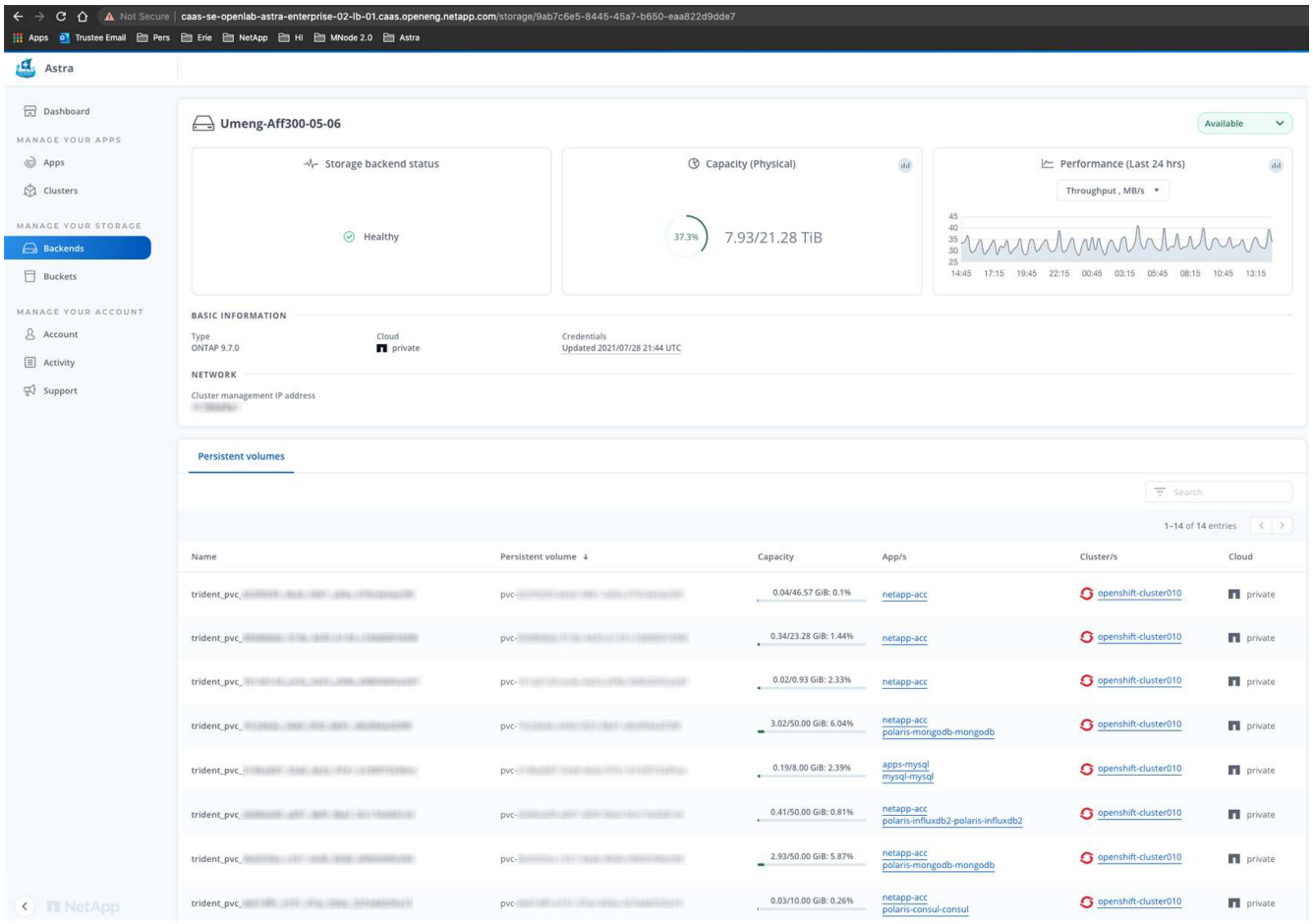
檢視後端健全狀況、容量和效能（IOPS處理量和/或延遲）的相關資訊。

您可以看到Kubernetes應用程式所使用的磁碟區、這些磁碟區儲存在選定的儲存後端。有了此功能、您可以查看更多資訊。Cloud Insights請參閱 "本文檔 Cloud Insights"。

步驟

1. 在左側導覽區域中、選取*後端*。
2. 選取儲存後端。

如果您連線至NetApp Cloud Insights 解決方案、Cloud Insights 則會在「後端」頁面上顯示來自於《》的資料摘錄。



3. 若要直接移至Cloud Insights 「不顯示」、請選取Cloud Insights 「指標」影像旁的*「不顯示」圖示。

編輯儲存後端驗證詳細資料

Astra Control Center 提供兩種驗證 ONTAP 後端的模式。

- * 認證型驗證 * ：具有必要權限的 ONTAP 使用者的使用者名稱和密碼。您應該使用預先定義的安全登入角色、例如 admin 、以確保與 ONTAP 版本的最大相容性。
- * 憑證型驗證 * ： Astra 控制中心也可以使用安裝在後端的憑證與 ONTAP 叢集通訊。您應該使用用戶端憑證、金鑰和信任的 CA 憑證（如果使用）（建議使用）。

您可以更新現有的後端、以從一種驗證類型移至另一種方法。一次只支援一種驗證方法。

如需啟用憑證型驗證的詳細資訊、請參閱 "在 ONTAP 儲存後端啟用驗證"。

步驟

1. 從左側導覽中選取*後端*。
2. 選取儲存後端。
3. 在「認證」欄位中、選取 * 編輯 * 圖示。
4. 在「編輯」頁面中、選取下列其中一項。
   - * 使用管理員認證 * ：輸入 ONTAP 叢集管理 IP 位址和管理認證。認證資料必須是整個叢集的認證資料。

     您在此處輸入認證的使用者必須擁有 `ontapi` 使用者登入存取方法已在ONTAP 支援的叢集上的「支援系統管理程式」中啟用ONTAP 。如果您打算使用SnapMirror複寫、請套用具有「admin」角色的使用者認證、該角色具有存取方法 `ontapi` 和 `http`、在來源ONTAP 和目的地等叢集上。請參閱 "管理ONTAP 使用者帳戶、請參閱本文檔" 以取得更多資訊。

   - * 使用憑證 * ：上傳憑證 `.pem` 檔案、憑證金鑰 `.key` 檔案、以及選擇性的憑證授權單位檔案。
5. 選擇*保存*。

管理探索到的儲存後端

您可以選擇管理未受管理但已探索到的儲存後端。當您管理儲存後端時、 Astra Control 會指出驗證憑證是否已過期。

步驟

1. 從左側導覽中選取*後端*。
2. 選取 * 探索 * 選項。
3. 選取儲存後端。
4. 從 * 動作 * 欄的選項功能表中、選取 * 管理 * 。
5. 進行變更。
6. 選擇*保存*。

取消管理儲存後端

您可以取消管理後端。

步驟

1. 從左側導覽中選取*後端*。
2. 選取儲存後端。
3. 從「動作」欄的「選項」功能表中、選取「取消管理」。
4. 輸入「unManage（取消管理）」以確認此動作。
5. 選擇*是、取消管理儲存後端*。

