使用ONTAP Cyber Vault 进行数据保护

NetApp data management solutions

NetApp
January 28, 2026

目录

使用ONTAP Cyber Vault 进行数据保护

ONTAP Cyber Vault 概述

需要实施网络保险库的主要驱动威胁是网络攻击（尤其是勒索软件和数据泄露）的日益普遍和日益复杂化。"随着网络钓鱼的增多"以及越来越复杂的凭证窃取方法，用于发起勒索软件攻击的凭证随后可用于访问基础设施系统。在这些情况下，即使是坚固的基础设施系统也面临受到攻击的风险。抵御系统入侵的唯一方法就是将数据保护并隔离在网络保险库中。

NetApp 基于ONTAP的网络保险库为组织提供了全面而灵活的解决方案来保护其最关键的数据资产。通过利用逻辑隔离和强大的强化方法， ONTAP使您能够创建安全、隔离的存储环境，以抵御不断演变的网络威胁。借助ONTAP，您可以确保数据的机密性、完整性和可用性，同时保持存储基础架构的灵活性和效率。

从 2024 年 7 月开始，之前以 PDF 形式发布的技术报告内容已与ONTAP产品文档集成。此外，诸如本文档之类的新技术报告 (TR) 将不再获得 TR 编号。

什么是网络保险库?

网络保险库是一种特定的数据保护技术，涉及将关键数据存储在与主要 IT 基础设施分开的隔离环境中。

“气隙”、*不可变*和*不可删除*的数据存储库，可免受影响主网络的威胁，例如恶意软件、勒索软件甚至内部威胁。可以通过*不可变*和*不可磨灭*的快照实现网络保险库。

使用传统方法的气隙备份涉及创建空间并物理分离主媒体和辅助媒体。通过将媒体移出现场和/或切断连接，不良行为者就无法访问数据。这可以保护数据，但会导致恢复时间变慢。

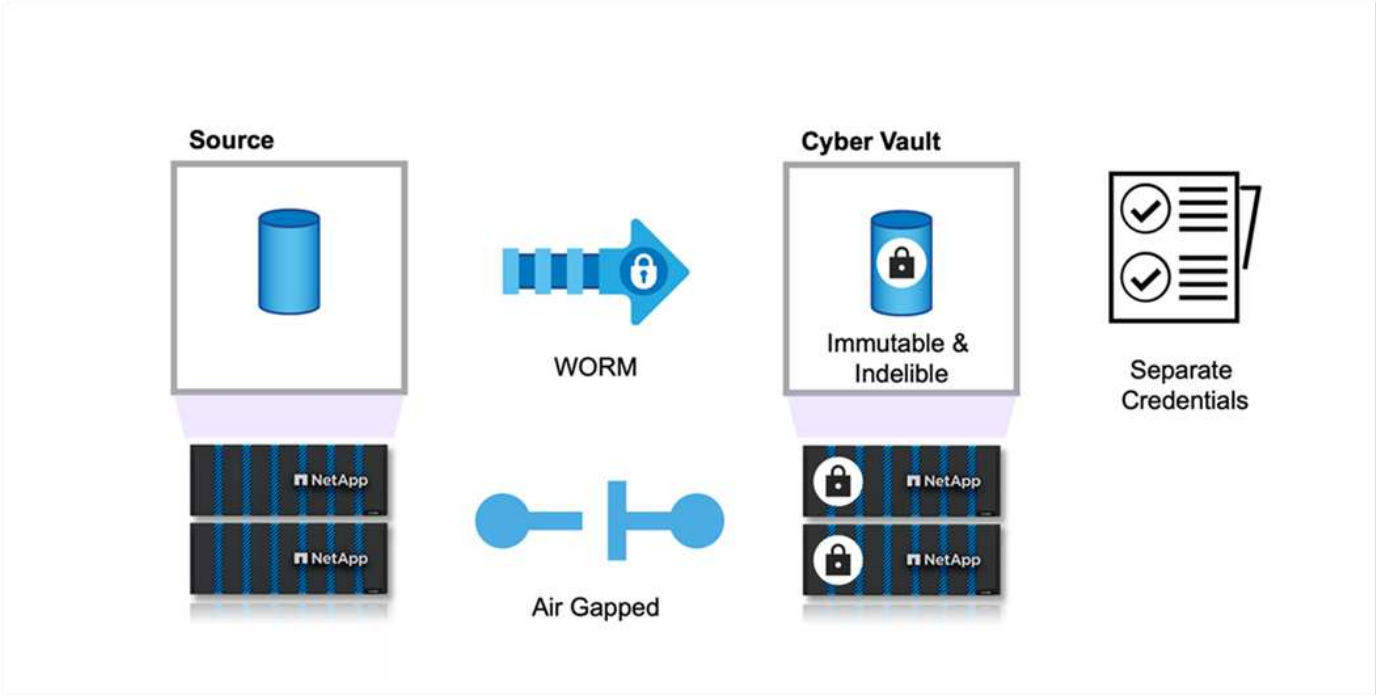
NetApp 的网络保险库方法

NetApp网络保险库参考架构的主要功能包括：

- 安全、独立的存储基础设施（例如，隔离存储系统）
- 数据副本必须毫无例外地*不可变*且*不可删除*
- 严格的访问控制和多因素身份验证
- 快速数据恢复能力

您可以将NetApp存储与ONTAP结合使用，作为隔离网络保险库，方法是利用"SnapLock Compliance对 Snapshot 副本进行 WORM 保护"。您可以在 Cyber Vault 上执行所有基本的SnapLock Compliance任务。配置完成后，Cyber vault 卷将自动受到保护，无需手动将 Snapshot 副本提交到 WORM。有关逻辑气隙的更多信息，请参阅"博客"

SnapLock Compliance用于遵守银行和金融法规 SEC 70-a-4(f)、FINRA 4511(c) 和 CFTC 1.31(c)-(d)。该公司已通过 Cohasset Associates 认证，符合这些规定（可根据要求提供审计报告）。通过使用获得此认证的SnapLock Compliance，您将获得一种强化的数据隔离机制，世界上最大的金融机构依靠该机制来确保银行记录的保留和检索。



Cyber Vault ONTAP术语

这些是网络保险库架构中常用的术语。

自主勒索软件防护 **(ARP)** - 自主勒索软件防护 (ARP) 功能使用 NAS（NFS 和 SMB）环境中的工作负载分析来主动、实时地检测并警告可能表明勒索软件攻击的异常活动。当怀疑受到攻击时，ARP 除了现有的计划 Snapshot 副本保护之外，还会创建新的 Snapshot 副本。有关详细信息，请参阅"ONTAP关于自主勒索软件防护的文档"

气隙（逻辑） - 您可以通过利用以下方式将带有ONTAP的NetApp存储配置为逻辑气隙网络保险库："SnapLock Compliance对 Snapshot 副本进行 WORM 保护"

气隙（物理） - 物理气隙系统没有网络连接。使用磁带备份，您可以将图像移动到另一个位置。 SnapLock Compliance逻辑气隙与物理气隙系统一样强大。

堡垒主机 - 隔离网络上的专用计算机，配置用于抵御攻击。

不可变的 **Snapshot** 副本 - 无法修改的 Snapshot 副本，无一例外（包括支持组织或对存储系统进行低级格式化的能力）。

不可删除的 **Snapshot** 副本 - 无法删除的 Snapshot 副本，无一例外（包括支持组织或对存储系统进行低级格式化的能力）。

防篡改 **Snapshot** 副本 - 防篡改 Snapshot 副本使用SnapLock Compliance时钟功能将 Snapshot 副本锁定指定时间段。任何用户或NetApp支持人员都无法删除这些锁定的快照。如果卷受到勒索软件攻击、恶意软件、黑客、恶意管理员或意外删除的威胁，您可以使用锁定的 Snapshot 副本来恢复数据。有关详细信息，请参阅"有关防篡改 Snapshot 副本的ONTAP文档"

- SnapLock* - SnapLock是一种高性能合规解决方案，适用于使用 WORM 存储以未修改的形式保留文件以满足监管和治理目的的组织。有关更多信息，请参阅"有关SnapLock的ONTAP文档" 。
- SnapMirror* - SnapMirror是一种灾难恢复复制技术，旨在高效复制数据。 SnapMirror可以为本地或云中的

辅助系统创建镜像（或数据的精确副本）、保管库（具有较长 Snapshot 副本保留期的数据副本）或两者。这些副本可用于许多不同的目的，例如灾难、云爆发或网络保险库（使用保险库策略并锁定保险库时）。有关详细信息，请参阅"有关SnapMirror的ONTAP文档"

- SnapVault* - 在ONTAP 9.3 中， SnapVault已被弃用，取而代之的是使用 vault 或 mirror-vault 策略配置SnapMirror 。这个术语虽然仍在使用，但也已被贬低。有关更多信息，请参阅"有关SnapVault的ONTAP文档" 。

使用ONTAP调整网络保险库大小

确定网络保险库的规模需要了解在给定的恢复时间目标 (RTO) 内需要恢复多少数据。许多因素都会影响到正确设计合适规模的网络保险库解决方案。在确定网络保险库的规模时，必须同时考虑性能和容量。

性能大小考虑因素

1. 源平台型号有哪些（FAS v AFF A 系列 v AFF C 系列）？
2. 源和网络库之间的带宽和延迟是多少？
3. 文件大小有多大以及有多少个文件？
4. 您的恢复时间目标是多少？
5. 您需要在 RTO 内恢复多少数据？
6. 网络保险库将吸收多少个SnapMirror扇入关系？
7. 是否会同时发生单次或多次恢复？
8. 这些多次恢复是否会发生在同一个主服务器上？
9. 从保管库恢复期间， SnapMirror是否会复制到保管库？

尺寸示例

以下是不同网络保险库配置的示例。

Platform	AFF A1K	AFF C400	AFF C250	FAS70
Estimated RTO (100TB)	5 HR	18 HR	24 HR	24> HR
Relative cost	High	Moderate	Low	Ultra Low

容量大小考虑因素

ONTAP Cyber Vault 目标卷所需的磁盘空间量取决于多种因素，其中最重要的是源卷中数据的变化率。目标卷上的备份计划和快照计划都会影响目标卷上的磁盘使用情况，并且源卷上的变化率可能不会保持恒定。最好提供额外的存储容量缓冲区，以满足未来最终用户或应用程序行为变化的需求。

在ONTAP中确定 1 个月保留关系的大小需要根据多种因素计算存储需求，包括主数据集的大小、数据变化率（

每日变化率）以及重复数据删除和压缩节省（如果适用）。

以下是分步方法：

第一步是了解使用网络保险库保护的源卷的大小。这是最初复制到网络保险库目的地的基本数据量。接下来，估计数据集的每日变化率。这是每天变化的数据百分比。充分了解数据的动态性至关重要。

