



# **Amazon FSx for NetApp ONTAP管理**

## **NetApp Automation**

NetApp  
November 18, 2025

This PDF was generated from <https://docs.netapp.com/zh-cn/netapp-automation/solutions/fsxn-burst-to-cloud.html> on November 18, 2025. Always check docs.netapp.com for the latest.

# 目录

Amazon FSx for NetApp ONTAP管理 .....	1
Amazon FSx for NetApp ONTAP管理 - 爆发至云端 .....	1
第1步：安装和配置Docker .....	1
第2步：安装Docker配置 .....	2
第3步：准备Docker映像 .....	3
第4步：为AWS凭据创建环境文件 .....	4
第5步：创建外部卷 .....	4
步骤 6：Amazon FSx for NetApp ONTAP管理和FlexCache配置 Amazon FSx .....	5
步骤 7：销毁Amazon FSx for NetApp ONTAP管理和FlexCache .....	5
Amazon FSx for NetApp ONTAP管理 - 灾难恢复 .....	6
第1步：安装和配置Docker .....	6
第2步：安装Docker配置 .....	7
第3步：准备Docker映像 .....	8
第4步：为AWS凭据创建环境文件 .....	9
第5步：创建外部卷 .....	9
第6步：部署备份解决方案 .....	10

# Amazon FSx for NetApp ONTAP管理

## Amazon FSx for NetApp ONTAP管理 - 爆发至云端

您可以使用此自动化解决方案Amazon FSx for NetApp ONTAP以及卷和相关的FlexCache。



Amazon FSx for NetApp ONTAP管理也称为 **FSx for ONTAP**。

关于该解决方案

概括地说、此解决方案提供的自动化代码将执行以下操作：

- 为ONTAP文件系统配置目标FSx
- 为文件系统配置Storage Virtual Machine (SVM)
- 在源系统和目标系统之间创建集群对等关系
- 在FlexCache的源系统和目标系统之间创建SVM对等关系
- (可选)使用FSx for ONTAP创建FlexVol卷
- 在FSx for ONTAP中创建一个FlexCache卷、其中源卷指向内置存储

此自动化操作基于Docker和Docker编制、必须按如下所述将其安装在Linux虚拟机上。

开始之前

要完成配置和配置、您必须满足以下条件：

- 你需要下载 "[Amazon FSx for NetApp ONTAP管理 - 爆发至云端](#)" 通过NetApp Console Web UI 实现自动化解决方案。该解决方案打包为文件 `AWS_FSxN_BTC.zip`。
- 源系统和目标系统之间的网络连接。
- 具有以下特征的Linux VM：
  - 基于Debian的Linux分发版
  - 部署在用于FSx for ONTAP配置的同VPC子集上
- AWS帐户。

### 第1步：安装和配置Docker

在基于Debian的Linux虚拟机中安装和配置Docker。

步骤

1. 准备环境。

```
sudo apt-get update
sudo apt-get install apt-transport-https ca-certificates curl gnupg-
agent software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key
add -
sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
sudo apt-get update
```

## 2. 安装Docker并验证安装。

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
docker --version
```

## 3. 添加具有关联用户的所需Linux组。

首先检查Linux系统中是否存在组\*Docker\*。如果没有、请创建组并添加用户。默认情况下、当前shell用户将添加到组中。

```
sudo groupadd docker
sudo usermod -aG docker $(whoami)
```

## 4. 激活新的组和用户定义

如果您使用用户创建了新组、则需要激活这些定义。要执行此操作、您可以注销Linux、然后重新进入。或者、您可以运行以下命令。

```
newgrp docker
```

# 第2步：安装Docker配置

在基于Debian的Linux虚拟机中安装Docker编制。

## 步骤

### 1. 安装Docker配置。

```
sudo curl -L
"https://github.com/docker/compose/releases/latest/download/docker-
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

2. 验证安装是否成功。

```
docker-compose --version
```

### 第3步：准备Docker映像

您需要提取并加载随自动化解决方案提供的Docker映像。

#### 步骤

1. 将解决方案文件复制 `AWS\_FSxN\_BTC.zip` 到要运行自动化代码的虚拟机。

```
scp -i ~/<private-key.pem> -r AWS_FSxN_BTC.zip user@<IP_ADDRESS_OF_VM>
```

输入参数 `private-key.pem` 是用于AWS虚拟机身份验证(EC2实例)的专用密钥文件。

2. 导航到包含解决方案文件的正确文件夹、然后解压缩该文件。

```
unzip AWS_FSxN_BTC.zip
```

3. 导航到通过解压缩操作创建的新文件夹 `AWS_FSxN_BTC`、并列出文件。您应看到文件 `aws_fsxn_flexcache_image_latest.tar.gz`。

```
ls -la
```

