安全的Astra資料儲存區

Astra Data Store

NetApp
July 19, 2022

目錄

安全的Astra資料儲存區

管理安全性憑證

Astra Data Store在叢集的軟體元件之間使用相互傳輸層安全性（MTLS）加密。每個Astra Data Store叢集都有自我簽署的根CA憑證（「astrads-cert-root」）和中介CA憑證（「astrads-cert」（「叢集名稱」）。這些憑證由Astra Data Store營運者管理；營運者會在每個憑證到期日前7天自動續訂。您也可以手動撤銷憑證。

撤銷憑證

如果Astra Data Store控制器、節點或CA憑證遭入侵、您可以刪除其MTLS機密來撤銷它。當您這麼做時、Astra Data Store營運者會自動發出新的憑證。您可以隨時撤銷Astra Data Store憑證。

如果您撤銷CA憑證、這會撤銷該CA所簽署的任何憑證。

步驟

1. 登入Astra Data Store叢集中的控制器節點。
2. 列出系統上現有的憑證。例如：

```
kubectl get secrets -n astrads-system | grep astrads-cert
```

輸出應類似於下列內容：

```
astrads-cert-astrads-cluster-controller
kubernetes.io/tls     4      6d6h
astrads-cert-astrads-cluster-f23d158
kubernetes.io/tls     4      6d6h
astrads-cert-astrads-ds-dms-astrads-cluster-f23d158
kubernetes.io/tls     4      6d6h
astrads-cert-astrads-ds-support-astrads-cluster-f23d158
kubernetes.io/tls     4      6d6h
astrads-cert-astrads-support-astrads-cluster-f23d158
kubernetes.io/tls     4      6d6h
astrads-cert-root
kubernetes.io/tls     4      6d6h
astrads-cert-sti-net-com
kubernetes.io/tls     5      6d6h
```

3. 在輸出中、記下您需要撤銷的憑證名稱。
4. 使用「kubecll」公用程式來撤銷憑證、並以憑證名稱取代「Certificate _name」（憑證名稱）。例如：

```
kubectl delete secret CERTIFICATE_NAME -n astrads-system
```

現有的憑證會被撤銷、並自動產生新的憑證。

管理外部金鑰

您可以使用一或多個外部金鑰管理伺服器來保護叢集用來存取加密資料的金鑰。外部金鑰管理伺服器是儲存環境中的第三方系統、使用金鑰管理互通性傳輸協定（KMIP）為節點提供金鑰。

Astra Data Store在建立Astra Data Store叢集時、預設會使用內部金鑰提供者啟用靜止軟體加密（sear）功能。

管理金鑰包括下列自訂資源定義（客戶需求日）：

- 適用**DSKeyProvider**：設定外部KMIP伺服器、此伺服器可以是伺服器叢集。
- 《**DSSEARKeyRotate**：從金鑰提供者取得新的金鑰加密金鑰、並提供給Astra Data Store。

您可以執行下列與外部金鑰管理相關的工作：

- [Set up external key management]
- [Check the software encryption at rest status]
- [Change external to internal key management]
- [Rotate keys for security]

設定外部金鑰管理

在Astra Data Store中設定外部金鑰管理時、會使用「kubectl astrads」命令。

您需要叢集或KMIP伺服器上的SSL憑證、才能設定外部金鑰、例如使用OpenSSL。

步驟

1. 準備金鑰提供者用戶端的憑證。包括用戶端憑證、用戶端私密金鑰及信任CA套裝組合。

   您將在叢集或KMIP伺服器上準備SSL憑證、以便設定外部金鑰、例如使用OpenSSL。

2. 登入Astra Data Store叢集中的其中一個節點。
3. 輸入下列kubecll副檔名命令、設定Astra Data Store叢集的金鑰提供者：

```
kubectl-astrads key-provider certs --key key.pem
--client-cert client_cert.pem --ca-cert server_ca.pem
--hostnames=<kmip_server_ip> <key_provider_cr_name>
--namespace astrads-system --cluster <ads_cluster_name>
```

下列範例會針對As叢集「astradse-Cluster-f23d158」設定名為「hashicorp」的外部金鑰提供者。

```
kubectl-astrads key-provider certs --key key.pem
--client-cert client_cert.pem --ca-cert server_ca.pem
--hostnames=10.235.nnn.nnn hashicorp
--namespace astrads-system --cluster astrads-cluster-f23d158
```

1. 將Astra Data Store叢集設定為使用外部金鑰管理程式、透過適用的適用項（適用）。顯示說明。

   ```
   kubectl-astrads clusters sears -h
   ```

   回應：

```
Configure SEARS in AstraDS cluster

Usage:
  astrads clusters sears [flags]

Flags:
  -d, --duration string   Duration for key rotation (default "2160h")
  -h, --help              help for sears


Global Flags:
      --ads-cluster-name string        Name of the ADS Cluster
      --ads-cluster-namespace string   Namespace of the ADS Cluster
      ...
```

下列命令可將Astra Data Store叢集設定為使用「適用的」「適用的DSKeyProvider hashicorp」做為sar的金鑰管理程式。命令也會使用按鍵旋轉時間、預設值為90天（2160小時）。

```
kubectl-astrads clusters sears -d 500h hashicorp
--ads-cluster-name=astrads-cluster-f23d158
--ads-cluster-namespace=astrads-system
```

檢查軟體加密的靜止狀態

您可以在閒置時檢查軟體加密的組態。

步驟

1. 檢查適用的電池。

```
Name:         astrads-cluster-f23d158
Namespace:    astrads-system
Labels:       <none>
Annotations:  <none>
API Version:  astrads.netapp.io/v1beta1
Kind:         AstraDSCluster
...
Spec:
...
  Software Encryption At Rest:
    Ads Key Provider:     hashicorp
    Key Rotation Period:  500h0m0s
...
Status:
...
  Software Encryption At Rest Status:
    Key Active Time:       2022-05-16T15:53:47Z
    Key Provider Name:     hashicorp
    Key Provider UUID:     ccfc2b0b-dd98-5ca4-b778-99debef83550
    Key UUID:              nnnnnnnn-nnnn-nnnn-nnnn-nnnnnnnnnnnn
```

將外部變更為內部金鑰管理

如果您目前使用外部金鑰管理程式、可以將其變更為內部金鑰管理程式。

步驟

1. 移除SoftwareEncryptionAtRest組態、以變更適用的DSCluster CR。
2. （選用）刪除先前的適用的適用選項。

> 不會自動移除先前的金鑰提供者和密碼。

旋轉金鑰以確保安全性

金鑰輪替可強化安全性。依預設、Astra Data Store每90天自動旋轉金鑰一次。您可以變更預設設定。此外、您也可以視需要隨時旋轉按鍵。

設定自動金鑰旋轉

1. 更新CRD中的「適用」參數。

```
kubectl patch astradscluster astrads-cluster-f23d158
-n astrads-system
--type=merge -p '{"spec": {"softwareEncryptionAtRest": {
"keyRotationPeriod": "3000h"}}}'
```

設定隨需金鑰旋轉

1. 建立可旋轉金鑰的適用的適用選項：「Request CR」（建立適用的適用選項）。

```
cat << EOF | kubectl apply -f  -
apiVersion: astrads.netapp.io/v1beta1
kind: AstraDSSEARKeyRotateRequest
metadata:
  name: manual
  namespace: astrads-system
spec:
  cluster: astrads-cluster-f23d158
EOF
```