例如：

- 主数据集大小 = 5TB
- 每日变化率=5%（0.05）
- 重复数据删除和压缩效率 = 50% (0.50)

现在，让我们来看一下计算过程：

- 计算每日数据变化率：

```
Changed data per day = 5000 * 5% = 250GB
```

- 计算30天的总变化数据：

```
Total changed data in 30 days = 250 GB * 30 = 7.5TB
```

- 计算所需的总存储量：

```
TOTAL = 5TB + 7.5TB = 12.5TB
```

- 应用重复数据删除和压缩节省：

```
EFFECTIVE = 12.5TB * 50% = 6.25TB
```

存储需求摘要

- 如果没有效率：存储 30 天的网络保险库数据将需要 **12.5TB**。
- 效率为 50%：重复数据删除和压缩后需要 **6.25TB** 的存储空间。

> 快照副本可能会因元数据而产生额外的开销，但这通常很小。

> 如果每天进行多次备份，则根据每天拍摄的 Snapshot 副本数量调整计算。

> 考虑数据随时间的增长，以确保规模适合未来。

使用ONTAP创建网络保险库

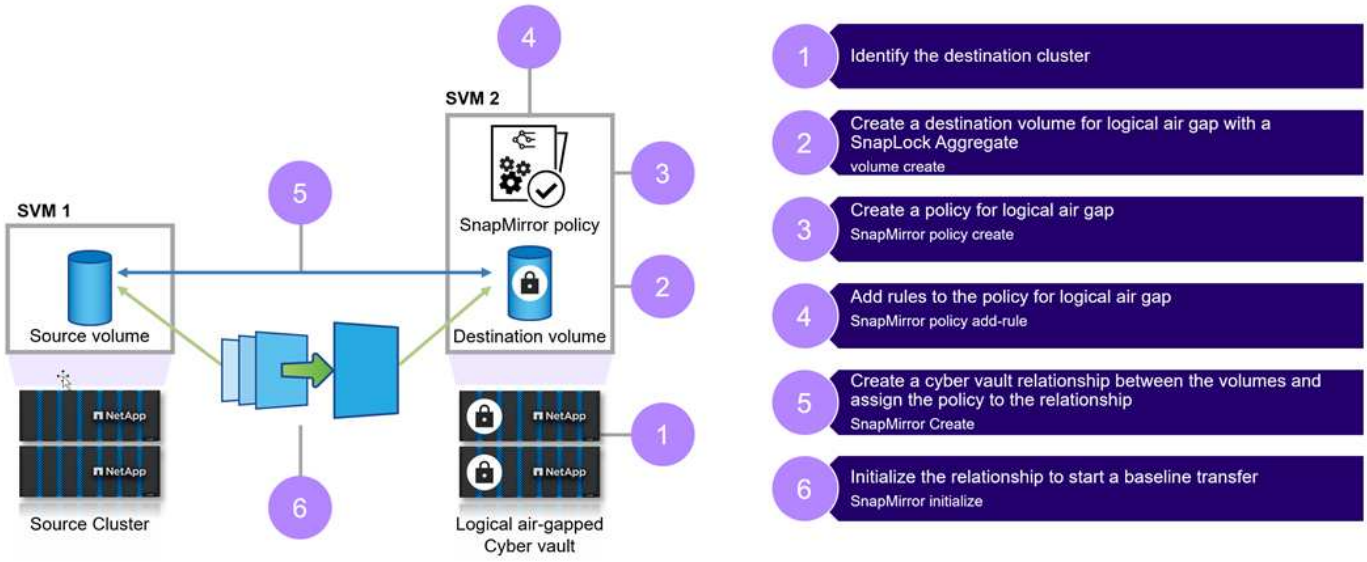
以下步骤将有助于使用ONTAP创建网络保险库。

开始之前

- 源集群必须运行ONTAP 9 或更高版本。
- 源聚合和目标聚合必须是 64 位。
- 必须在具有对等 SVM 的对等集群中创建源卷和目标卷。有关更多信息，请参阅"创建集群对等关系" 。
- 如果禁用卷自动增长，则目标卷上的可用空间必须至少比源卷上的已用空间多 5%。

关于此任务

下图显示了初始化SnapLock Compliance保险库关系的过程：



步骤

1. 确定目标阵列将成为接收隔离数据的网络保险库。
2. 在目标阵列上，准备网络保险库，"安装ONTAP One 许可证" ，"初始化合规时钟" ，如果您使用的是 9.10.1 之前的ONTAP版本，"创建SnapLock Compliance聚合" 。
3. 在目标阵列上，创建 DP 类型的SnapLock Compliance目标卷：

   ```
   volume create -vserver SVM_name -volume volume_name -aggregate aggregate_name
   -snaplock-type compliance|enterprise -type DP -size size
   ```

4. 从ONTAP 9.10.1 开始， SnapLock和非SnapLock卷可以存在于同一个聚合上；因此，如果您使用ONTAP 9.10.1，则不再需要创建单独的SnapLock聚合。您使用音量 `-snaplock-type`选项来指定合规性类型。在ONTAP 9.10.1 之前的ONTAP版本中， SnapLock模式、Compliance 是从聚合继承的。不支持版本灵活的目标卷。目标卷的语言设置必须与源卷的语言设置相匹配。

   以下命令将创建一个名为 2 GB 的SnapLock Compliance卷 `dstvolB`在 `SVM2`总体而言 `node01_aggr`：

   ```
   cluster2::> volume create -vserver SVM2 -volume dstvolB -aggregate node01_aggr
   -snaplock-type compliance -type DP -size 2GB
   ```

5. 在目标集群上，要创建气隙，请设置默认保留期，如中所述"设置默认保留期"。作为保管库目标的SnapLock卷具有分配的默认保留期。此期限的值最初设置为最短 0 年，最长 100 年（从ONTAP 9.10.1 开始）。对于早期版本的ONTAP ，对于SnapLock Compliance卷，该值为 0 - 70。每个NetApp Snapshot 副本最初都遵循此默认保留期。必须更改默认保留期。如果需要，保留期可以延长，但绝不能缩短。有关更多信息，请参

阅"设置保留时间概述" 。

> 服务提供商在确定保留期时应考虑客户的合同结束日期。例如，如果网络保险库的保留期为 30 天，而客户的合同在保留期到期之前结束，则网络保险库中的数据在保留期到期之前不能被删除。

6. "创建新的复制关系"非SnapLock源和您在步骤 3 中创建的新SnapLock目标之间。

   此示例使用 XDPDefault 策略创建与目标SnapLock卷 dstvolB 的新SnapMirror关系，以按小时计划存储标记为每日和每周的 Snapshot 副本：

   ```
   cluster2::> snapmirror create -source-path SVM1:srcvolA -destination-path
   SVM2:dstvolB -vserver SVM2 -policy XDPDefault -schedule hourly
   ```

   "创建自定义复制策略"或"自定义时间表"如果可用的默认值不合适。

7. 在目标 SVM 上，初始化在步骤 5 中创建的SnapVault关系：

   ```
   snapmirror initialize -destination-path destination_path
   ```

8. 以下命令初始化 SVM1 上的源卷 srcvolA 与 SVM2 上的目标卷 dstvolB 之间的关系：

   ```
   cluster2::> snapmirror initialize -destination-path SVM2:dstvolB
   ```

9. 关系初始化并处于空闲状态后，使用目标上的 snapshot show 命令来验证应用于复制的 Snapshot 副本的SnapLock到期时间。

   此示例列出了卷 dstvolB 上具有SnapMirror标签和SnapLock到期日期的 Snapshot 副本：

   ```
   cluster2::> snapshot show -vserver SVM2 -volume dstvolB -fields snapmirror-
   label, snaplock-expiry-time
   ```

网络保险库强化

这些是强化ONTAP网络保险库的额外建议。请参阅下面的ONTAP强化指南以获取更多建议和程序。

网络保险库强化建议

- 隔离网络保险库的管理平面
- 不要在目标集群上启用数据 LIF，因为它们是额外的攻击媒介
- 在目标集群上，使用服务策略限制对源集群的集群间 LIF 访问
- 使用服务策略和堡垒主机对目标集群上的管理 LIF 进行分段，以限制访问
- 限制从源集群到网络保管库的所有数据流量，仅允许SnapMirror流量所需的端口
- 尽可能禁用ONTAP中任何不需要的管理访问方法，以减少攻击面
- 启用审计日志记录和远程日志存储

- 启用多管理员验证，并要求常规存储管理员之外的管理员（例如 CISO 员工）进行验证
- 实施基于角色的访问控制
- 要求系统管理员和 ssh 进行多因素管理身份验证
- 对脚本和 REST API 调用使用基于令牌的身份验证

请参阅"ONTAP强化指南"，"多管理员验证概述"和"ONTAP多因素身份验证指南"如何完成这些强化步骤。

网络保险库互操作性

ONTAP硬件和软件可用于创建网络保险库配置。

ONTAP硬件建议

所有ONTAP统一物理阵列均可用于网络保险库实施。

- FAS混合存储提供了最具成本效益的解决方案。
- AFF C 系列提供最高效的功耗和密度。
- AFF A 系列是性能最高的平台，提供最佳的 RTO。随着我们最新AFF A 系列的发布，该平台将提供最佳的存储效率，同时不影响性能。

ONTAP软件建议

从ONTAP 9.14.1 开始，您可以在SnapMirror关系的SnapMirror策略中为特定SnapMirror标签指定保留期，以便从源卷到目标卷复制的 Snapshot 副本保留规则中指定的保留期。如果未指定保留期，则使用目标卷的默认保留期。

从ONTAP 9.13.1 开始，您可以通过创建FlexClone并将 snaplock-type 选项设置为“non-snaplock”并在执行卷克隆创建操作时将 Snapshot 副本指定为“parent-snapshot”，从而立即恢复SnapLockSnapLock库关系的目标 SnapLock 卷上的锁定 Snapshot 副本。详细了解"创建具有SnapLock类型的FlexClone卷" 。

MetroCluster 配置

对于MetroCluster配置，您应该注意以下事项：

- 您只能在同步源 SVM 之间创建SnapVault关系，而不能在同步源 SVM 和同步目标 SVM 之间创建 SnapVault 关系。
- 您可以创建从同步源 SVM 上的卷到数据服务 SVM 的SnapVault关系。
- 您可以创建从数据服务 SVM 上的卷到同步源 SVM 上的 DP 卷的SnapVault关系。

网络保险库常见问题解答

此常见问题解答适用于NetApp客户和合作伙伴。它回答了有关 NetApp 基于ONTAP的网络保险库参考架构的常见问题。

什么是NetApp网络保险库?