4. 加载Docker映像文件。加载操作通常应在几秒钟内完成。

```
docker load -i aws_fsxn_flexcache_image_latest.tar.gz
```

5. 确认Docker映像已加载。

```
docker images
```

您应看到标记为的 `latest` Docker映像 `aws_fsxn_flexcache_image`。

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
aws_fsxn_flexcahce_image	latest	ay98y7853769	2 weeks ago	1.19GB

## 第4步：为AWS凭据创建环境文件

您必须使用访问和机密密钥创建一个用于身份验证的本地变量文件。然后将该文件添加到该文件中 .env。

### 步骤

1. 在以下位置创建 `awsauth.env` 文件：

```
path/to/env-file/awsauth.env
```

2. 将以下内容添加到文件中：

```
access_key=<>
secret_key=<>
```

格式“必须”与上面所示完全相同，并且和 value 之间没有任何空格 `key`。

3. 使用变量将绝对文件路径添加到此文件 `AWS\_CREDS` 中 ` .env`。例如：

```
AWS_CREDS=path/to/env-file/awsauth.env
```

## 第5步：创建外部卷

您需要一个外部卷来确保Terraform状态文件和其他重要文件是永久性的。必须为Terraform提供这些文件、才能运行工作流和部署。

### 步骤

1. 在Docker撰写之外创建外部卷。

请确保在运行命令之前将卷名称(Last参数)更新为适当的值。

```
docker volume create aws_fsxn_volume
```

2. 使用命令将外部卷的路径添加到环境文件中 .env：

```
PERSISTENT_VOL=path/to/external/volume:/volume_name
```

请务必保留现有文件内容和冒号格式。例如：

```
PERSISTENT_VOL=aws_fsxn_volume:/aws_fsxn_flexcache
```

而是可以使用以下命令将NFS共享添加为外部卷：

```
PERSISTENT_VOL=nfs/mnt/document:/aws_fsx_flexcache
```

3. 更新Terraform变量。

- a. 导航到文件夹 `aws_fsxn_variables`。
- b. 确认存在以下两个文件： `terraform.tfvars` 和 `variables.tf`。
- c. 根据环境需要更新中的值 `terraform.tfvars`。

有关详细信息、请参见 ["Terraform资源：aws\\_FSX\\_raf\\_File\\_system ONTAP"](#)。

## 步骤 6：Amazon FSx for NetApp ONTAP管理和FlexCache配置 Amazon FSx

您可以为 NetApp ONTAP 管理和FlexCache配置Amazon FSx for NetApp ONTAP。

### 步骤

1. 导航到文件夹根目录(`aws_fs_bTC`)、然后发出配置命令。

```
docker-compose -f docker-compose-provision.yml up
```

此命令将创建两个容器。第一个容器部署FSx for ONTAP、第二个容器创建集群对等、SVM对等、目标卷和FlexCache。

2. 监控配置过程。

```
docker-compose -f docker-compose-provision.yml logs -f
```

此命令可实时提供输出，但已配置为通过文件捕获日志 `deployment.log`。您可以通过编辑这些日志文件并更新变量来更改这些文件的 `DEPLOYMENT_LOGS` 名称 `.env`。

## 步骤 7：销毁Amazon FSx for NetApp ONTAP管理和FlexCache

您可以选择删除和移除Amazon FSx for NetApp ONTAP管理和FlexCache。

1. 将文件中的 `terraform.tfvars` 变量设置 `flexcache_operation` 为 "Destroy"。
2. 导航到文件夹根目录(`aws_fs_bTC`)、然后发出以下命令。

```
docker-compose -f docker-compose-destroy.yml up
```

此命令将创建两个容器。第一个容器删除FlexCache、第二个容器删除FSx for ONTAP。

3. 监控配置过程。

```
docker-compose -f docker-compose-destroy.yml logs -f
```

# Amazon FSx for NetApp ONTAP管理 - 灾难恢复

您可以使用此自动化解决方案通过Amazon FSx for NetApp ONTAP管理对源系统进行灾难恢复备份。



Amazon FSx for NetApp ONTAP管理也称为 **FSx for ONTAP**。

关于该解决方案

概括地说、此解决方案提供的自动化代码将执行以下操作：

- 为ONTAP文件系统配置目标FSx
- 为文件系统配置Storage Virtual Machine (SVM)
- 在源系统和目标系统之间创建集群对等关系
- 在SnapMirror的源系统和目标系统之间创建SVM对等关系
- 创建目标卷
- 在源卷和目标卷之间创建SnapMirror关系
- 在源卷和目标卷之间启动SnapMirror传输

