온보드 키 관리를 구성합니다

ONTAP 9

NetApp
April 24, 2024

목차

온보드 키 관리를 구성합니다

ONTAP 9.6 이상에서 온보드 키 관리를 활성화합니다

Onboard Key Manager를 사용하여 FIPS 드라이브 또는 SED에 대한 클러스터 노드를 인증할 수 있습니다. Onboard Key Manager는 데이터와 동일한 스토리지 시스템의 노드에 인증 키를 제공하는 기본 제공 도구입니다. Onboard Key Manager는 FIPS-140-2 레벨 1을 준수합니다.

Onboard Key Manager를 사용하여 클러스터가 암호화된 데이터에 액세스하는 데 사용하는 키를 보호할 수 있습니다. 암호화된 볼륨 또는 자체 암호화 디스크에 액세스하는 각 클러스터에서 Onboard Key Manager를 활성화해야 합니다.

이 작업에 대해

클러스터에 노드를 추가할 때마다 보안 키 관리자 온보드 활성화 명령을 실행해야 합니다. MetroCluster 구성에서는 먼저 로컬 클러스터에서 보안 키 관리자 온보드 활성화를 실행한 다음 원격 클러스터에서 동일한 암호를 사용하여 보안 키 관리자 온보드 동기화를 실행해야 합니다.

기본적으로 노드를 재부팅할 때는 키 관리자 암호를 입력할 필요가 없습니다. MetroCluster를 제외하고, 사용자가 재부팅 후 암호문을 입력하도록 'cc-mode-enabled=yes' 옵션을 사용할 수 있습니다.

Common Criteria 모드('cc-mode-enabled=yes')에서 Onboard Key Manager를 활성화하면 다음과 같은 방식으로 시스템 동작이 변경됩니다.

- 시스템은 Common Criteria 모드에서 작동 중일 때 연속 실패한 클러스터 암호 시도를 모니터링합니다.

  NSE(NetApp 스토리지 암호화)가 활성화되어 있고 부팅 시 올바른 클러스터 암호를 입력하지 않으면 시스템이 드라이브를 인증할 수 없고 자동으로 재부팅됩니다. 이 문제를 해결하려면 부팅 프롬프트에서 올바른 클러스터 암호를 입력해야 합니다. 시스템이 부팅되면 24시간 동안 클러스터 암호를 매개 변수로 요구하는 명령에 대해 최대 5회 연속 클러스터 암호를 올바르게 입력할 수 있습니다. 제한에 도달한 경우(예: 클러스터 암호를 5회 연속으로 올바르게 입력하지 않은 경우) 24시간 제한 시간이 경과할 때까지 기다리거나 노드를 재부팅하여 제한을 재설정해야 합니다.

- 시스템 이미지 업데이트는 NetApp RSA-3072 코드 서명 인증서와 SHA-384 코드 서명 다이제스트를 함께 사용하여 일반적인 NetApp RSA-2048 코드 서명 인증서와 SHA-256 코드 서명 다이제스트 대신 이미지 무결성을 확인합니다.

  업그레이드 명령은 다양한 디지털 서명을 확인하여 이미지 내용이 변경되거나 손상되지 않았는지 확인합니다. 유효성 검사에 성공하면 이미지 업데이트 프로세스가 다음 단계로 진행되고, 그렇지 않으면 이미지 업데이트가 실패합니다. 시스템 업데이트에 대한 자세한 내용은 ""클러스터 이미지"" man 페이지를 참조하십시오.

Onboard Key Manager는 휘발성 메모리에 키를 저장합니다. 시스템을 재부팅하거나 정지하면 휘발성 메모리 내용이 지워집니다. 정상적인 작동 조건에서는 시스템을 정지하면 30초 이내에 휘발성 메모리 콘텐츠가 지워집니다.

시작하기 전에

- NSE를 외부 키 관리(KMIP) 서버와 함께 사용할 경우 외부 키 관리자 데이터베이스를 삭제해야 합니다.

  "외부 키 관리에서 온보드 키 관리로 전환"

- 이 작업을 수행하려면 클러스터 관리자여야 합니다.
- Onboard Key Manager를 구성하기 전에 MetroCluster 환경을 구성해야 합니다.

단계

1. 키 관리자 설정 명령을 시작합니다.

   '보안 키 관리자 온보드 활성화-cc-모드 사용 예|아니오'

   > 재부팅 후 키 관리자 암호를 입력하도록 하려면 'cc-mode-enabled=yes'를 설정합니다. MetroCluster 구성에서는 '-cc-mode-enabled' 옵션이 지원되지 않습니다. 보안 키매니저 온보드 활성화 명령은 보안 키매니저 설정 명령을 대체합니다.

   다음 예제에서는 재부팅할 때마다 암호를 입력할 필요 없이 키 관리자 설치 명령을 cluster1에서 시작합니다.