网络保险库是一种特定的数据保护技术，涉及将数据存储在与主要 IT 基础设施分开的隔离环境中。

网络保险库是一个“隔离的”、不可变的、不可磨灭的数据存储库，可以免受影响主要数据的威胁，例如恶意软件、勒索软件或内部威胁。可以使用不可变的NetApp ONTAP Snapshot 副本实现网络保险库，并使用NetApp SnapLock Compliance使其不可磨灭。在SnapLock Compliance保护下，数据无法被修改或删除，即使是ONTAP管理员或NetApp支持人员也无法修改或删除。

使用传统方法的气隙备份涉及创建空间并物理分离主媒体和辅助媒体。网络保险库的气隙隔离包括使用标准数据访问网络之外的单独数据复制网络将 Snapshot 副本复制到不可磨灭的目的地。

除了隔离网络之外，进一步的措施还包括在不需要时禁用网络保险库上的所有数据访问和复制协议。这可以防止目标站点的数据访问或数据泄露。有了SnapLock Compliance，就不需要物理分离。 SnapLock Compliance可保护您存储的、时间点的、只读的 Snapshot 副本，从而实现快速数据恢复，确保数据不会被删除且不可改变。

NetApp 的网络保险库方法

NetApp网络保险库由SnapLock提供支持，为组织提供全面而灵活的解决方案来保护其最关键的数据资产。通过利用ONTAP中的强化技术， NetApp使您能够创建一个安全、隔离且气隙封闭的网络保险库，以抵御不断演变的网络威胁。借助NetApp，您可以确保数据的机密性、完整性和可用性，同时保持存储基础架构的灵活性和效率。

NetApp网络保险库参考架构的主要功能包括：

- 安全、独立的存储基础设施（例如，隔离存储系统）
- 您的数据的备份是不可改变且不可消除的
- 严格且独立的访问控制、多管理员验证和多因素身份验证
- 快速数据恢复能力

网络保险库常见问题解答

cyber vault 是NetApp的产品吗?

不，“网络保险库”是一个行业通用的术语。 NetApp创建了一个参考架构，使客户可以轻松构建自己的网络保险库，并利用数十种ONTAP安全功能来帮助保护他们的数据免受网络威胁。更多信息请访问"ONTAP文档站点"。

NetApp的网络保险库是否只是 LockVault 或SnapVault的另一个名称?

LockVault 是Data ONTAP 7 模式的一项功能，在当前版本的ONTAP中不可用。

SnapVault是一个遗留术语，指的是现在通过 SnapMirror 的保险库策略实现的功能。此策略允许目标保留与源卷不同数量的 Snapshot 副本。

Cyber vault 将SnapMirror与保险库策略和SnapLock Compliance结合使用，以创建不可变且不可消除的数据副本。

我可以将哪种**NetApp**硬件用于网络保险库、 **FAS**、容量闪存或性能闪存?

此网络保险库参考架构适用于整个ONTAP硬件产品组合。客户可以使用AFF A 系列、 AFF C 系列或FAS平台作为保险库。基于闪存的平台将提供最快的恢复时间，而基于磁盘的平台将提供最具成本效益的解决方案。根据要恢复的数据量以及是否同时进行多个恢复，使用基于磁盘的系统（FAS）可能需要几天到几周才能完成。请咨询NetApp或合作伙伴代表，以适当调整网络保险库解决方案的规模来满足业务需求。

我可以使用**Cloud Volumes ONTAP**作为网络保险库源吗?

是的，但是使用 CVO 作为源需要将数据复制到本地网络保险库目标，因为SnapLock Compliance是ONTAP网络保险库的要求。从基于超大规模的 CVO 实例进行数据复制可能会产生出口费用。

我可以使用**Cloud Volumes ONTAP**作为网络保险库目标吗?

Cyber Vault 架构依赖于 ONTAP 的SnapLock Compliance 的不可磨灭性，专为本地实施而设计。基于云的 Cyber Vault 架构正在接受调查，以供将来发布。

我可以使用**ONTAP Select**作为网络保险库源吗?

是的， ONTAP Select可以用作基于本地硬件的网络保险库目标的源。

我可以使用**ONTAP Select**作为网络保险库目标吗?

不可以， ONTAP Select不应用作网络保险库目标，因为它不具备使用SnapLock Compliance 的能力。

NetApp的网络保险库是否仅使用**SnapMirror**?

不是， NetApp网络保险库架构利用许多ONTAP功能来创建安全、隔离、隔离和强化的数据副本。有关可使用哪些附加技术的更多信息，请参阅下一个问题。

网络保险库是否使用了其他技术或配置?

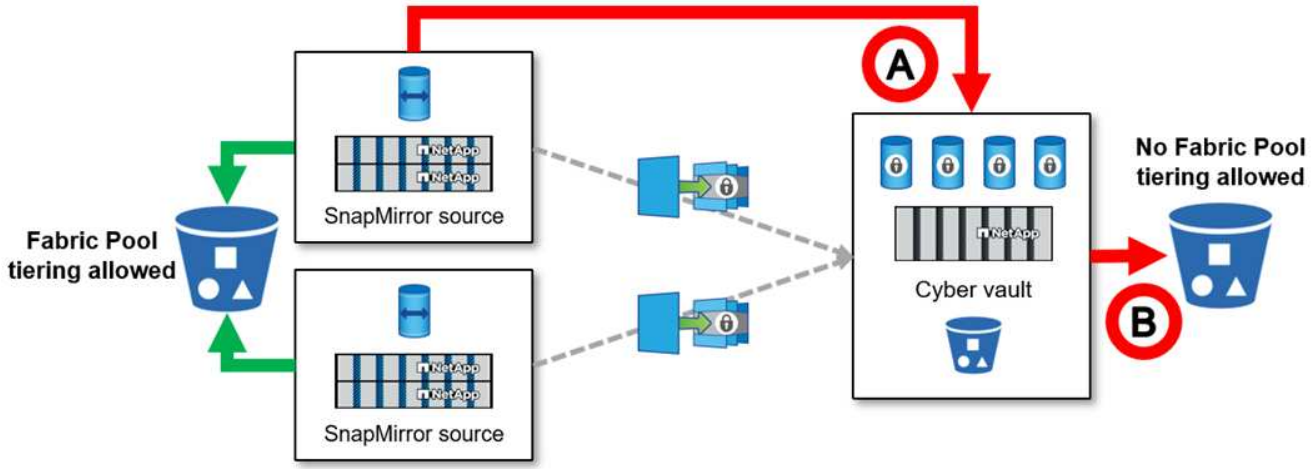
NetApp网络保险库的基础是SnapMirror和SnapLock Compliance，但使用其他ONTAP功能（如防篡改 Snapshot 副本、多因素身份验证 (MFA)、多管理员验证、基于角色的访问控制以及远程和本地审计日志记录）可提高数据的安全性。

对于网络保险库来说， **ONTAP Snapshot** 副本为何比其他副本更胜一筹?

ONTAP Snapshot 副本默认是不可变的，并且可以通过SnapLock Compliance使其不可消除。甚至NetApp支持也无法删除SnapLock Snapshot 副本。更好的问题是，是什么让NetApp网络保险库比业内其他网络保险库更好。首先， ONTAP是地球上最安全的存储，并已获得 CSfC 验证，可以在硬件和软件层静态存储机密和绝密数据。更多信息"CSfC 可在此处找到"。此外， ONTAP可以在存储层进行隔离，由网络保险库系统控制复制，从而可以在网络保险库网络内创建隔离。

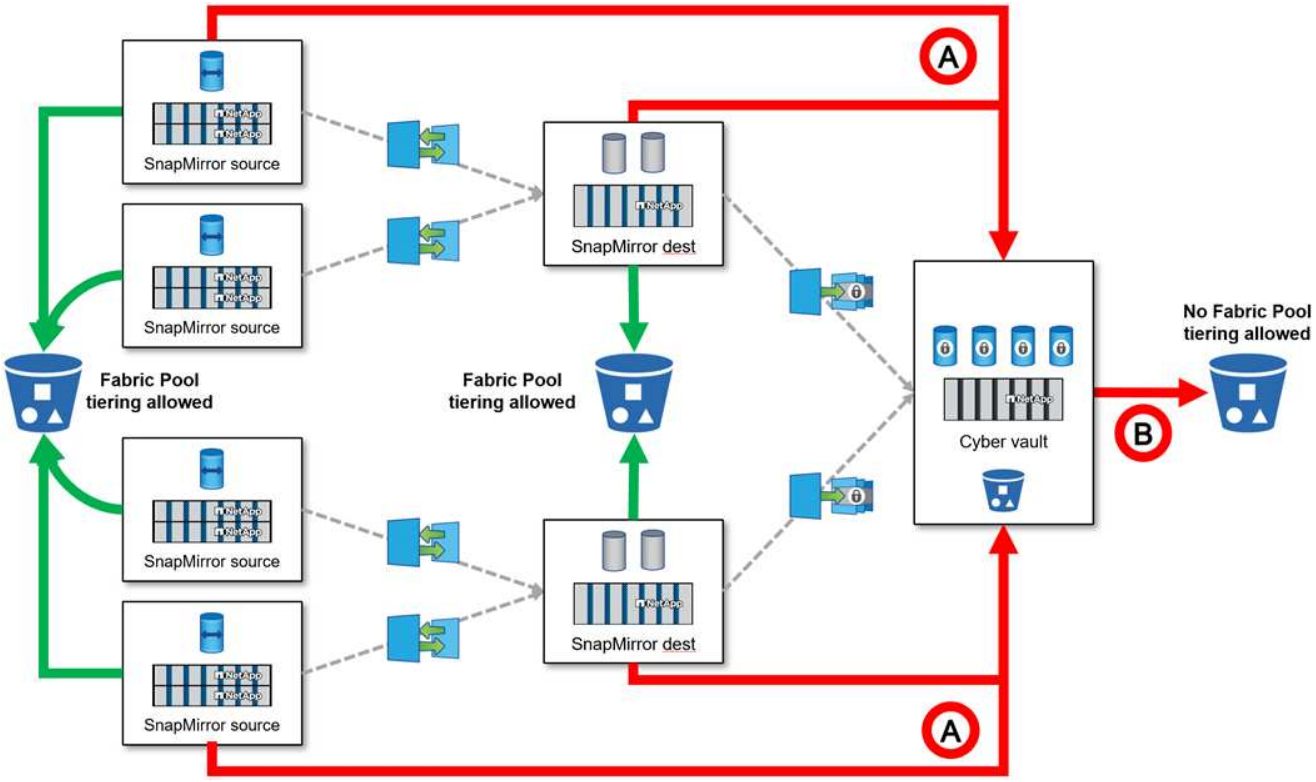
网络保险库上的卷可以使用**ONTAP Fabric** 池吗?