移除儲存後端

您可以移除不再使用的儲存後端。您可能會想要這麼做、讓您的組態保持簡單且最新狀態。

開始之前

- 確保儲存後端未受管理。
- 確保儲存後端沒有任何與叢集相關的磁碟區。

步驟

1. 從左側導覽中選取*後端*。
2. 如果管理後端、請取消管理。
   a. 選擇*託管*。
   b. 選取儲存後端。
   c. 從 * 動作 * 選項中、選取 * 取消管理 * 。
   d. 輸入「unManage（取消管理）」以確認此動作。
   e. 選擇*是、取消管理儲存後端*。
3. 選擇*已探索*。
   a. 選取儲存後端。
   b. 從 **Actions** 選項中，選擇 **Remove** 。
   c. 輸入「移除」以確認動作。
   d. 選擇*是、移除儲存後端*。

如需詳細資訊、請參閱

- "使用Astra Control API"

監控執行中的工作

您可以在Astra Control中檢視過去24小時內已完成、失敗或已取消的執行工作和工作詳細資料。例如、您可以檢視執行中備份、還原或複製作業的狀態、並查看完成百分比和預估剩餘時間等詳細資料。您可以檢視已執行的排程作業或手動啟動的作業狀態。

檢視執行中或完成的工作時、您可以展開工作詳細資料、以查看每個子工作的狀態。工作進度列會顯示綠色、代表進行中或已完成的工作、藍色代表已取消的工作、紅色代表因錯誤而失敗的工作。

對於複製作業、工作子任務包含快照和快照還原作業。

如需失敗工作的詳細資訊、請參閱 "監控帳戶活動"。

步驟

1. 當工作正在執行時、請前往*應用程式*。
2. 從清單中選取應用程式名稱。
3. 在應用程式的詳細資料中、選取*工作*索引標籤。

   您可以檢視目前或過去工作的詳細資料、並依工作狀態篩選。

工作會保留在*工作*清單中長達24小時。您可以使用設定此限制和其他工作監控設定 "Astra Control API"。

利用Cloud Insights 支援的鏈接功能來監控基礎架構

您可以設定多項選用設定、以增強Astra Control Center體驗。若要監控並深入瞭解您的完整基礎架構、請建立與NetApp Cloud Insights 的連線、設定Prometheus、或新增Fluentd連線。

如果您執行Astra Control Center的網路需要Proxy才能連線至網際網路（將支援套件上傳NetApp 支援網站 至靜態或建立Cloud Insights 連線至靜態）、您應該在Astra Control Center中設定Proxy伺服器。

- 連線Cloud Insights 至
- 連線至Prometheus
- 連接至Flud

新增Proxy伺服器以連線Cloud Insights 至指令集或NetApp 支援網站 到指令集

如果您執行Astra Control Center的網路需要Proxy才能連線至網際網路（將支援套件上傳NetApp 支援網站 至靜態或建立Cloud Insights 連線至靜態）、您應該在Astra Control Center中設定Proxy伺服器。

Astra Control Center不會驗證您為Proxy伺服器輸入的詳細資料。請確認輸入正確的值。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取*「Connect*」以新增Proxy伺服器。

4. 輸入Proxy伺服器名稱或IP位址及Proxy連接埠號碼。
5. 如果您的Proxy伺服器需要驗證、請選取核取方塊、然後輸入使用者名稱和密碼。
6. 選擇*連接*。

結果

如果您輸入的代理資訊已儲存、則「帳戶>*連線*」頁面的「* HTTP Proxy*」區段會指出其已連線、並顯示伺服器名稱。

編輯**Proxy**伺服器設定

您可以編輯Proxy伺服器設定。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取 * 編輯 * 以編輯連線。
4. 編輯伺服器詳細資料和驗證資訊。
5. 選擇*保存*。

停用**Proxy**伺服器連線

您可以停用Proxy伺服器連線。在停用之前、系統會先警告您、否則可能會對其他連線造成潛在的中斷。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取*「Disconnect*（中斷連線）」以停用連線。
4. 在開啟的對話方塊中、確認作業。

連線**Cloud Insights** 至

若要監控並深入瞭解完整的基礎架構、請將NetApp Cloud Insights 知識與Astra Control Center執行個體連結起來。包含在您的Astra Control Center授權中。Cloud Insights

應可從Astra Control Center使用的網路存取、或透過Proxy伺服器間接存取。Cloud Insights

當Astra Control Center連線Cloud Insights 至不實時、就會建立一個擷取單元Pod。此Pod可從Astra Control Center管理的儲存後端收集資料、並將資料推送到Cloud Insights此Pod需要8 GB RAM和2個CPU核心。

當 Astra 控制中心與 Cloud Insights 配對時、您不應使用 Cloud Insights 中的 * 修改部署 * 選項。

啟用 Cloud Insights 連線之後、您可以在 **Backends** 頁面上檢視處理量資訊、並在選取儲存後端後端之後連線至 Cloud Insights 。您也可以在「叢集」區段的 * 儀表板 * 上找到相關資訊、然後從該處連線至 Cloud Insights 。

開始之前

- 具有*管理*/*擁有者*權限的Astra Control Center帳戶。
- 有效的Astra Control Center授權。
- 如果您執行Astra Control Center的網路需要Proxy才能連線至網際網路、則為Proxy伺服器。

如果您是Cloud Insights 不熟悉的人、請熟悉這些功能。請參閱 "本文檔 Cloud Insights"。

步驟

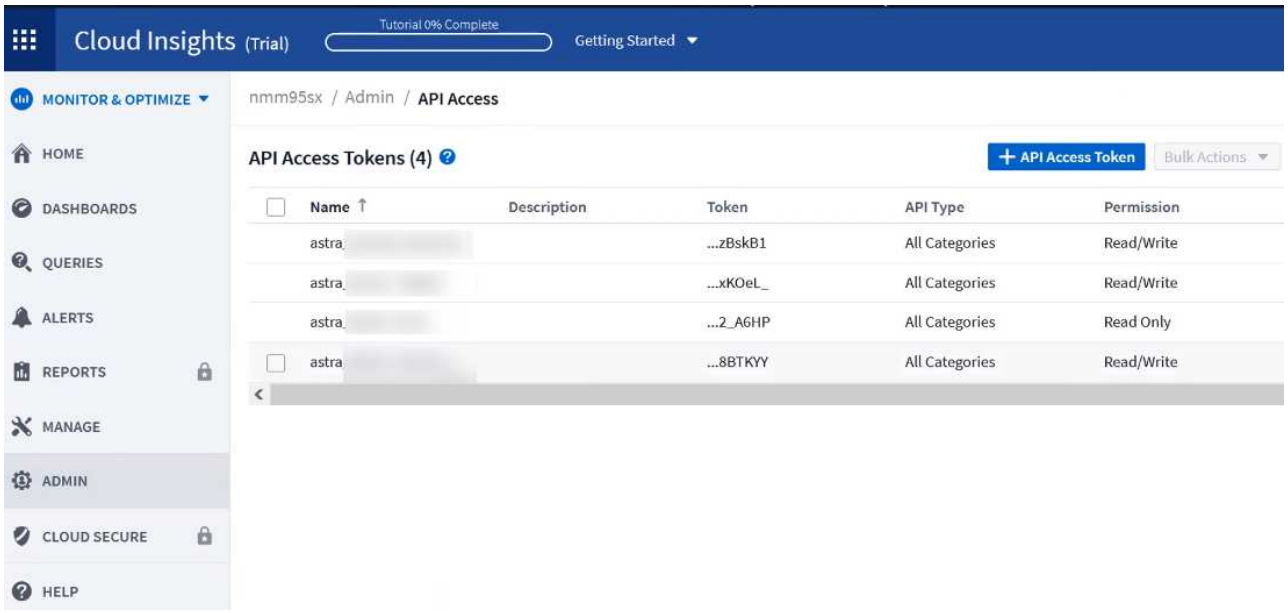
1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 在下拉式清單中選擇*「Connect*（連線*）」顯示*「Disconnected（中斷連線）」的位置、以新增連線。