   ```
   cluster1::> security key-manager onboard enable

   Enter the cluster-wide passphrase for onboard key management in Vserver
   "cluster1"::    <32..256 ASCII characters long text>
   Reenter the cluster-wide passphrase:    <32..256 ASCII characters long
   text>
   ```

2. 암호문 프롬프트에서 32자에서 256자 사이의 암호문을 입력하거나 64에서 256자 사이의 암호문을 "cc-mode"로 입력합니다.

   > 지정된 ""cc-mode"" 암호가 64자 미만이면 키 관리자 설정 작업에 암호 프롬프트가 다시 표시되기 전에 5초의 지연이 발생합니다.

3. 암호 확인 프롬프트에서 암호를 다시 입력합니다.
4. 인증 키가 생성되었는지 확인합니다.

   '보안 키 관리자 키 쿼리 노드

   > 보안 키-관리자 키 쿼리 명령은 보안 키-관리자 쿼리 키 명령을 대체합니다. 전체 명령 구문은 man 페이지를 참조하십시오.

   다음 예제에서는 "cluster1"에 대해 인증 키가 생성되었는지 확인합니다.

```
cluster1::> security key-manager key query
       Vserver: cluster1
   Key Manager: onboard
          Node: node1

Key Tag                                  Key Type  Restored
-----------------------------------------  --------  --------
node1                                    NSE-AK    yes
    Key ID:
000000000000000002000000000001000c11b3863f78c2273343d7ec5a67762e00000000
00000000
node1                                    NSE-AK    yes
    Key ID:
000000000000000002000000000001006f4e2513353a674305872a4c9f3bf79700000000
00000000

       Vserver: cluster1
   Key Manager: onboard
          Node: node2

Key Tag                                  Key Type  Restored
-----------------------------------------  --------  --------
node1                                    NSE-AK    yes
    Key ID:
000000000000000002000000000001000c11b3863f78c2273343d7ec5a67762e00000000
00000000
node2                                    NSE-AK    yes
    Key ID:
000000000000000002000000000001006f4e2513353a674305872a4c9f3bf79700000000
00000000
```

작업을 마친 후

나중에 사용할 수 있도록 암호를 스토리지 시스템 외부의 안전한 위치에 복사합니다.

모든 키 관리 정보는 클러스터의 복제된 데이터베이스(RDB)에 자동으로 백업됩니다. 또한 재해 발생 시 사용할 수 있도록 정보를 수동으로 백업해야 합니다.

ONTAP 9.5 이전 버전에서 온보드 키 관리를 활성화합니다

Onboard Key Manager를 사용하여 FIPS 드라이브 또는 SED에 대한 클러스터 노드를 인증할 수 있습니다. Onboard Key Manager는 데이터와 동일한 스토리지 시스템의 노드에 인증 키를 제공하는 기본 제공 도구입니다. Onboard Key Manager는 FIPS-140-2 레벨 1을 준수합니다.

Onboard Key Manager를 사용하여 클러스터가 암호화된 데이터에 액세스하는 데 사용하는 키를 보호할 수 있습니다. 암호화된 볼륨 또는 자체 암호화 디스크에 액세스하는 각 클러스터에서 Onboard Key Manager를 활성화해야 합니다.

이 작업에 대해

클러스터에 노드를 추가할 때마다 보안 키 관리자 설정 명령을 실행해야 합니다.

MetroCluster 구성이 있는 경우 다음 지침을 검토하십시오.

- ONTAP 9.5에서는 로컬 클러스터에서 보안 키 관리자 설정, 원격 클러스터에서 보안 키 관리자 설정 -동기화 -MetroCluster -구성 예 를 각각 동일한 암호를 사용하여 실행해야 합니다.
- ONTAP 9.5 이전에는 로컬 클러스터에서 보안 키 관리자 설정을 실행하고 약 20초 정도 기다린 다음 원격 클러스터에서 동일한 암호를 사용하여 보안 키 관리자 설정을 실행해야 합니다.

기본적으로 노드를 재부팅할 때는 키 관리자 암호를 입력할 필요가 없습니다. ONTAP 9.4부터 '-enable-cc-mode yes' 옵션을 사용하여 재부팅 후 사용자가 암호를 입력하도록 할 수 있습니다.

NVE의 경우 '-enable-cc-mode yes'를 설정하면 볼륨 생성 및 볼륨 이동 시작 명령으로 생성한 볼륨이 자동으로 암호화됩니다. 볼륨 만들기에는 -encrypt true를 지정할 필요가 없습니다. 볼륨 이동 시작의 경우 -encrypt-destination true를 지정하지 않아도 됩니다.

> 실패한 암호 구문을 시도한 후에는 노드를 다시 재부팅해야 합니다.

시작하기 전에

- NSE를 외부 키 관리(KMIP) 서버와 함께 사용할 경우 외부 키 관리자 데이터베이스를 삭제해야 합니다.

  "외부 키 관리에서 온보드 키 관리로 전환"

- 이 작업을 수행하려면 클러스터 관리자여야 합니다.
- Onboard Key Manager를 구성하기 전에 MetroCluster 환경을 구성해야 합니다.

단계

1. 키 관리자 설정을 시작합니다.