不可以，无论策略如何，网络保险库卷（SnapLock Compliance SnapMirror目标）都不能使用 Fabric Pool 进行分层。



在多种情况下，Fabric 池不能与网络保险库一起使用。

1. Fabric Pool 冷层*不能*使用网络保险库集群。这是因为启用 S3 协议会使网络保险库参考架构的安全性失效。此外，用于 Fabric 池的 S3 存储桶无法受到保护。
2. 由于数据被锁定在卷中，因此网络保险库上的SnapLock Compliance卷*不能*分层到 S3 存储桶。



ONTAP S3 Worm 可以在网络保险库中使用吗?

不，S3 是一种数据访问协议，它使参考架构的安全性无效。

NetApp Cyber Vault 是否在不同的**ONTAP**个性或配置文件上运行?

不，这是一个参考架构。客户可以使用"参考架构"并建立一个网络保险库，或者可以使用"用于创建、强化和验证的 PowerShell 脚本"网络保险库。

我可以在网络保险库中打开 **NFS**、**SMB** 和 **S3** 等数据协议吗?

默认情况下，应禁用网络保险库上的数据协议以确保其安全。但是，可以在网络保险库上启用数据协议来访问数据以进行恢复或在需要时访问数据。这应该是临时进行的，并在恢复完成后禁用。

您可以将现有的**SnapVault**环境转换为网络保险库吗？还是需要重新播种所有内容?

是可以采用一个作为SnapMirror目标的系统（具有保管库策略），禁用数据协议，并根据"ONTAP强化指南"，将其隔离在安全的位置，并按照参考架构中的其他程序使其成为网络保险库，而无需重新播种目的地。

*还有其他问题吗？*请发送电子邮件至 ng-cyber-vault@netapp.com 并提出您的问题！我们将回复您的问题并将其添加到常见问题解答中。

网络保险库资源

要了解有关此网络保险库信息中描述的信息的更多信息，请参阅以下附加信息和安全概念。

- "NetApp网络保险库：多层数据保护解决方案简介"
- "NetApp凭借业界首款 AI 驱动的盒内勒索软件检测解决方案荣获 AAA 评级"
- "利用全球最安全的存储提升网络弹性"
- "ONTAP安全强化指南"
- "NetApp零信任"
- "NetApp网络弹性"
- "NetApp数据保护"
- "使用 CLI 的集群和 SVM 对等概述"
- "SnapVault归档"
- "配置、分析、cron脚本"

使用 **PowerShell** 创建、强化和验证**ONTAP**网络保管库

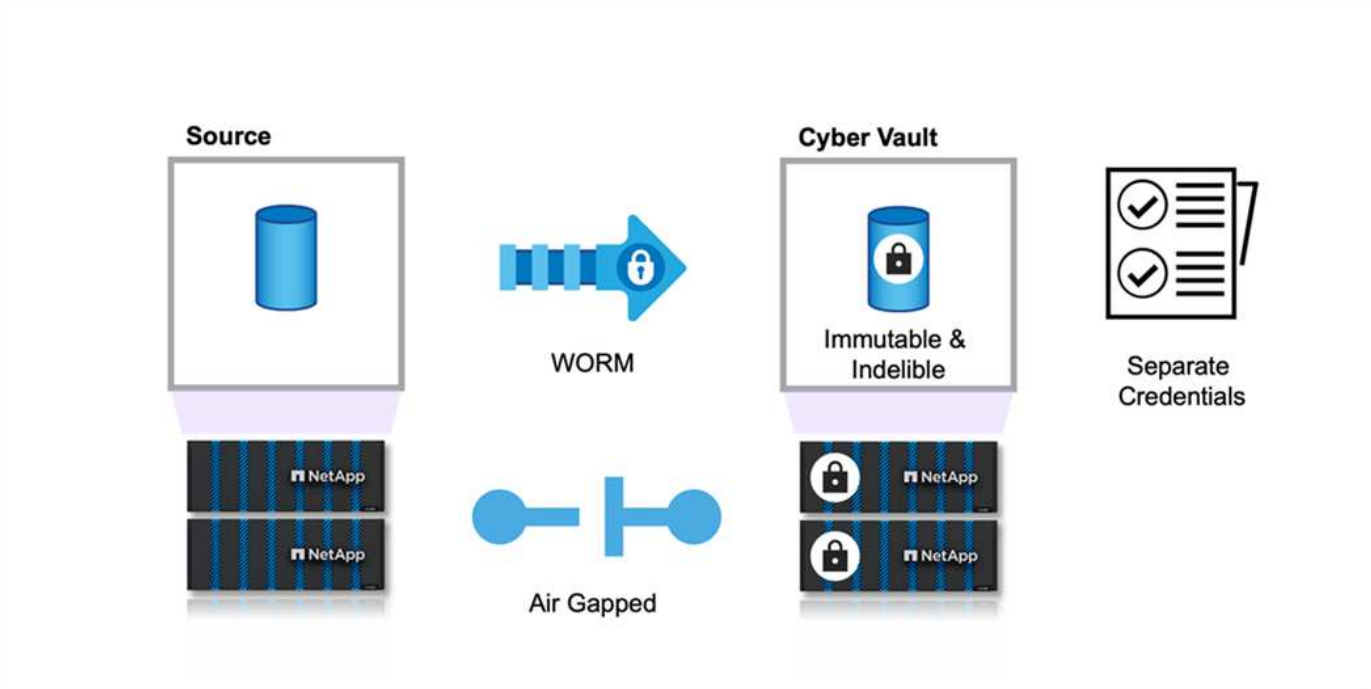
使用 **PowerShell** 的**ONTAP Cyber Vault** 概述

在当今的数字环境中，保护组织的关键数据资产不仅是一种最佳实践，更是业务的当务之急。网络威胁正以前所未有的速度发展，传统的数据保护措施已不足以保证敏感信息的安全。这就是网络保险库的作用所在。NetApp 基于ONTAP的尖端解决方案将先进的隔离技术与强大的数据保护措施相结合，打造出一道抵御网络威胁的坚不可摧的屏障。通过使用安全强化技术隔离最有价值的数据，网络保险库可以最大限度地减少攻击面，从而确保最关键的数据在需要时保持机密、完整且随时可用。

网络保险库是一种安全的存储设施，由防火墙、网络和存储等多层保护组成。这些组件保护关键业务运营所需的重要恢复数据。网络保险库的组件会根据保险库策略定期与基本生产数据同步，但除此之外仍然无法访问。这种隔离且断开的设置可确保在发生危害生产环境的网络攻击时，可以轻松地从网络保险库进行可靠的最终恢复。

NetApp通过配置网络、禁用 LIF、更新防火墙规则以及将系统与外部网络和互联网隔离，轻松为网络保险库创建隔离间隙。这种强大的方法有效地将系统与外部网络和互联网断开，提供无与伦比的保护，防止远程网络攻击和未经授权的访问尝试，使系统免受基于网络的威胁和入侵。

将此与SnapLock Compliance保护相结合，数据无法被修改或删除，即使是ONTAP管理员或NetApp支持人员也无法修改或删除。 SnapLock定期接受 SEC 和 FINRA 法规的审核，确保数据弹性符合银行业严格的 WORM 和数据保留法规。 NetApp是唯一一家经过 NSA CSfC 验证可以存储绝密数据的企业存储公司。

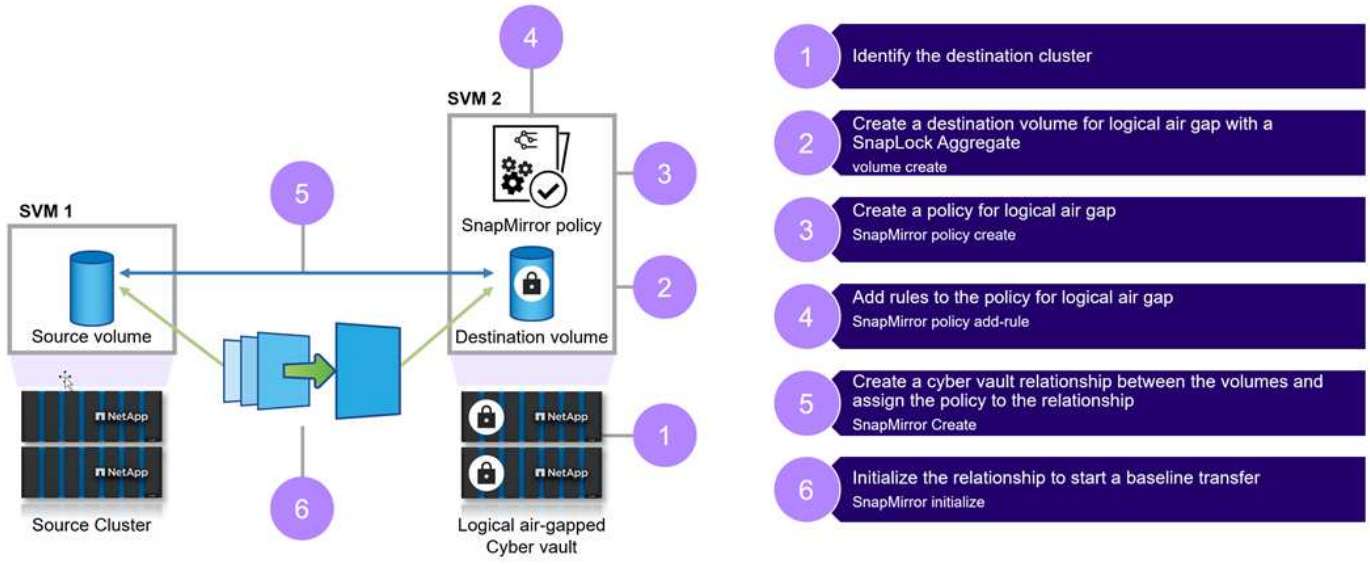


本文档介绍了 NetApp 网络保险库从本地ONTAP存储自动配置到另一个指定ONTAP存储的过程，其中不可变快照增加了额外的保护层，以防止网络攻击，从而实现快速恢复。作为此架构的一部分，整个配置均按照ONTAP最佳实践应用。最后一部分介绍了在发生攻击时执行恢复的说明。

同样的解决方案也适用于使用 FSx ONTAP在 AWS 中创建指定的网络保险库。

创建**ONTAP**网络保险库的高级步骤

- 创建对等关系
  - 使用ONTAP存储的生产站点与指定的网络保管库ONTAP存储对等
- 创建SnapLock Compliance卷
- 设置SnapMirror关系和规则以设置标签
  - 已配置SnapMirror关系和适当的计划
- 在启动SnapMirror （保管库）传输之前设置保留
  - 对复制的数据应用保留锁，进一步防止数据被任何内部人员窃取或数据故障。使用此功能，在保留期到期之前无法删除数据
  - 组织可以根据自己的需求将这些数据保存几周/几个月
- 根据标签初始化SnapMirror关系
  - 初始播种和永久增量传输根据SnapMirror计划进行
  - 数据受到SnapLock合规性的保护（不可变且不可删除），并且数据可供恢复
- 实施严格的数据传输控制
  - 网络保险库在有限的时间内解锁，包含来自生产现场的数据，并与保险库中的数据同步。传输完成后，连接断开、关闭并再次锁定
- 快速恢复
  - 如果生产站点的主要数据受到影响，则网络保管库中的数据可以安全地恢复到原始生产环境或其他选定的环境