4. 輸入Cloud Insights 「不再使用API」權杖和租戶URL。租戶URL的格式如下：

```
https://<environment-name>.c01.cloudinsights.netapp.com/
```

當您取得Cloud Insights 不含功能的授權時、就會收到租戶URL。如果您沒有租戶URL、請參閱 "本文檔 Cloud Insights"。

a. 以取得 "API權杖"、登入Cloud Insights 您的URL。
b. 在支援區中、按一下「管理」>「* API存取*」、即可產生*讀取/寫入*和*唯讀* API存取權杖。Cloud Insights



c. 複製*唯讀*金鑰。您必須將其貼到Astra Control Center視窗中、才能啟用Cloud Insights 此功能的鏈路。如需讀取API存取權杖金鑰權限、請選取：資產、警示、擷取單位和資料收集。

d. 複製*讀取/寫入*金鑰。您需要將其貼到Astra Control Center * Connect Cloud Insights S還原*視窗中。如需讀取/寫入API存取權杖金鑰權限、請選取：資料擷取、記錄擷取、擷取設備和資料收集。

> 我們建議您產生*唯讀*金鑰和*讀取/寫入*金鑰、而不要將相同的金鑰用於這兩種用途。根據預設、權杖過期期間設為一年。我們建議您保留預設選項、以便在權杖過期之前提供最長持續時間。如果您的權杖過期、遙測就會停止。

e. 將您從Cloud Insights 整個過程中複製的金鑰貼到Astra Control Center。

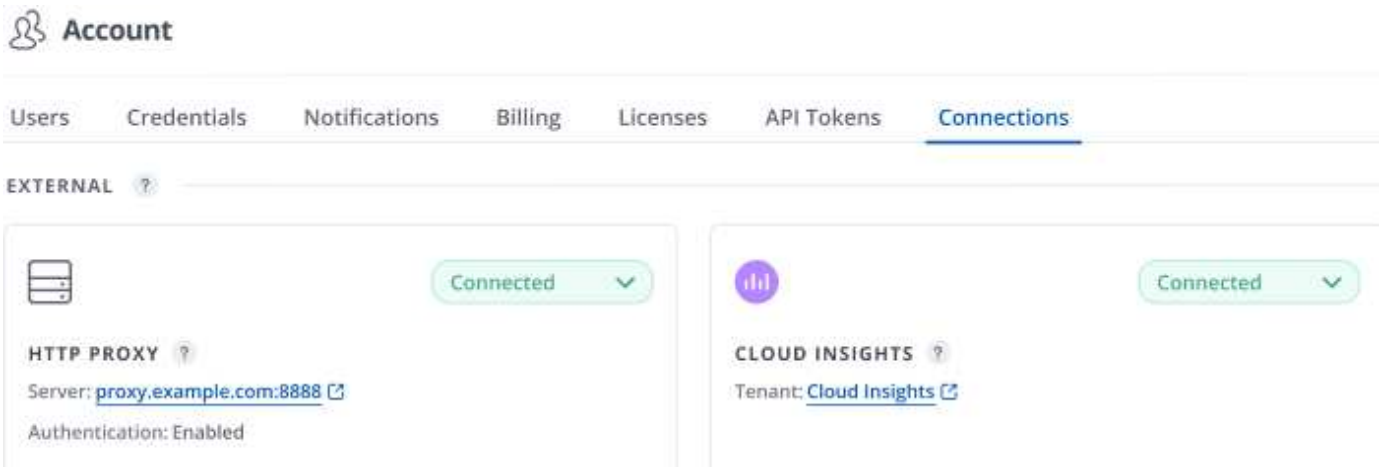
5. 選擇*連接*。

> 在您選取*連線之後、* Cloud Insights 帳戶*>*連線*頁面的*更新*區段中、連線狀態會變更為*擱置*。啟用連線並將狀態變更為「已連線」可能需要幾分鐘的時間。

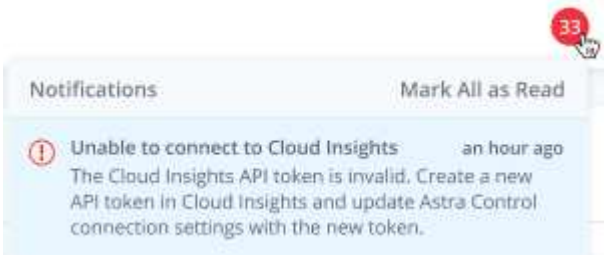
> 若要在Astra Control Center和Cloud Insights UI之間輕鬆來回、請確定您已登入這兩個項目。

檢視**Cloud Insights** 資料

如果連線成功、Cloud Insights 「帳戶>*連線*」頁面的* SURS*區段會指出連線狀態、並顯示租戶URL。您可以造訪Cloud Insights 景點、查看成功接收及顯示的資料。