   '보안 키 관리자 설정-활성화-cc-모드 예|아니오'

   > ONTAP 9.4부터는 사용자가 재부팅 후 키 관리자 암호를 입력하도록 하는 '-enable-cc-mode yes' 옵션을 사용할 수 있습니다. NVE의 경우 '-enable-cc-mode yes'를 설정하면 볼륨 생성 및 볼륨 이동 시작 명령으로 생성한 볼륨이 자동으로 암호화됩니다.

   다음 예제에서는 재부팅할 때마다 암호를 입력할 필요 없이 키 관리자를 cluster1에서 설정하기 시작합니다.

```
cluster1::> security key-manager setup
Welcome to the key manager setup wizard, which will lead you through
the steps to add boot information.

...

Would you like to use onboard key-management? {yes, no} [yes]:
Enter the cluster-wide passphrase:    <32..256 ASCII characters long
text>
Reenter the cluster-wide passphrase:    <32..256 ASCII characters long
text>
```

2. 온보드 키 관리를 구성하라는 메시지가 나타나면 "예"를 입력합니다.
3. 암호문 프롬프트에서 32자에서 256자 사이의 암호문을 입력하거나 64에서 256자 사이의 암호문을 "cc-mode"로 입력합니다.

> 지정된 ""cc-mode"" 암호가 64자 미만이면 키 관리자 설정 작업에 암호 프롬프트가 다시 표시되기 전에 5초의 지연이 발생합니다.

4. 암호 확인 프롬프트에서 암호를 다시 입력합니다.
5. 모든 노드에 대해 키가 구성되었는지 확인합니다.

   보안 키 관리자 키 쇼

   전체 명령 구문은 man 페이지를 참조하십시오.

```
cluster1::> security key-manager key show

Node: node1
Key Store: onboard
Key ID                                                                   Used By
-------------------------------------------------------------------
--------
00000000000000000200000000000010059851742AF2703FC91369B7DB47C4722 NSE-AK
00000000000000000200000000000001008C07CC0AF1EF49E0105300EFC83004BF NSE-AK

Node: node2
Key Store: onboard
Key ID                                                                   Used By
-------------------------------------------------------------------
--------
00000000000000000200000000000010059851742AF2703FC91369B7DB47C4722 NSE-AK
00000000000000000200000000000001008C07CC0AF1EF49E0105300EFC83004BF NSE-AK
```

작업을 마친 후

모든 키 관리 정보는 클러스터의 복제된 데이터베이스(RDB)에 자동으로 백업됩니다.

Onboard Key Manager 암호를 구성할 때마다 재해 발생 시 사용할 수 있도록 정보를 스토리지 시스템 외부의 안전한 위치에 수동으로 백업해야 합니다. 을 참조하십시오 "온보드 키 관리 정보를 수동으로 백업합니다".

FIPS 드라이브 또는 SED(온보드 키 관리)에 데이터 인증 키 할당

'스토리지 암호화 디스크 수정' 명령을 사용하여 데이터 인증 키를 FIPS 드라이브 또는 SED에 할당할 수 있습니다. 클러스터 노드는 이 키를 사용하여 드라이브의 데이터에 액세스합니다.

이 작업에 대해

자체 암호화 드라이브는 인증 키 ID가 기본값이 아닌 값으로 설정된 경우에만 무단 액세스로부터 보호됩니다. 키 ID 0x0이 있는 제조업체 보안 ID(MSID)는 SAS 드라이브의 표준 기본값입니다. NVMe 드라이브의 경우 표준 기본값은 빈 키 ID로 표시되는 null 키입니다. 키 ID를 자체 암호화 드라이브에 할당하면 시스템은 해당 인증 키 ID를 기본값이 아닌 값으로 변경합니다.

시작하기 전에

이 작업을 수행하려면 클러스터 관리자여야 합니다.

단계

1. FIPS 드라이브 또는 SED에 데이터 인증 키 할당:

   'Storage encryption disk modify -disk_disk_ID_-data-key-id_key_ID_'

   전체 명령 구문은 명령에 대한 man 페이지를 참조하십시오.

   ⓘ 'Security key-manager key query-key-type NSE-AK' 명령어를 이용하여 키 ID를 확인할 수 있다.

   ```
   cluster1::> storage encryption disk modify -disk 0.10.* -data-key-id
   00000000000000000200000000000010019215b9738bc7b43d4698c80246db1f4

   Info: Starting modify on 14 disks.
         View the status of the operation by using the
         storage encryption disk show-status command.
   ```

2. 인증 키가 할당되었는지 확인합니다.

   스토리지 암호화 디스크 표시

   전체 명령 구문은 man 페이지를 참조하십시오.

```
cluster1::> storage encryption disk show
Disk    Mode Data Key ID
-----   ----
--------------------------------------------------------------------
0.0.0   data
000000000000000002000000000000010019215b9738bc7b43d4698c80246db1f4
0.0.1   data
000000000000000002000000000000010059851742AF2703FC91369B7DB47C4722
[...]
```