解决方案组件

NetApp ONTAP在源集群和目标集群上运行 9.15.1。

ONTAP One： NetApp ONTAP 的一体化许可证。

ONTAP One 许可证使用的功能：

- SnapLock Compliance
- SnapMirror
- 多管理员验证
- ONTAP提供的所有强化功能
- 为网络保险库提供单独的 RBAC 凭证

所有ONTAP统一物理阵列均可用于网络保险库，但基于AFF C 系列容量的闪存系统和FAS混合闪存系统是实现此目的最具成本效益的理想平台。请咨询"ONTAP网络保险库大小调整"以获得尺寸指导。

使用 **PowerShell** 创建**ONTAP**网络保险库

使用传统方法的气隙备份涉及创建空间并物理分离主媒体和辅助媒体。通过将媒体移出现场和/或切断连接，不良行为者就无法访问数据。这可以保护数据，但会导致恢复时间变慢。有了SnapLock Compliance，就不需要物理分离。 SnapLock Compliance可保护存储的快照时间点、只读副本，从而使数据可快速访问、不会被删除或无法擦除，也不会被修改或无法改变。

先决条件

在开始执行本文档下一部分中的步骤之前，请确保满足以下先决条件：

- 源集群必须运行ONTAP 9 或更高版本。
- 源聚合和目标聚合必须是 64 位。
- 源集群和目标集群必须对等。
- 源 SVM 和目标 SVM 必须对等。
- 确保集群对等加密已启用。

设置到ONTAP网络保险库的数据传输需要几个步骤。在主卷上，配置快照策略，使用适当的计划指定要创建哪些副本以及何时创建这些副本，并分配标签以指定应由SnapVault传输哪些副本。在辅助服务器上，必须创建一个SnapMirror策略，指定要传输的 Snapshot 副本的标签以及应在网络保管库中保留多少个副本。配置这些策略后，创建SnapVault关系并建立传输计划。

本文档假设主存储和指定的ONTAP网络保险库已经设置和配置。

Cyber vault 集群可以与源数据位于相同或不同的数据中心。

创建**ONTAP**网络保险库的步骤

1. 使用ONTAP CLI 或系统管理器初始化合规时钟。
2. 创建启用了SnapLock合规性的数据保护卷。
3. 使用SnapMirror create 命令创建SnapVault数据保护关系。
4. 设置目标卷的默认SnapLock Compliance保留期。

默认保留时间为“设置为最小值”。作为保管库目标的SnapLock卷具有分配的默认保留期。此期限的值最初设置为最短 0 年，最长 100 年（从ONTAP 9.10.1 开始）。对于早期版本的ONTAP ，对于SnapLock Compliance卷，该值为 0 - 70。每个NetApp Snapshot 副本最初都遵循此默认保留期。如果需要，保留期可以延长，但绝不能缩短。有关更多信息，请参阅"设置保留时间概述"。

以上包括手动步骤。安全专家建议将流程自动化，以避免手动管理带来的巨大错误。下面是完全自动化SnapLock合规性的先决条件和配置以及时钟初始化的代码片段。

以下是初始化ONTAP合规时钟的 PowerShell 代码示例。

```
function initializeSnapLockComplianceClock {
    try {
        $nodes = Get-NcNode

        $isInitialized = $false
        logMessage -message "Cheking if snaplock compliance clock is
initialized"
        foreach($node in $nodes) {
            $check = Get-NcSnaplockComplianceClock -Node $node.Node
            if ($check.SnaplockComplianceClockSpecified -eq "True") {
                $isInitialized = $true
            }
        }

        if ($isInitialized) {
            logMessage -message "SnapLock Compliance clock already
initialized" -type "SUCCESS"
        } else {
            logMessage -message "Initializing SnapLock compliance clock"
            foreach($node in $nodes) {
                Set-NcSnaplockComplianceClock -Node $node.Node
            }
            logMessage -message "Successfully initialized SnapLock
Compliance clock" -type "SUCCESS"
        }
    } catch {
        handleError -errorMessage $_.Exception.Message
    }
}
```

以下是配置ONTAP网络保险库的 PowerShell 代码示例。

```
function configureCyberVault {
    for($i = 0; $i -lt $DESTINATION_VOLUME_NAMES.Length; $i++) {
        try {
```

```
            # checking if the volume already exists and is of type
snaplock compliance
            logMessage -message "Checking if SnapLock Compliance volume
$($DESTINATION_VOLUME_NAMES[$i]) already exists in vServer
$DESTINATION_VSERVER"
            $volume = Get-NcVol -Vserver $DESTINATION_VSERVER -Volume
$DESTINATION_VOLUME_NAMES[$i] | Select-Object -Property Name, State,
TotalSize, Aggregate, Vserver, Snaplock | Where-Object { $_.Snaplock.Type
-eq "compliance" }
            if($volume) {
                $volume
                logMessage -message "SnapLock Compliance volume
$($DESTINATION_VOLUME_NAMES[$i]) already exists in vServer
$DESTINATION_VSERVER" -type "SUCCESS"
            } else {
                # Create SnapLock Compliance volume
                logMessage -message "Creating SnapLock Compliance volume:
$($DESTINATION_VOLUME_NAMES[$i])"
                New-NcVol -Name $DESTINATION_VOLUME_NAMES[$i] -Aggregate
$DESTINATION_AGGREGATE_NAMES[$i] -SnaplockType Compliance -Type DP -Size
$DESTINATION_VOLUME_SIZES[$i] -ErrorAction Stop | Select-Object -Property
Name, State, TotalSize, Aggregate, Vserver
                logMessage -message "Volume $($DESTINATION_VOLUME_NAMES[
$i]) created successfully" -type "SUCCESS"
            }

            # Set SnapLock volume attributes
            logMessage -message "Setting SnapLock volume attributes for
volume: $($DESTINATION_VOLUME_NAMES[$i])"
            Set-NcSnaplockVolAttr -Volume $DESTINATION_VOLUME_NAMES[$i]
-MinimumRetentionPeriod $SNAPLOCK_MIN_RETENTION -MaximumRetentionPeriod
$SNAPLOCK_MAX_RETENTION -ErrorAction Stop | Select-Object -Property Type,
MinimumRetentionPeriod, MaximumRetentionPeriod
            logMessage -message "SnapLock volume attributes set
successfully for volume: $($DESTINATION_VOLUME_NAMES[$i])" -type "SUCCESS"

            # checking snapmirror relationship
            logMessage -message "Checking if SnapMirror relationship
exists between source volume $($SOURCE_VOLUME_NAMES[$i]) and destination
SnapLock Compliance volume $($DESTINATION_VOLUME_NAMES[$i])"
            $snapmirror = Get-NcSnapmirror | Select-Object SourceCluster,
SourceLocation, DestinationCluster, DestinationLocation, Status,
MirrorState | Where-Object { $_.SourceCluster -eq
$SOURCE_ONTAP_CLUSTER_NAME -and $_.SourceLocation -eq "$($SOURCE_VSERVER)
:$($SOURCE_VOLUME_NAMES[$i])" -and $_.DestinationCluster -eq
$DESTINATION_ONTAP_CLUSTER_NAME -and $_.DestinationLocation -eq "
```

```
$($DESTINATION_VSERVER):$($DESTINATION_VOLUME_NAMES[$i])" -and ($_.Status
-eq "snapmirrored" -or $_.Status -eq "uninitialized") }
            if($snapmirror) {
                $snapmirror
                logMessage -message "SnapMirror relationship already
exists for volume: $($DESTINATION_VOLUME_NAMES[$i])" -type "SUCCESS"
            } else {
                # Create SnapMirror relationship
                logMessage -message "Creating SnapMirror relationship for
volume: $($DESTINATION_VOLUME_NAMES[$i])"
                New-NcSnapmirror -SourceCluster $SOURCE_ONTAP_CLUSTER_NAME
-SourceVserver $SOURCE_VSERVER -SourceVolume $SOURCE_VOLUME_NAMES[$i]
-DestinationCluster $DESTINATION_ONTAP_CLUSTER_NAME -DestinationVserver
$DESTINATION_VSERVER -DestinationVolume $DESTINATION_VOLUME_NAMES[$i]
-Policy $SNAPMIRROR_PROTECTION_POLICY -Schedule $SNAPMIRROR_SCHEDULE
-ErrorAction Stop | Select-Object -Property SourceCluster, SourceLocation,
DestinationCluster, DestinationLocation, Status, Policy, Schedule
                logMessage -message "SnapMirror relationship created
successfully for volume: $($DESTINATION_VOLUME_NAMES[$i])" -type "SUCCESS"
            }

        } catch {
            handleError -errorMessage $_.Exception.Message
        }
    }
}
```

1. 完成上述步骤后，使用SnapLock Compliance和SnapVault 的隔离网络保险库就准备就绪了。

在将快照数据传输到网络保管库之前，必须初始化SnapVault关系。然而，在此之前，必须进行安全强化以确保保险库的安全。

使用 PowerShell 强化ONTAP Cyber Vault

与传统解决方案相比， ONTAP网络保险库具有更好的抵御网络攻击的能力。在设计增强安全性的架构时，考虑减少攻击面的措施至关重要。这可以通过各种方法实现，例如实施强化密码策略、启用 RBAC、锁定默认用户帐户、配置防火墙以及利用审批流程对保险库系统进行任何更改。此外，限制特定 IP 地址的网络访问协议有助于限制潜在的漏洞。

ONTAP提供了一组控制措施，可以强化ONTAP存储。使用"ONTAP的指导和配置设置"帮助组织满足信息系统机密性、完整性和可用性的规定安全目标。

强化最佳实践

手动步骤

1. 创建具有预定义和自定义管理角色的指定用户。

2. 创建新的 IP 空间来隔离网络流量。
3. 创建驻留在新 IP 空间中的新 SVM。
4. 确保防火墙路由策略配置正确，并且所有规则都定期审核并根据需要更新。

ONTAP CLI 或通过自动化脚本

1. 除了多因素身份验证 (MFA) 之外，还使用多管理员验证 (MAV) 来保护管理，从而增强对数据存储 VM 的管理访问的安全性。
2. 启用集群之间“传输中”的标准数据加密。
3. 使用强加密密码保护 SSH 并强制使用安全密码。
4. 启用全局 FIPS。
5. 应禁用 Telnet 和远程 Shell (RSH)。
6. 锁定默认管理员帐户。
7. 禁用数据 LIF 并保护远程访问点。
8. 禁用并删除未使用或多余的协议和服务。
9. 加密网络流量。
10. 设置超级用户和管理角色时使用最小特权原则。
11. 使用允许的 IP 选项限制来自特定 IP 地址的 HTTPS 和 SSH。
12. 根据传输计划停止并恢复复制。