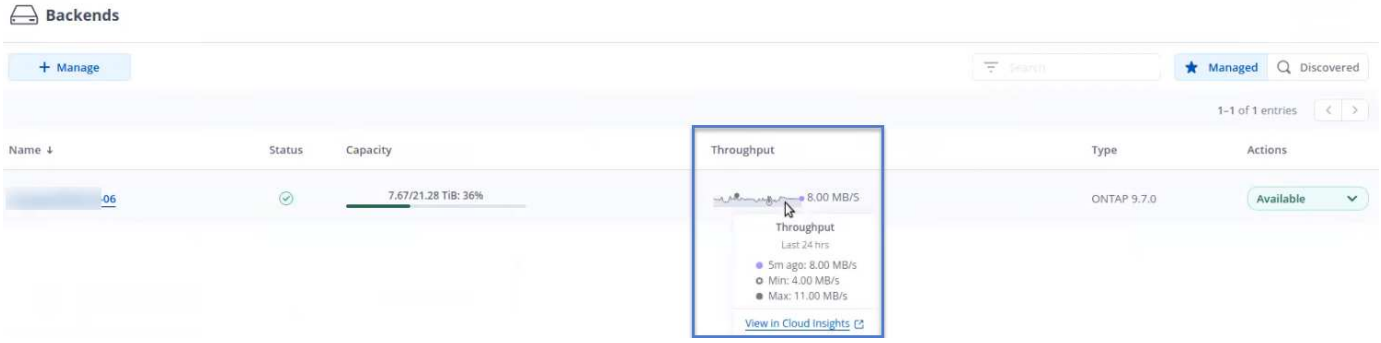


如果連線因為某種原因而失敗、狀態會顯示*失敗*。您可以在UI右上角的*通知*下找到失敗的原因。



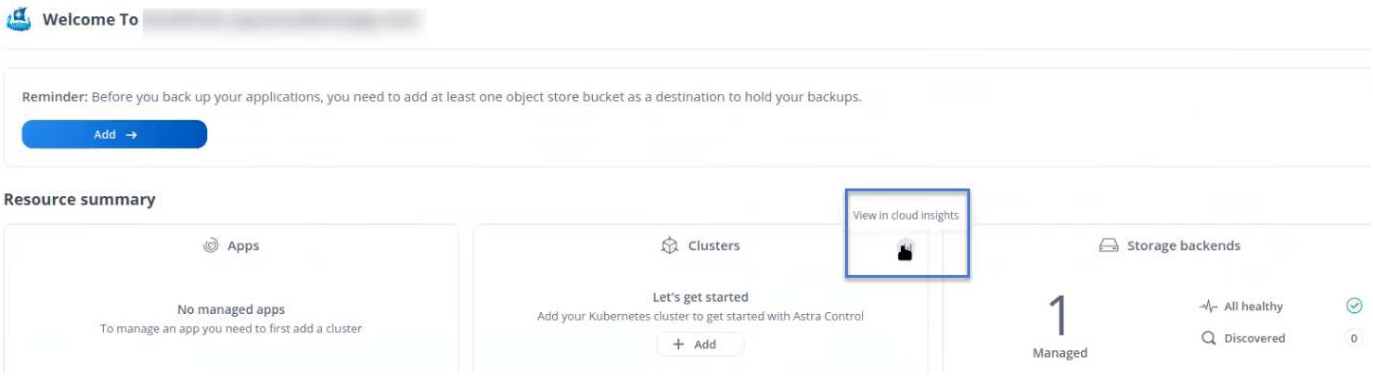
您也可以在*帳戶*>*通知*下找到相同的資訊。

從Astra Control Center、您可以在*後端*頁面上檢視處理量資訊、Cloud Insights 並在選擇儲存後端後端後、從此處連線至



若要直接移至Cloud Insights 「不顯示」、請選取Cloud Insights 「指標」影像旁的*「不顯示」圖示。

您也可以在*儀表板*上找到相關資訊。



啟用Cloud Insights 完「支援不支援」連線後、如果您移除Astra Control Center中新增的後端、後端會停止向Cloud Insights 「支援不支援」回報。

編輯**Cloud Insights** 鏈接

您可以編輯Cloud Insights 此「不同步連線」。

您只能編輯API金鑰。若要變更Cloud Insights 此URL、我們建議您中斷Cloud Insights 連接此鏈接、並使用新的URL進行連線。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取 * 編輯 * 以編輯連線。
4. 編輯Cloud Insights 「還原連線」設定。
5. 選擇*保存*。

停用**Cloud Insights** 鏈接

您可以停用Cloud Insights 由Astra Control Center管理的Kubernetes叢集的支援功能。停用Cloud Insights 此功能不會刪除已上傳至Cloud Insights 更新的遙測資料。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取*「Disconnect*（中斷連線）」以停用連線。
4. 在開啟的對話方塊中、確認作業。
   在您確認操作之後、Cloud Insights 在*帳戶*>*連線*頁面上、顯示的「畫面」狀態會變更為*「待處理」*。狀態變更為*中斷連線*需要幾分鐘的時間。

連線至**Prometheus**

您可以使用Prometheus監控Astra Control Center資料。您可以設定Prometheus從Kubernetes叢集度量端點收集度量、也可以使用Prometheus將度量資料視覺化。

如需使用Prometheus的詳細資訊、請參閱其文件、網址為 "Prometheus入門"。

您需要的產品

請確定您已在Astra Control Center叢集或其他可與Astra Control Center叢集通訊的叢集上下載並安裝Prometheus套件。

請依照正式文件中的指示進行 "安裝Prometheus"。

Prometheus需要能夠與Astra Control Center Kubernetes叢集通訊。如果未在Astra Control Center叢集上安裝Prometheus、您必須確保它們能與Astra Control Center叢集上執行的度量服務通訊。

設定**Prometheus**

Astra Control Center會在Kubernetes叢集中的TCP連接埠9090上公開度量服務。您必須設定Prometheus、才能從此服務收集指標。

步驟

1. 登入Prometheus伺服器。
2. 將叢集項目新增至 `prometheus.yml` 檔案：在中 `yml` 檔案中、針對中的叢集新增類似下列的項目 `scrape_configs section`：

```
job_name: '<Add your cluster name here. You can abbreviate. It just
needs to be a unique name>'
  metrics_path: /accounts/<replace with your account ID>/metrics
  authorization:
     credentials: <replace with your API token>
  tls_config:
     insecure_skip_verify: true
  static_configs:
    - targets: ['<replace with your astraAddress. If using FQDN, the
prometheus server has to be able to resolve it>']
```

如果您設定 `tls_config insecure_skip_verify` 至 `true`、不需要TLS加密傳輸協定。

3. 重新啟動Prometheus服務：

```
sudo systemctl restart prometheus
```

存取**Prometheus**

存取Prometheus URL。

步驟

1. 在瀏覽器中、輸入連接埠9090的Prometheus URL。
2. 選取*狀態*>*目標*來驗證您的連線。

檢視**Prometheus**中的資料

您可以使用Prometheus來檢視Astra Control Center資料。

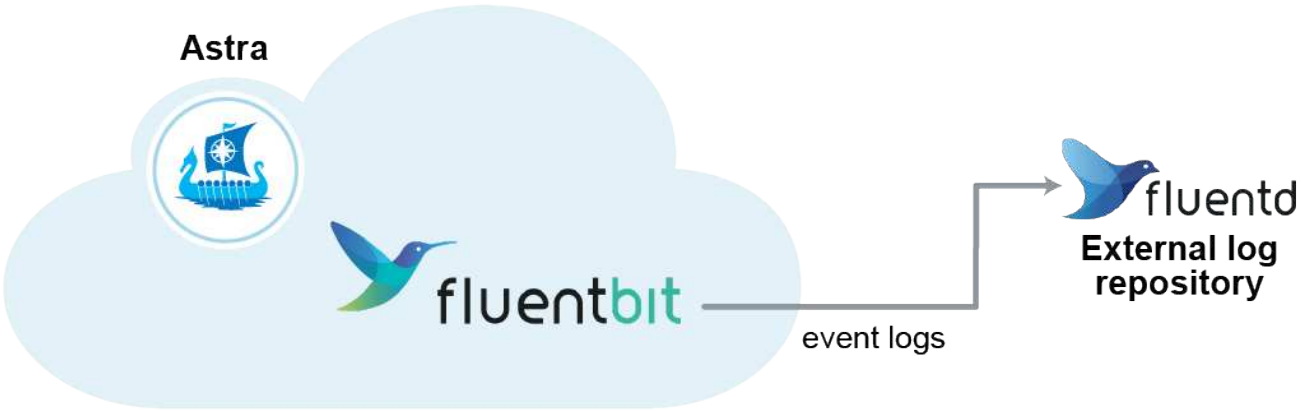
步驟

1. 在瀏覽器中、輸入Prometheus URL。
2. 在Prometheus功能表中、選取* Graph*。
3. 若要使用度量資源管理器、請選取「執行」旁的圖示。
4. 選取 `scrape_samples_scraped` 並選擇*執行*。
5. 若要查看隨時間推移的擷取範例、請選取* Graph*。