项目 1-4 需要手动干预，例如指定隔离网络、隔离 IP 空间等，并且需要事先执行。有关配置强化的详细信息，请参阅"ONTAP安全强化指南"。其余部分可以轻松实现自动化，以便于部署和监控。这种精心设计的方法的目的是提供一种机制来自动化强化步骤，以确保保险库控制器的未来安全。网络保险库隔离间隙打开的时间范围尽可能短。 SnapVault利用永久增量技术，该技术只会将自上次更新以来的更改移动到网络保险库中，从而最大限度地减少网络保险库必须保持开放的时间。为了进一步优化工作流程，网络保险库的开放与复制计划相协调，以确保最小的连接窗口。

以下是用于强化ONTAP控制器的 PowerShell 代码示例。

```
function removeSvmDataProtocols {
    try {

        # checking NFS service is disabled
        logMessage -message "Checking if NFS service is disabled on
vServer $DESTINATION_VSERVER"
        $nfsService = Get-NcNfsService
        if($nfsService) {
            # Remove NFS
            logMessage -message "Removing NFS protocol on vServer :
$DESTINATION_VSERVER"
            Remove-NcNfsService -VserverContext $DESTINATION_VSERVER
-Confirm:$false
            logMessage -message "NFS protocol removed on vServer :
```

```
$DESTINATION_VSERVER" -type "SUCCESS"
        } else {
            logMessage -message "NFS service is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

        # checking CIFS/SMB server is disabled
        logMessage -message "Checking if CIFS/SMB server is disabled on
vServer $DESTINATION_VSERVER"
        $cifsServer = Get-NcCifsServer
        if($cifsServer) {
            # Remove SMB/CIFS
            logMessage -message "Removing SMB/CIFS protocol on vServer :
$DESTINATION_VSERVER"
            $domainAdministratorUsername = Read-Host -Prompt "Enter Domain
administrator username"
            $domainAdministratorPassword = Read-Host -Prompt "Enter Domain
administrator password" -AsSecureString
            $plainPassword = [Runtime.InteropServices.Marshal
]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($
domainAdministratorPassword))
            Remove-NcCifsServer -VserverContext $DESTINATION_VSERVER
-AdminUsername $domainAdministratorUsername -AdminPassword $plainPassword
-Confirm:$false -ErrorAction Stop
            logMessage -message "SMB/CIFS protocol removed on vServer :
$DESTINATION_VSERVER" -type "SUCCESS"
        } else {
            logMessage -message "CIFS/SMB server is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

        # checking iSCSI service is disabled
        logMessage -message "Checking if iSCSI service is disabled on
vServer $DESTINATION_VSERVER"
        $iscsiService = Get-NcIscsiService
        if($iscsiService) {
            # Remove iSCSI
            logMessage -message "Removing iSCSI protocol on vServer :
$DESTINATION_VSERVER"
            Remove-NcIscsiService -VserverContext $DESTINATION_VSERVER
-Confirm:$false
            logMessage -message "iSCSI protocol removed on vServer :
$DESTINATION_VSERVER" -type "SUCCESS"
        } else {
            logMessage -message "iSCSI service is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
```

```
        }

        # checking FCP service is disabled
        logMessage -message "Checking if FCP service is disabled on
vServer $DESTINATION_VSERVER"
        $fcpService = Get-NcFcpService
        if($fcpService) {
            # Remove FCP
            logMessage -message "Removing FC protocol on vServer :
$DESTINATION_VSERVER"
            Remove-NcFcpService -VserverContext $DESTINATION_VSERVER
-Confirm:$false
            logMessage -message "FC protocol removed on vServer :
$DESTINATION_VSERVER" -type "SUCCESS"
        } else {
            logMessage -message "FCP service is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

    } catch {
        handleError -errorMessage $_.Exception.Message
    }
}

function disableSvmDataLifs {
    try {
        logMessage -message "Finding all data lifs on vServer :
$DESTINATION_VSERVER"
        $dataLifs = Get-NcNetInterface -Vserver $DESTINATION_VSERVER |
Where-Object { $_.Role -contains "data_core" }
        $dataLifs | Select-Object -Property InterfaceName, OpStatus,
DataProtocols, Vserver, Address

        logMessage -message "Disabling all data lifs on vServer :
$DESTINATION_VSERVER"
        # Disable the filtered data LIFs
        foreach ($lif in $dataLifs) {
            $disableLif = Set-NcNetInterface -Vserver $DESTINATION_VSERVER
-Name $lif.InterfaceName -AdministrativeStatus down -ErrorAction Stop
            $disableLif | Select-Object -Property InterfaceName, OpStatus,
DataProtocols, Vserver, Address
        }
        logMessage -message "Disabled all data lifs on vServer :
$DESTINATION_VSERVER" -type "SUCCESS"

    } catch {
```

```
        handleError -errorMessage $_.Exception.Message
    }
}

function configureMultiAdminApproval {
    try {

        # check if multi admin verification is enabled
        logMessage -message "Checking if multi-admin verification is
enabled"
        $maaConfig = Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "set -privilege advanced;
security multi-admin-verify show"
        if ($maaConfig.Value -match "Enabled" -and $maaConfig.Value -match
"true") {
            $maaConfig
            logMessage -message "Multi-admin verification is configured
and enabled" -type "SUCCESS"
        } else {
            logMessage -message "Setting Multi-admin verification rules"
            # Define the commands to be restricted
            $rules = @(
                "cluster peer delete",
                "vserver peer delete",
                "volume snapshot policy modify",
                "volume snapshot rename",
                "vserver audit modify",
                "vserver audit delete",
                "vserver audit disable"
            )
            foreach($rule in $rules) {
                Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "security multi-admin-verify
rule create -operation `"$rule`""
            }

            logMessage -message "Creating multi admin verification group
for ONTAP Cluster $DESTINATION_ONTAP_CLUSTER_MGMT_IP, Group name :
$MULTI_ADMIN_APPROVAL_GROUP_NAME, Users : $MULTI_ADMIN_APPROVAL_USERS,
Email : $MULTI_ADMIN_APPROVAL_EMAIL"
            Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "security multi-admin-verify
approval-group create -name $MULTI_ADMIN_APPROVAL_GROUP_NAME -approvers
$MULTI_ADMIN_APPROVAL_USERS -email `"$MULTI_ADMIN_APPROVAL_EMAIL`""
            logMessage -message "Created multi admin verification group
for ONTAP Cluster $DESTINATION_ONTAP_CLUSTER_MGMT_IP, Group name :
```

```
$MULTI_ADMIN_APPROVAL_GROUP_NAME, Users : $MULTI_ADMIN_APPROVAL_USERS,
Email : $MULTI_ADMIN_APPROVAL_EMAIL" -type "SUCCESS"

            logMessage -message "Enabling multi admin verification group
$MULTI_ADMIN_APPROVAL_GROUP_NAME"
            Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "security multi-admin-verify
modify -approval-groups $MULTI_ADMIN_APPROVAL_GROUP_NAME -required
-approvers 1 -enabled true"
            logMessage -message "Enabled multi admin verification group
$MULTI_ADMIN_APPROVAL_GROUP_NAME" -type "SUCCESS"

            logMessage -message "Enabling multi admin verification for
ONTAP Cluster $DESTINATION_ONTAP_CLUSTER_MGMT_IP"
            Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "security multi-admin-verify
modify -enabled true"
            logMessage -message "Successfully enabled multi admin
verification for ONTAP Cluster $DESTINATION_ONTAP_CLUSTER_MGMT_IP" -type
"SUCCESS"

            logMessage -message "Enabling multi admin verification for
ONTAP Cluster $DESTINATION_ONTAP_CLUSTER_MGMT_IP"
            Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "security multi-admin-verify
modify -enabled true"
            logMessage -message "Successfully enabled multi admin
verification for ONTAP Cluster $DESTINATION_ONTAP_CLUSTER_MGMT_IP" -type
"SUCCESS"
        }

    } catch {
        handleError -errorMessage $_.Exception.Message
    }
}

function additionalSecurityHardening {
    try {
        $command = "set -privilege advanced -confirmations off;security
protocol modify -application telnet -enabled false;"
        logMessage -message "Disabling Telnet"
        Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP -Credential
$DESTINATION_ONTAP_CREDS -Command $command
        logMessage -message "Disabled Telnet" -type "SUCCESS"

        #$command = "set -privilege advanced -confirmations off;security
```

```
config modify -interface SSL -is-fips-enabled true;"
        #logMessage -message "Enabling Global FIPS"
        ##Invoke-SSHCommand -SessionId $sshSession.SessionId -Command
$command -ErrorAction Stop
        #logMessage -message "Enabled Global FIPS" -type "SUCCESS"

        $command = "set -privilege advanced -confirmations off;network
interface service-policy modify-service -vserver cluster2 -policy default-
management -service management-https -allowed-addresses $ALLOWED_IPS;"
        logMessage -message "Restricting IP addresses $ALLOWED_IPS for
Cluster management HTTPS"
        Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP -Credential
$DESTINATION_ONTAP_CREDS -Command $command
        logMessage -message "Successfully restricted IP addresses
$ALLOWED_IPS for Cluster management HTTPS" -type "SUCCESS"

        #logMessage -message "Checking if audit logs volume audit_logs
exists"
        #$volume = Get-NcVol -Vserver $DESTINATION_VSERVER -Name
audit_logs -ErrorAction Stop

        #if($volume) {
        #    logMessage -message "Volume audit_logs already exists!
Skipping creation"
        #} else {
        #    # Create audit logs volume
        #    logMessage -message "Creating audit logs volume : audit_logs"
        #    New-NcVol -Name audit_logs -Aggregate
$DESTINATION_AGGREGATE_NAME -Size 5g -ErrorAction Stop | Select-Object
-Property Name, State, TotalSize, Aggregate, Vserver
        #    logMessage -message "Volume audit_logs created successfully"
-type "SUCCESS"
        #}

        ## Mount audit logs volume to path /vol/audit_logs
        #logMessage -message "Creating junction path for volume audit_logs
at path /vol/audit_logs for vServer $DESTINATION_VSERVER"
        #Mount-NcVol -VserverContext $DESTINATION_VSERVER -Name audit_logs
-JunctionPath /audit_logs | Select-Object -Property Name, -JunctionPath
        #logMessage -message "Created junction path for volume audit_logs
at path /vol/audit_logs for vServer $DESTINATION_VSERVER" -type "SUCCESS"

        #logMessage -message "Enabling audit logging for vServer
$DESTINATION_VSERVER at path /vol/audit_logs"
        #$command = "set -privilege advanced -confirmations off;vserver
audit create -vserver $DESTINATION_VSERVER -destination /audit_logs
```

```
-format xml;"
        #Invoke-SSHCommand -SessionI  $sshSession.SessionId -Command
$command -ErrorAction Stop
        #logMessage -message "Successfully enabled audit logging for
vServer $DESTINATION_VSERVER at path /vol/audit_logs"

    } catch {
        handleError -errorMessage $_.Exception.Message
    }
}
```

使用 PowerShell 验证ONTAP网络保险库

强大的网络保险库应该能够抵御复杂的攻击，即使攻击者拥有以提升的权限访问环境的凭证。

一旦规则到位，尝试（假设攻击者能够以某种方式进入）删除保险库端的快照将会失败。通过施加必要的限制和保护系统，所有强化设置均适用相同的规定。

PowerShell 代码示例，用于按计划验证配置。

```
function analyze {

    for($i = 0; $i -lt $DESTINATION_VOLUME_NAMES.Length; $i++) {
        try {
            # checking if volume is of type SnapLock Compliance
            logMessage -message "Checking if SnapLock Compliance volume
$($DESTINATION_VOLUME_NAMES[$i]) exists in vServer $DESTINATION_VSERVER"
            $volume = Get-NcVol -Vserver $DESTINATION_VSERVER -Volume
$DESTINATION_VOLUME_NAMES[$i] | Select-Object -Property Name, State,
TotalSize, Aggregate, Vserver, Snaplock | Where-Object { $_.Snaplock.Type
-eq "compliance" }
            if($volume) {
                $volume
                logMessage -message "SnapLock Compliance volume
$($DESTINATION_VOLUME_NAMES[$i]) exists in vServer $DESTINATION_VSERVER"
-type "SUCCESS"
            } else {
                handleError -errorMessage "SnapLock Compliance volume
$($DESTINATION_VOLUME_NAMES[$i]) does not exist in vServer
$DESTINATION_VSERVER. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to create and configure the cyber vault SnapLock Compliance
volume"
            }

            # checking SnapMirror relationship
```

```
            logMessage -message "Checking if SnapMirror relationship
exists between source volume $($SOURCE_VOLUME_NAMES[$i]) and destination
SnapLock Compliance volume $($DESTINATION_VOLUME_NAMES[$i])"
            $snapmirror = Get-NcSnapmirror | Select-Object SourceCluster,
SourceLocation, DestinationCluster, DestinationLocation, Status,
MirrorState | Where-Object { $_.SourceCluster -eq
$SOURCE_ONTAP_CLUSTER_NAME -and $_.SourceLocation -eq "$($SOURCE_VSERVER)
:$($SOURCE_VOLUME_NAMES[$i])" -and $_.DestinationCluster -eq
$DESTINATION_ONTAP_CLUSTER_NAME -and $_.DestinationLocation -eq "
$($DESTINATION_VSERVER):$($DESTINATION_VOLUME_NAMES[$i])" -and $_.Status
-eq "snapmirrored" }
            if($snapmirror) {
                $snapmirror
                logMessage -message "SnapMirror relationship successfully
configured and in healthy state" -type "SUCCESS"
            } else {
                handleError -errorMessage "SnapMirror relationship does
not exist between the source volume $($SOURCE_VOLUME_NAMES[$i]) and
destination SnapLock Compliance volume $($DESTINATION_VOLUME_NAMES[$i])
(or) SnapMirror status uninitialized/unhealthy. Recommendation: Run the
script with SCRIPT_MODE `"configure`" to create and configure the cyber
vault SnapLock Compliance volume and configure the SnapMirror
relationship"
            }
        }
        catch {
            handleError -errorMessage $_.Exception.Message
        }
    }

    try {

        # checking NFS service is disabled
        logMessage -message "Checking if NFS service is disabled on
vServer $DESTINATION_VSERVER"
        $nfsService = Get-NcNfsService
        if($nfsService) {
            handleError -errorMessage "NFS service running on vServer
$DESTINATION_VSERVER. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to disable NFS on vServer $DESTINATION_VSERVER"
        } else {
            logMessage -message "NFS service is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

        # checking CIFS/SMB server is disabled
```

```
        logMessage -message "Checking if CIFS/SMB server is disabled on
vServer $DESTINATION_VSERVER"
        $cifsServer = Get-NcCifsServer
        if($cifsServer) {
            handleError -errorMessage "CIFS/SMB server running on vServer
$DESTINATION_VSERVER. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to disable CIFS/SMB on vServer $DESTINATION_VSERVER"
        } else {
            logMessage -message "CIFS/SMB server is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

        # checking iSCSI service is disabled
        logMessage -message "Checking if iSCSI service is disabled on
vServer $DESTINATION_VSERVER"
        $iscsiService = Get-NcIscsiService
        if($iscsiService) {
            handleError -errorMessage "iSCSI service running on vServer
$DESTINATION_VSERVER. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to disable iSCSI on vServer $DESTINATION_VSERVER"
        } else {
            logMessage -message "iSCSI service is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

        # checking FCP service is disabled
        logMessage -message "Checking if FCP service is disabled on
vServer $DESTINATION_VSERVER"
        $fcpService = Get-NcFcpService
        if($fcpService) {
            handleError -errorMessage "FCP service running on vServer
$DESTINATION_VSERVER. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to disable FCP on vServer $DESTINATION_VSERVER"
        } else {
            logMessage -message "FCP service is disabled on vServer
$DESTINATION_VSERVER" -type "SUCCESS"
        }

        # checking if all data lifs are disabled on vServer
        logMessage -message "Finding all data lifs on vServer :
$DESTINATION_VSERVER"
        $dataLifs = Get-NcNetInterface -Vserver $DESTINATION_VSERVER |
Where-Object { $_.Role -contains "data_core" }
        $dataLifs | Select-Object -Property InterfaceName, OpStatus,
DataProtocols, Vserver, Address
```

```
        logMessage -message "Checking if all data lifs are disabled for
vServer : $DESTINATION_VSERVER"
        # Disable the filtered data LIFs
        foreach ($lif in $dataLifs) {
            $checkLif = Get-NcNetInterface -Vserver $DESTINATION_VSERVER
-Name $lif.InterfaceName | Where-Object { $_.OpStatus -eq "down" }
            if($checkLif) {
                logMessage -message "Data lif $($lif.InterfaceName)
disabled for vServer $DESTINATION_VSERVER" -type "SUCCESS"
            } else {
                handleError -errorMessage "Data lif $($lif.InterfaceName)
is enabled. Recommendation: Run the script with SCRIPT_MODE `"configure`"
to disable Data lifs for vServer $DESTINATION_VSERVER"
            }
        }
        logMessage -message "All data lifs are disabled for vServer :
$DESTINATION_VSERVER" -type "SUCCESS"

        # check if multi-admin verification is enabled
        logMessage -message "Checking if multi-admin verification is
enabled"
        $maaConfig = Invoke-NcSsh -Name $DESTINATION_ONTAP_CLUSTER_MGMT_IP
-Credential $DESTINATION_ONTAP_CREDS -Command "set -privilege advanced;
security multi-admin-verify show"
        if ($maaConfig.Value -match "Enabled" -and $maaConfig.Value -match
"true") {
            $maaConfig
            logMessage -message "Multi-admin verification is configured
and enabled" -type "SUCCESS"
        } else {
            handleError -errorMessage "Multi-admin verification is not
configured or not enabled. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to enable and configure Multi-admin verification"
        }

        # check if telnet is disabled
        logMessage -message "Checking if telnet is disabled"
        $telnetConfig = Invoke-NcSsh -Name
$DESTINATION_ONTAP_CLUSTER_MGMT_IP -Credential $DESTINATION_ONTAP_CREDS
-Command "set -privilege advanced; security protocol show -application
telnet"
        if ($telnetConfig.Value -match "enabled" -and $telnetConfig.Value
-match "false") {
            logMessage -message "Telnet is disabled" -type "SUCCESS"
        } else {
            handleError -errorMessage "Telnet is enabled. Recommendation:
```

```
Run the script with SCRIPT_MODE `"configure`" to disable telnet"
        }

        # check if network https is restricted to allowed IP addresses
        logMessage -message "Checking if HTTPS is restricted to allowed IP
addresses $ALLOWED_IPS"
        $networkServicePolicy = Invoke-NcSsh -Name
$DESTINATION_ONTAP_CLUSTER_MGMT_IP -Credential $DESTINATION_ONTAP_CREDS
-Command "set -privilege advanced; network interface service-policy show"
        if ($networkServicePolicy.Value -match "management-https:
$($ALLOWED_IPS)") {
            logMessage -message "HTTPS is restricted to allowed IP
addresses $ALLOWED_IPS" -type "SUCCESS"
        } else {
            handleError -errorMessage "HTTPS is not restricted to allowed
IP addresses $ALLOWED_IPS. Recommendation: Run the script with SCRIPT_MODE
`"configure`" to restrict allowed IP addresses for HTTPS management"
        }
    }
    catch {
        handleError -errorMessage $_.Exception.Message
    }
}
```

此屏幕截图显示保险库控制器上没有连接。

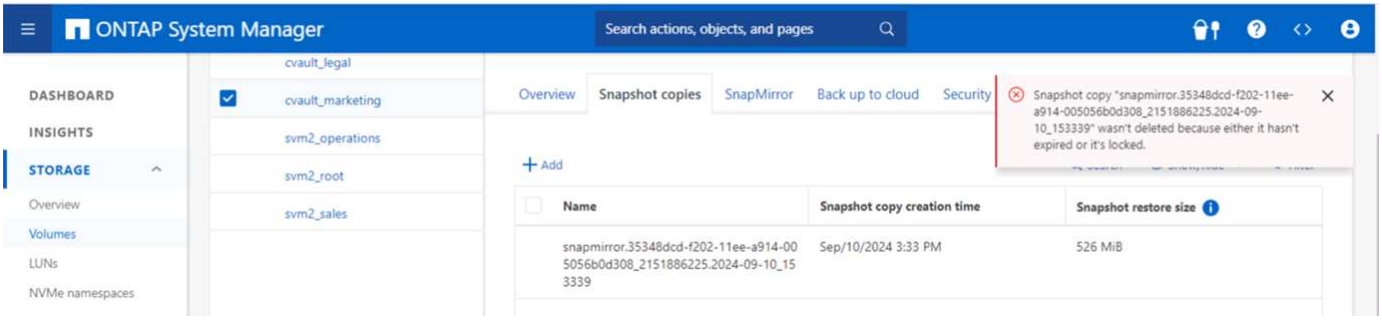
```
cluster2::> network connections  listening show
This table is currently empty.

cluster2::> network connections active show-services
This table is currently empty.

cluster2::> network connections active show-protocols
This table is currently empty.

cluster2::>
```