如果收集多個叢集資料、每個叢集的度量會以不同的色彩顯示。

連接至Flud

您可以將記錄（ Kubernetes 事件）從 Astra Control Center 監控的系統傳送至 Fluentd 端點。Fluentd連線預設為停用。



只有來自託管叢集的事件記錄會轉送至Fluentd。

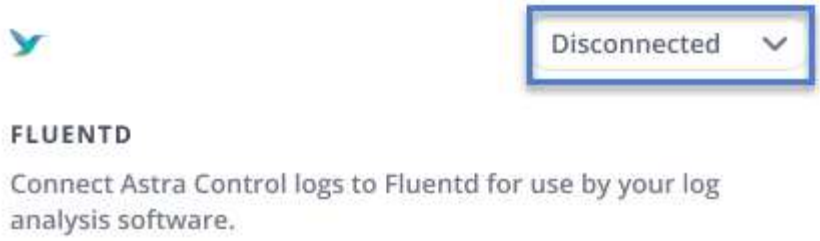
開始之前

- 具有*管理*/*擁有者*權限的Astra Control Center帳戶。
- Astra Control Center安裝並在Kubernetes叢集上執行。

Astra Control Center不會驗證您為Fluentd伺服器輸入的詳細資料。請確認輸入正確的值。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從顯示*中斷連線*的下拉式清單中選取*「Connect*（連線*）」以新增連線。



4. 輸入您的Fluentd伺服器的主機IP位址、連接埠號碼和共用金鑰。
5. 選擇*連接*。

結果

如果您為Fluentd伺服器輸入的詳細資料已儲存、則「帳戶>*連線*」頁面的「變動」區段會指出該資料已連線。現在您可以造訪您所連線的Fluentd伺服器、並檢視事件記錄。

如果連線因為某種原因而失敗、狀態會顯示*失敗*。您可以在UI右上角的*通知*下找到失敗的原因。

您也可以在*帳戶*>*通知*下找到相同的資訊。

如果您在記錄收集方面遇到問題、請登入您的工作節點、並確保中有可用的記錄 `/var/log/containers/`。

編輯**Fluentd**連線

您可以編輯Astra Control Center執行個體的Fluentd連線。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取 * 編輯 * 以編輯連線。
4. 變更Fluentd端點設定。
5. 選擇*保存*。

停用**Fluentd**連線

您可以停用Astra Control Center執行個體的Fluentd連線。

步驟

1. 使用具有*管理*/*擁有者*權限的帳戶登入Astra Control Center。
2. 選擇*帳戶*>*連線*。
3. 從下拉式清單中選取*「Disconnect*（中斷連線）」以停用連線。
4. 在開啟的對話方塊中、確認作業。

取消管理應用程式和叢集

從Astra Control Center移除不再需要管理的任何應用程式或叢集。

取消管理應用程式

停止管理不再想從Astra Control Center備份、快照或複製的應用程式。

當您取消管理應用程式時：

- 任何現有的備份與快照都會刪除。
- 應用程式與資料仍可繼續使用。

步驟

1. 從左側導覽列選取*應用程式*。
2. 選取應用程式。
3. 從「動作」欄的「選項」功能表中、選取*「取消管理」*。
4. 檢閱資訊。
5. 輸入「unManage（取消管理）」以確認。
6. 選擇*是、取消管理應用程式*。

結果

Astra Control Center停止管理應用程式。

取消管理叢集

停止從Astra Control Center管理您不想再管理的叢集。

在取消管理叢集之前、您應該取消管理與叢集相關的應用程式。

當您取消管理叢集時：

- 此動作可防止您的叢集受到Astra Control Center的管理。它不會對叢集的組態進行任何變更、也不會刪除叢集。
- Astra Trident不會從叢集解除安裝。 "瞭解如何解除安裝Astra Trident"。

步驟

1. 從左側導覽列選取*叢集*。
2. 選取您不想再管理之叢集的核取方塊。
3. 從「動作」欄的「選項」功能表中、選取「取消管理」。
4. 確認您要取消管理叢集、然後選取*是、取消管理叢集*。

結果

叢集的狀態會變更為*移除*。之後、叢集就會從「叢集」頁面移除、而且不再由Astra Control Center管理。

*如果Astra Control Center和Cloud Insights Sfunk*未連線、取消叢集管理會移除所有安裝用於傳送遙測資料的資源。*如果Astra Control Center和Cloud Insights Sf1*已連線、則取消管理叢集只會刪除 `fluentbit` 和 `event-exporter` Pod：

升級Astra Control Center

若要升級Astra Control Center、請從NetApp 支援網站 下列網址下載安裝套裝軟體、並完成這些指示。您可以使用此程序、在連線網際網路或無線環境中升級Astra Control Center。

開始之前

升級之前、請確保您的環境仍符合 "Astra Control Center 部署的最低需求"。您的環境應具備下列條件：

- 支援的 Astra Trident 版本

  判斷您正在執行的 Trident 版本：

  ```
  kubectl get tridentversion -n trident
  ```

  請參閱 "Astra Trident文件" 升級舊版。

  您必須升級至Astra Trident 22.10 * PRIOS*、才能升級至Kubernetes 1.25。

- 支援的 Kubernetes 發佈

  判斷您執行的 Kubernetes 版本：

  ```
  kubectl get nodes -o wide
  ```

- 足夠的叢集資源

  判斷可用的叢集資源：

  ```
  kubectl describe node <node name>
  ```

- 您可以用來推送和上傳Astra Control Center映像的登錄
- 預設儲存類別

判斷您的預設儲存類別：

```
kubectl get storageclass
```

- 健全且可用的 API 服務

確保所有API服務均處於健全狀態且可供使用：

```
kubectl get apiservices
```

- （僅限 OpenShift ）健全且可用的叢集操作員

確保所有叢集操作員都處於健全狀態且可用。

```
kubectl get clusteroperators
```

在本程序中、您需要 如果您要升級 Astra Control Center 。您無法使用此更新的運算子升級至舊版 Astra Control Center 。

關於這項工作

Astra Control Center升級程序會引導您完成下列高層級步驟：

在開始升級之前、請先登出Astra Control Center UI。

- 下載並擷取Astra Control Center
- 移除NetApp Astra kubectl外掛程式、然後重新安裝
- [將映像新增至本機登錄]
- 安裝更新的Astra Control Center操作員
- 升級Astra Control Center
- [驗證系統狀態]

請勿刪除Astra Control Center運算子（例如、 `kubectl delete -f astra_control_center_operator_deploy.yaml`）在Astra Control Center升級或操作期間、隨時避免刪除Pod。