该屏幕截图显示快照无法被篡改。



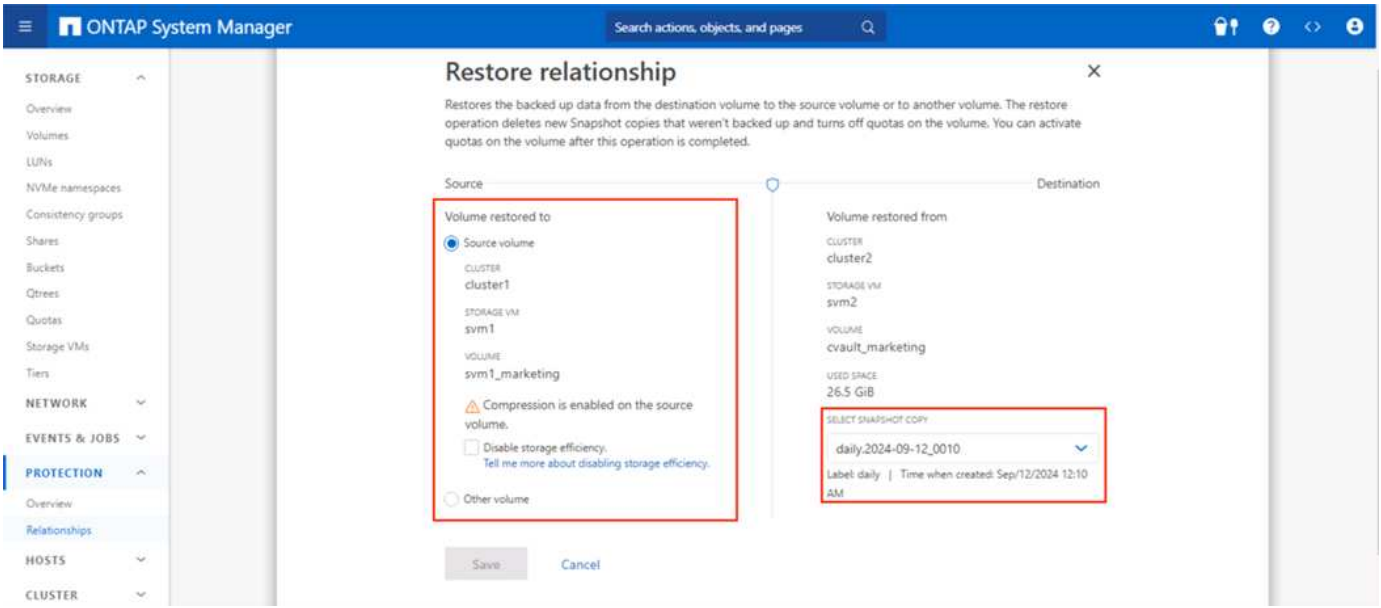
要验证并确认气隙功能，请按照以下步骤操作：

- 测试网络隔离功能以及在未传输数据时停止连接的能力。
- 验证管理接口不能从除允许的 IP 地址之外的任何实体访问。
- 验证多管理员验证是否到位以提供额外的批准层。
- 验证通过 CLI 和 REST API 访问的能力
- 从源头触发向保险库的传输操作，并确保保险库副本无法被修改。
- 尝试删除传输到保管库的不可变快照副本。
- 尝试通过篡改系统时钟来修改保留期。

ONTAP网络保险库数据恢复

如果生产数据中心的数据被破坏，则可以将网络保险库中的数据安全地恢复到所选环境。与物理隔离解决方案不同，隔离ONTAP网络保险库是使用SnapLock Compliance和SnapMirror等原生ONTAP功能构建的。结果是恢复过程既快速又易于执行。

如果发生勒索软件攻击并需要从网络保险库中恢复，则恢复过程简单易行，因为可以使用网络保险库中存储的快照副本来恢复加密数据。



如果要求提供一种更快的方法，在必要时使数据重新上线，以便快速验证、隔离和分析数据以进行恢复。通过使用FlexClone并将 snaplock-type 选项设置为非 snaplock 类型，可以轻松实现这一点。

从ONTAP 9.13.1 开始，可以通过创建FlexClone并将 snaplock-type 选项设置为“non-snaplock”来立即恢复SnapLock保险库关系的目标SnapLock卷上的锁定 Snapshot 副本。执行卷克隆创建操作时，指定Snapshot副本作为“父快照”。有关创建具有SnapLock类型的FlexClone卷的更多信息"这里。"

实践网络保险库的恢复程序将确保建立连接到网络保险库和检索数据的正确步骤。规划和测试程序对于网络攻击事件期间的任何恢复都至关重要。

其他注意事项

在设计和部署基于ONTAP 的网络保险库时，还需要考虑其他因素。

容量大小考虑因素

ONTAP Cyber Vault 目标卷所需的磁盘空间量取决于多种因素，其中最重要的是源卷中数据的变化率。目标卷上的备份计划和快照计划都会影响目标卷上的磁盘使用情况，并且源卷上的变化率可能不会保持恒定。最好提供额外的存储容量缓冲区，以满足未来最终用户或应用程序行为变化的需求。

在ONTAP中确定 1 个月保留关系的大小需要根据多种因素计算存储需求，包括主数据集的大小、数据变化率（每日变化率）以及重复数据删除和压缩节省（如果适用）。

以下是分步方法：

第一步是了解使用网络保险库保护的源卷的大小。这是最初复制到网络保险库目的地的基本数据量。接下来，估计数据集的每日变化率。这是每天变化的数据百分比。充分了解数据的动态性至关重要。

例如：

- 主数据集大小 = 5TB
- 每日变化率=5%（0.05）
- 重复数据删除和压缩效率 = 50% (0.50)

现在，让我们来看一下计算过程：

- 计算每日数据变化率：

```
Changed data per day = 5000 * 5% = 250GB
```

- 计算30天的总变化数据：

```
Total changed data in 30 days = 250 GB * 30 = 7.5TB
```

- 计算所需的总存储量：

```
TOTAL = 5TB + 7.5TB = 12.5TB
```

- 应用重复数据删除和压缩节省：

```
EFFECTIVE = 12.5TB * 50% = 6.25TB
```

存储需求摘要

- 如果没有效率：存储 30 天的网络保险库数据将需要 **12.5TB**。
- 效率为 50%：重复数据删除和压缩后需要 **6.25TB** 的存储空间。

快照副本可能会因元数据而产生额外的开销，但这通常很小。

如果每天进行多次备份，则根据每天拍摄的 Snapshot 副本数量调整计算。

考虑数据随时间的增长，以确保规模适合未来。

对主要/源的性能影响

由于数据传输是一种拉取操作，因此对主存储性能的影响可能因工作负载、数据量和备份频率而异。然而，由于数据传输旨在将数据保护和备份任务卸载到网络保险库存储系统，因此对主系统的整体性能影响通常是适度且可控的。在初始关系设置和第一次完整备份期间，大量数据从主系统传输到网络保险库系统（ SnapLock Compliance卷）。这会导致主系统的网络流量和 I/O 负载增加。初始完整备份完成后， ONTAP只需要跟踪和传输自上次备份以来发生变化的块。与初始复制相比，这会导致 I/O 负载小得多。增量更新效率高，对主存储性能的影响最小。保险库进程在后台运行，从而减少了干扰主系统生产工作负载的可能性。

- 确保存储系统有足够的资源（CPU、内存和 IOP）来处理额外的负载，从而减轻性能影响。

配置、分析、**cron**脚本

NetApp创建了一个"可以下载的单个脚本"并用于配置、验证和安排网络保险库关系。

此脚本的作用

- 集群对等
- SVM 对等
- DP 卷创建
- SnapMirror关系和初始化
- 强化用于网络保险库的ONTAP系统
- 根据传输计划暂停和恢复关系
- 定期验证安全设置并生成显示任何异常的报告

如何使用此脚本

"下载脚本"要使用该脚本，只需按照以下步骤操作：

- 以管理员身份启动 Windows PowerShell。
- 导航到包含脚本的目录。
- 使用执行脚本 `.\`语法以及必需的参数

请确保输入所有信息。第一次运行（配置模式）时，它将要求提供生产系统和新网络保险库系统的凭证。之后，它将在系统之间创建 SVM 对等（如果不存在）、卷和SnapMirror并初始化它们。

Cron 模式可用于安排数据传输的静止和恢复。

操作模式

自动化脚本提供 3 种执行模式 - `configure`， `analyze`和 `cron`。

```
if($SCRIPT_MODE -eq "configure") {
        configure
    } elseif ($SCRIPT_MODE -eq "analyze") {
        analyze
    } elseif ($SCRIPT_MODE -eq "cron") {
        runCron
    }
```

- 配置 - 执行验证检查并将系统配置为隔离。
- 分析——自动监控和报告功能，向监控组发送异常和可疑活动的信息，以确保配置不会偏离。
- Cron - 为了启用断开连接的基础设施，cron 模式会自动禁用 LIF 并停止传输关系。

传输选定卷中的数据需要一些时间，具体取决于系统性能和数据量。

```
./script.ps1 -SOURCE_ONTAP_CLUSTER_MGMT_IP "172.21.166.157"
-SOURCE_ONTAP_CLUSTER_NAME "NTAP915_Src" -SOURCE_VSERVER "svm_NFS"
-SOURCE_VOLUME_NAME "Src_RP_Vol01" -DESTINATION_ONTAP_CLUSTER_MGMT_IP
"172.21.166.159" -DESTINATION_ONTAP_CLUSTER_NAME "NTAP915_Destn"
-DESTINATION_VSERVER "svm_nim_nfs" -DESTINATION_AGGREGATE_NAME
"NTAP915_Destn_01_VM_DISK_1" -DESTINATION_VOLUME_NAME "Dst_RP_Vol01_Vault"
-DESTINATION_VOLUME_SIZE "5g" -SNAPLOCK_MIN_RETENTION "15minutes"
-SNAPLOCK_MAX_RETENTION "30minutes" -SNAPMIRROR_PROTECTION_POLICY
"XDPDefault" -SNAPMIRROR_SCHEDULE "5min" -DESTINATION_CLUSTER_USERNAME
"admin" -DESTINATION_CLUSTER_PASSWORD "PASSWORD123"
```

ONTAP cyber Vault PowerShell 解决方案结论

通过利用ONTAP提供的气隙和强大的强化方法， NetApp使您能够创建一个安全、隔离的存储环境，以抵御不断演变的网络威胁。所有这些都是在保持现有存储基础设施的灵活性和效率的同时实现的。这种安全访问使公司能够以对现有人员、流程和技术框架进行最小限度的改变来实现其严格的安全和正常运行时间目标。

ONTAP Cyber Vault 使用ONTAP中的原生功能，是一种简单的方法，可提供额外的保护，以创建数据的不可变和不可磨灭的副本。将 NetApp 基于ONTAP的网络保险库添加到整体安全态势中将：

- 创建一个与生产和备份网络分离且断开的环境，并限制用户对它的访问。