當排程、備份和快照未執行時、請在維護期間執行升級。

下載並擷取Astra Control Center

1. 前往 "Astra Control Center產品下載頁面" 於 NetApp 支援網站。您可以從下拉式功能表中選取所需的最新版本或其他版本。
2. （建議但可選）下載Astra Control Center的憑證與簽名套件 (`astra-control-center-certs-[version].tar.gz`）驗證套件的簽名。

   展開以取得詳細資料

   ```
   tar -vxzf astra-control-center-certs-[version].tar.gz
   ```

   ```
   openssl dgst -sha256 -verify certs/AstraControlCenter-public.pub
   -signature certs/astra-control-center-[version].tar.gz.sig astra-
   control-center-[version].tar.gz
   ```

   隨即顯示輸出 `Verified OK` 驗證成功之後。

3. 從Astra Control Center套裝組合擷取映像：

   ```
   tar -vxzf astra-control-center-[version].tar.gz
   ```

移除NetApp Astra kubectl外掛程式、然後重新安裝

您可以使用 NetApp Astra kubectl 命令列外掛程式、將影像推送至本機 Docker 儲存庫。

1. 確定是否安裝了外掛程式：

   ```
   kubectl astra
   ```

2. 請採取下列其中一項行動：
   - 如果已安裝外掛程式、則命令應傳回 kubectl 外掛程式說明、您可以移除現有版本的 kubectl-Astra ：`delete /usr/local/bin/kubectl-astra`。
   - 如果命令傳回錯誤、表示外掛程式尚未安裝、您可以繼續下一步進行安裝。
3. 安裝外掛程式：
   a. 列出可用的NetApp Astra kubectl外掛程式二進位檔、並記下作業系統和CPU架構所需的檔案名稱：

      KECBECTl外掛程式庫是tar套件的一部分、會擷取到資料夾中 `kubectl-astra`。

```
ls kubectl-astra/
```

a. 將正確的二進位檔移至目前路徑、並將其重新命名為 `kubectl-astra`：

```
cp kubectl-astra/<binary-name> /usr/local/bin/kubectl-astra
```

將映像新增至本機登錄

1. 為您的Container引擎完成適當的步驟順序：

Docker

1. 切換到tar檔案的根目錄。您應該會看到 `acc.manifest.bundle.yaml` 檔案與這些目錄：

   ```
   acc/
   kubectl-astra/
   acc.manifest.bundle.yaml
   ```

2. 將Astra Control Center映像目錄中的套件映像推送到本機登錄。執行之前、請先進行下列替換 `push-images` 命令：

   - 以<BUNDLE_FILE> Astra Control套裝組合檔案的名稱取代 (`acc.manifest.bundle.yaml`）。
   - 以<MY_FULL_REGISTRY_PATH> Docker儲存庫的URL取代支援；例如 "<a href="https://<docker-registry>"" class="bare">https://<docker-registry>"。
   - 以<MY_REGISTRY_USER> 使用者名稱取代。
   - 以<MY_REGISTRY_TOKEN> 登錄的授權權杖取代。

     ```
     kubectl astra packages push-images -m <BUNDLE_FILE> -r
     <MY_FULL_REGISTRY_PATH> -u <MY_REGISTRY_USER> -p
     <MY_REGISTRY_TOKEN>
     ```

Podman

1. 切換到tar檔案的根目錄。您應該會看到這個檔案和目錄：

   ```
   acc.manifest.bundle.yaml
   acc/
   ```

2. 登入您的登錄：

   ```
   podman login <YOUR_REGISTRY>
   ```

3. 針對您使用的Podman版本、準備並執行下列其中一個自訂指令碼。以包含任何子目錄的儲存庫URL取代<MY_FULL_REGISTRY_PATH> 。

   ```
   <strong>Podman 4</strong>
   ```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

```
<strong>Podman 3</strong>
```

```
export REGISTRY=<MY_FULL_REGISTRY_PATH>
export PACKAGENAME=acc
export PACKAGEVERSION=23.07.0-25
export DIRECTORYNAME=acc
for astraImageFile in $(ls ${DIRECTORYNAME}/images/*.tar) ; do
astraImage=$(podman load --input ${astraImageFile} | sed 's/Loaded
image: //')
astraImageNoPath=$(echo ${astraImage} | sed 's:.*/::')
podman tag ${astraImageNoPath} ${REGISTRY}/netapp/astra/
${PACKAGENAME}/${PACKAGEVERSION}/${astraImageNoPath}
podman push ${REGISTRY}/netapp/astra/${PACKAGENAME}/${
PACKAGEVERSION}/${astraImageNoPath}
done
```

指令碼所建立的映像路徑應如下所示、視登錄組態而定：

```
https://netappdownloads.jfrog.io/docker-astra-control-
prod/netapp/astra/acc/23.07.0-25/image:version
```

安裝更新的Astra Control Center操作員

1. 變更目錄：

```
cd manifests
```

2. 編輯Astra Control Center營運者部署yaml (`astra_control_center_operator_deploy.yaml`）以參考您的本機登錄和機密。

```
vim astra_control_center_operator_deploy.yaml
```

   a. 如果您使用需要驗證的登錄、請取代或編輯的預設行 `imagePullSecrets: []` 提供下列功能：

```
imagePullSecrets: [{name: astra-registry-cred}]
```

   b. 變更 `ASTRA_IMAGE_REGISTRY` 適用於 `kube-rbac-proxy` 映像到您在中推入映像的登錄路徑 上一步。

   c. 變更 `ASTRA_IMAGE_REGISTRY` 適用於 `acc-operator` 映像到您在中推入映像的登錄路徑 上一步。

   d. 將下列值新增至 `env` 區段：

```
- name: ACCOP_HELM_UPGRADETIMEOUT
  value: 300m
```

Astra 控制中心運算子部署 **.yaml** 範例：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    control-plane: controller-manager
  name: acc-operator-controller-manager
  namespace: netapp-acc-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      control-plane: controller-manager
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        control-plane: controller-manager
    spec:
      containers:
      - args:
        - --secure-listen-address=0.0.0.0:8443
        - --upstream=http://127.0.0.1:8080/
        - --logtostderr=true
        - --v=10
        image: ASTRA_IMAGE_REGISTRY/kube-rbac-proxy:v4.8.0
        name: kube-rbac-proxy
        ports:
        - containerPort: 8443
          name: https
      - args:
        - --health-probe-bind-address=:8081
        - --metrics-bind-address=127.0.0.1:8080
        - --leader-elect
        env:
        - name: ACCOP_LOG_LEVEL
          value: "2"
        - name: ACCOP_HELM_UPGRADETIMEOUT
          value: 300m
        image: ASTRA_IMAGE_REGISTRY/acc-operator:23.07.25
        imagePullPolicy: IfNotPresent
        livenessProbe:
          httpGet:
```

```
            path: /healthz
            port: 8081
          initialDelaySeconds: 15
          periodSeconds: 20
        name: manager
        readinessProbe:
          httpGet:
            path: /readyz
            port: 8081
          initialDelaySeconds: 5
          periodSeconds: 10
        resources:
          limits:
            cpu: 300m
            memory: 750Mi
          requests:
            cpu: 100m
            memory: 75Mi
        securityContext:
          allowPrivilegeEscalation: false
      imagePullSecrets: []
      securityContext:
        runAsUser: 65532
      terminationGracePeriodSeconds: 10
```

3. 安裝更新的Astra Control Center操作員：

```
kubectl apply -f astra_control_center_operator_deploy.yaml
```

回應範例：

```
namespace/netapp-acc-operator unchanged
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.as
tra.netapp.io configured
role.rbac.authorization.k8s.io/acc-operator-leader-election-role
unchanged
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role
configured
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
unchanged
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role
unchanged
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding unchanged
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding configured
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding unchanged
configmap/acc-operator-manager-config unchanged
service/acc-operator-controller-manager-metrics-service unchanged
deployment.apps/acc-operator-controller-manager configured
```

4. 確認Pod正在執行：

```
kubectl get pods -n netapp-acc-operator
```

升級Astra Control Center

1. 編輯Astra Control Center自訂資源（CR）：

```
kubectl edit AstraControlCenter -n [netapp-acc or custom namespace]
```

2. 變更Astra版本號碼 (`astraVersion` 內部 `spec`）升級至您要升級的版本：

```
spec:
  accountName: "Example"
  astraVersion: "[Version number]"
```

3. 確認您的映像登錄路徑符合您在中推送映像的登錄路徑 上一步。更新 `imageRegistry` 內部 `spec` 如果登錄自上次安裝後有所變更。

```
  imageRegistry:
    name: "[your_registry_path]"
```

4. 將下列項目新增至 `crds` 的內部組態 `spec`：

```
crds:
  shouldUpgrade: true
```

5. 在中新增下列行 `additionalValues` 內部 `spec` 在Astra Control Center CR：

```
additionalValues:
    nautilus:
      startupProbe:
        periodSeconds: 30
        failureThreshold: 600
    polaris-keycloak:
      livenessProbe:
        initialDelaySeconds: 180
      readinessProbe:
        initialDelaySeconds: 180
```

6. 儲存並結束檔案編輯器。將套用變更、並開始升級。
7. （可選）驗證Pod是否終止並再次可用：

```
watch kubectl get pods -n [netapp-acc or custom namespace]
```

8. 等待Astra Control狀態狀態顯示升級已完成且準備就緒 (`True`）：

```
kubectl get AstraControlCenter -n [netapp-acc or custom namespace]
```

回應：

```
NAME    UUID                                   VERSION     ADDRESS
READY
astra   9aa5fdae-4214-4cb7-9976-5d8b4c0ce27f   23.07.0-25
10.111.111.111  True
```

若要在作業期間監控升級狀態、請執行下列命令： `kubectl get AstraControlCenter -o yaml -n [netapp-acc or custom namespace]`

若要檢查Astra控制中心的操作員記錄、請執行下列命令：
`kubectl logs deploy/acc-operator-controller-manager -n netapp-acc-operator -c manager -f`

驗證系統狀態

1. 登入Astra Control Center。
2. 確認版本已升級。請參閱UI中的* Support*頁面。
3. 確認您所有的託管叢集和應用程式仍存在且受到保護。

解除安裝Astra Control Center

如果您要從試用版升級至完整版產品、可能需要移除Astra Control Center元件。若要移除Astra Control Center和Astra Control Center操作員、請依序執行本程序中所述的命令。

如果您對解除安裝有任何問題、請參閱 [疑難排解解除安裝問題]。

開始之前

1. "取消管理所有應用程式" 在叢集上。
2. "取消管理所有叢集"。

步驟

1. 刪除Astra Control Center。下列範例命令是根據預設安裝而來。如果您進行自訂組態、請修改命令。

```
kubectl delete -f astra_control_center.yaml -n netapp-acc
```

結果：

```
astracontrolcenter.astra.netapp.io "astra" deleted
```

2. 使用下列命令刪除 `netapp-acc` （或自訂命名）命名空間：

```
kubectl delete ns [netapp-acc or custom namespace]
```

範例結果：

```
namespace "netapp-acc" deleted
```

3. 使用下列命令刪除Astra Control Center作業系統元件：

```
kubectl delete -f astra_control_center_operator_deploy.yaml
```

結果：

```
namespace/netapp-acc-operator deleted
customresourcedefinition.apiextensions.k8s.io/astracontrolcenters.astra.
netapp.io deleted
role.rbac.authorization.k8s.io/acc-operator-leader-election-role deleted
clusterrole.rbac.authorization.k8s.io/acc-operator-manager-role deleted
clusterrole.rbac.authorization.k8s.io/acc-operator-metrics-reader
deleted
clusterrole.rbac.authorization.k8s.io/acc-operator-proxy-role deleted
rolebinding.rbac.authorization.k8s.io/acc-operator-leader-election-
rolebinding deleted
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-manager-
rolebinding deleted
clusterrolebinding.rbac.authorization.k8s.io/acc-operator-proxy-
rolebinding deleted
configmap/acc-operator-manager-config deleted
service/acc-operator-controller-manager-metrics-service deleted
deployment.apps/acc-operator-controller-manager deleted
```

疑難排解解除安裝問題

請使用下列因應措施來解決您在解除安裝Astra Control Center時遇到的任何問題。

解除安裝**Astra Control Center**無法清除受管理叢集上的監控操作員**Pod**

如果在卸載Astra Control Center之前未取消管理叢集、您可以使用下列命令手動刪除NetApp監控命名空間和命名空間中的Pod：

步驟

1. 刪除 `acc-monitoring` 代理程式：

```
kubectl delete agents acc-monitoring -n netapp-monitoring
```

結果：

```
agent.monitoring.netapp.com "acc-monitoring" deleted
```

2. 刪除命名空間：

```
kubectl delete ns netapp-monitoring
```

結果：

```
namespace "netapp-monitoring" deleted
```

3. 確認移除的資源：

```
kubectl get pods -n netapp-monitoring
```

結果：

```
No resources found in netapp-monitoring namespace.
```

4. 確認監控代理程式已移除：

```
kubectl get crd|grep agent
```

結果範例：

```
agents.monitoring.netapp.com                    2021-07-21T06:08:13Z
```

5. 刪除自訂資源定義（CRD）資訊：

```
kubectl delete crds agents.monitoring.netapp.com
```

結果：

```
customresourcedefinition.apiextensions.k8s.io
"agents.monitoring.netapp.com" deleted
```

解除安裝**Astra Control Center**無法清除**Traefik CRD**

您可以手動刪除Traefik客戶需求日。客戶需求日是全域資源、刪除這些資源可能會影響叢集上的其他應用程式。

步驟

1. 列出叢集上安裝的Traefik客戶需求日：

```
kubectl get crds |grep -E 'traefik'
```

回應

```
ingressroutes.traefik.containo.us              2021-06-23T23:29:11Z
ingressroutetcps.traefik.containo.us           2021-06-23T23:29:11Z
ingressrouteudps.traefik.containo.us           2021-06-23T23:29:12Z
middlewares.traefik.containo.us                2021-06-23T23:29:12Z
middlewaretcps.traefik.containo.us             2021-06-23T23:29:12Z
serverstransports.traefik.containo.us          2021-06-23T23:29:13Z
tlsoptions.traefik.containo.us                 2021-06-23T23:29:13Z
tlsstores.traefik.containo.us                  2021-06-23T23:29:14Z
traefikservices.traefik.containo.us            2021-06-23T23:29:15Z
```

2. 刪除客戶需求日：

```
kubectl delete crd ingressroutes.traefik.containo.us
ingressroutetcps.traefik.containo.us
ingressrouteudps.traefik.containo.us middlewares.traefik.containo.us
serverstransports.traefik.containo.us tlsoptions.traefik.containo.us
tlsstores.traefik.containo.us traefikservices.traefik.containo.us
middlewaretcps.traefik.containo.us
```

如需詳細資訊、請參閱

- "解除安裝的已知問題"

利用Astra Control REST API實現自動化

使用Astra Control REST API實現自動化

Astra Control有REST API、可讓您使用程式語言或程式（例如Curl）直接存取Astra Control功能。您也可以使用Ansible和其他自動化技術來管理Astra Control部署。

若要設定及管理Kubernetes應用程式、您可以使用Astra Control Center UI或Astra Control API。

若要深入瞭解、請前往 "Astra自動化文件"。

知識與支援

疑難排解

瞭解如何解決您可能遇到的一些常見問題。

"NetApp Astra知識庫"

如需詳細資訊、請參閱

- "如何將檔案上傳至NetApp（需要登入）"
- "如何手動上傳檔案至NetApp（需要登入）"

取得協助

NetApp以多種方式支援Astra Control。我們全年無休提供豐富的免費自助支援選項、例如知識庫（KB）文章和不和管道。您的Astra Control帳戶透過網路票證提供遠端技術支援。

如果您擁有Astra Control Center的評估授權、您可以取得技術支援。不過，無法透過 NetApp 支援網站（NSS）建立案例。您可以透過意見回饋選項與Support聯絡、或使用不和管道進行自助服務。

您必須優先 "啟動NetApp序號支援" 以使用這些非自助服務支援選項。需有 NetApp 支援網站（NSS）SSO 帳戶，才能進行聊天和網路提交問題單，以及案例管理。

自我支援選項

您可以從主功能表選取* Support*索引標籤、從Astra Control Center UI存取支援選項。

這些選項全年無休免費提供：

- "知識庫（需要登入）"：搜尋與Astra Control相關的文章、常見問題集或中斷修復資訊。
- 文件中心：這是您目前正在檢視的文件網站。
- "*透過不和*取得協助"：前往酒吧類別的Astra、與同儕和專家交流。
- 建立支援案例：產生支援套裝組合、以提供給NetApp支援人員進行疑難排解。
- 針對**Astra Control**提供意見回饋：傳送電子郵件至astra.feedback@netapp.com、讓我們知道您的想法、想法或疑慮。

啟用每日排程支援服務套件上傳至**NetApp**支援

在Astra Control Center安裝期間（如果您指定） `enrolled: true` 適用於 `autoSupport` 在Astra Control Center自訂資源（CR）檔案中 (`astra_control_center.yaml`）、每日支援服務組合會自動上傳至 "NetApp支援網站"。

產生支援套裝組合以提供給NetApp支援部門

Astra Control Center可讓管理員使用者產生套裝組合、其中包含NetApp支援所需的資訊、包括Astra部署所有元件的記錄、事件、度量、以及有關所管理叢集和應用程式的拓撲資訊。如果您已連線至網際網路，可以直接從Astra Control Center UI 將支援套件上傳至 NetApp 支援網站（NSS）。

Astra Control Center產生套裝組合所花費的時間、取決於您的Astra Control Center安裝規模、以及所要求支援套裝組合的參數。您在申請支援服務組合時所指定的時間長度、會決定產生服務組合所需的時間（例如、縮短時間會導致更快產生服務組合）。

開始之前

判斷是否需要代理連線才能將套件上傳至NSS。如果需要Proxy連線、請確認Astra Control Center已設定為使用Proxy伺服器。

1. 選擇*帳戶*>*連線*。
2. 檢查*連線設定*中的Proxy設定。

步驟

1. 使用Astra Control Center UI * Support*頁面上列出的授權序號、在NSS入口網站上建立案例。
2. 使用Astra Control Center UI來產生支援服務組合、請執行下列步驟：
   a. 在* Support*頁面的Support bunds（支援服務）方塊中、選取* Generat*（產生*）。
   b. 在*「產生支援產品組合*」視窗中、選取時間範圍。

      您可以選擇快速或自訂的時間範圍。

      您可以選擇自訂日期範圍、並在日期範圍內指定自訂時間範圍。

   c. 在您進行選擇之後、請選取*確認*。
   d. 選取「**Upload the bundle to the NetApp Support Site when generated**（產生後將套件上傳至**NetApp** 支援網站）」核取方塊。
   e. 選擇*產生產品組合*。

      當支援服務組合準備就緒時、「警示」區域的「帳戶」>「通知」頁面、「活動」頁面、以及「通知」清單中都會顯示通知（可在UI右上角選取圖示來存取）。

如果產生失敗、「產生產品組合」頁面上會出現圖示。選取圖示以查看訊息。

UI右上角的通知圖示提供與支援服務組合相關的事件資訊、例如成功建立服務組合、建立服務組合失敗、無法上傳服務組合、無法下載服務組合等。

如果您安裝的是無線設備

如果您安裝的是無線設備、請在產生「支援」套裝組合之後、執行下列步驟。
當套裝組合可供下載時、「支援」頁面的「支援套裝組合」區段中、「下載」圖示會出現在「產生」旁邊。

步驟

1. 選取「下載」圖示、即可在本機下載套裝組合。
2. 手動將套件上傳至nss.

   您可以使用下列其中一種方法來執行此作業：

   - 使用 "NetApp驗證檔案上傳（需要登入）"。
   - 將套裝組合直接附加至nss.
   - 使用NetApp Active IQ 解決方案。

如需詳細資訊、請參閱

- "如何將檔案上傳至NetApp（需要登入）"
- "如何手動上傳檔案至NetApp（需要登入）"

舊版Astra Control Center文件

您可以取得先前版本的文件。

- "Astra Control Center 23.04 文件"
- "Astra Control Center 22.11 文件"
- "Astra Control Center 22.08 文件"
- "Astra Control Center 22.04 文件"
- "Astra Control Center 21.12 文件"

法律聲明

法律聲明提供版權聲明、商標、專利等存取權限。

版權

"https://www.netapp.com/company/legal/copyright/"

商標

NetApp 、 NetApp 標誌及 NetApp 商標頁面上列出的標章均為 NetApp 、 Inc. 的商標。其他公司與產品名稱可能為其各自所有者的商標。

"https://www.netapp.com/company/legal/trademarks/"

專利

如需最新的 NetApp 擁有專利清單、請參閱：

https://www.netapp.com/pdf.html?item=/media/11887-patentspage.pdf

隱私權政策

"https://www.netapp.com/company/legal/privacy-policy/"

開放原始碼

通知檔案提供有關 NetApp 軟體所使用之協力廠商版權與授權的資訊。

- "Astra Control Center 23.07.0 注意事項"

Astra Control API授權

https://docs.netapp.com/us-en/astra-automation/media/astra-api-license.pdf

版權資訊

Copyright © 2023 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變更本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。 美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 http://www.netapp.com/TM 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。