此自动化操作基于Docker和Docker编制、必须按如下所述将其安装在Linux虚拟机上。

开始之前

要完成配置和配置、您必须满足以下条件：

- 您需要下载 "[Amazon FSx for NetApp ONTAP管理 - 灾难恢复](#)" 通过NetApp ConsoleWeb UI 实现自动化解决方案。该解决方案打包如下：FSxN\_DR.zip。此压缩包包含 AWS\_FSxN\_Bck\_Prov.zip 您将使用此文件来部署本文档中描述的解决方案。
- 源系统和目标系统之间的网络连接。
- 具有以下特征的Linux VM：
  - 基于Debian的Linux分发版
  - 部署在用于FSx for ONTAP配置的同VPC子集上
- AWS帐户。

## 第1步：安装和配置Docker

在基于Debian的Linux虚拟机中安装和配置Docker。

步骤

1. 准备环境。



```
sudo apt-get update
sudo apt-get install apt-transport-https ca-certificates curl gnupg-
agent softwareproperties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key
add -
sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
sudo apt-get update
```

## 2. 安装Docker并验证安装。

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
docker --version
```

## 3. 添加具有关联用户的所需Linux组。

首先检查Linux系统中是否存在组\*Docker\*。如果不存在、请创建组并添加用户。默认情况下、当前shell用户将添加到组中。

```
sudo groupadd docker
sudo usermod -aG docker $(whoami)
```

## 4. 激活新的组和用户定义

如果您使用用户创建了新组、则需要激活这些定义。要执行此操作、您可以注销Linux、然后重新进入。或者、您可以运行以下命令。

```
newgrp docker
```

# 第2步：安装Docker配置

在基于Debian的Linux虚拟机中安装Docker编制。

## 步骤

### 1. 安装Docker配置。

```
sudo curl -L
"https://github.com/docker/compose/releases/latest/download/docker-
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

2. 验证安装是否成功。

```
docker-compose --version
```

### 第3步：准备Docker映像

您需要提取并加载随自动化解决方案提供的Docker映像。

#### 步骤

1. 将解决方案文件复制 `AWS\_FSxN\_Bck\_Prov.zip` 到要运行自动化代码的虚拟机。

```
scp -i ~/<private-key.pem> -r AWS_FSxN_Bck_Prov.zip  
user@<IP_ADDRESS_OF_VM>
```

输入参数 `private-key.pem` 是用于AWS虚拟机身份验证(EC2实例)的专用密钥文件。

2. 导航到包含解决方案文件的正确文件夹、然后解压缩该文件。

```
unzip AWS_FSxN_Bck_Prov.zip
```

3. 导航到通过解压缩操作创建的新文件夹 `AWS_FSxN_Bck_Prov`、并列出文件。您应看到文件 `aws_fsxn_bck_image_latest.tar.gz`。

```
ls -la
```

4. 加载Docker映像文件。加载操作通常应在几秒钟内完成。

```
docker load -i aws_fsxn_bck_image_latest.tar.gz
```

5. 确认Docker映像已加载。

```
docker images
```

您应看到标记为的 `latest` Docker映像 `aws_fsxn_bck_image`。

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
aws_fsxn_bck_image	latest	da87d4974306	2 weeks ago	1.19GB

## 第4步：为AWS凭据创建环境文件

您必须使用访问和机密密钥创建一个用于身份验证的本地变量文件。然后将该文件添加到该文件中 `.env`。

### 步骤

1. 在以下位置创建 `awsauth.env` 文件：

```
path/to/env-file/awsauth.env
```

2. 将以下内容添加到文件中：

```
access_key=<>
secret_key=<>
```

格式“必须”与上面所示完全相同，并且和 `value`` 之间没有任何空格 ``key``。

3. 使用变量将绝对文件路径添加到此文件 `AWS_CREDS`` 中 ``.env`。例如：

```
AWS_CREDS=path/to/env-file/awsauth.env
```

## 第5步：创建外部卷

您需要一个外部卷来确保Terraform状态文件和其他重要文件是永久性的。必须为Terraform提供这些文件、才能运行工作流和部署。

### 步骤

1. 在Docker撰写之外创建外部卷。

请确保在运行命令之前将卷名称(Last参数)更新为适当的值。

```
docker volume create aws_fsxn_volume
```

2. 使用命令将外部卷的路径添加到环境文件中 `.env`：

```
PERSISTENT_VOL=path/to/external/volume:/volume_name
```

请务必保留现有文件内容和冒号格式。例如：

```
PERSISTENT_VOL=aws_fsxn_volume:/aws_fsxn_bck
```

而是可以使用以下命令将NFS共享添加为外部卷：

```
PERSISTENT_VOL=nfs/mnt/document:/aws_fsx_bck
```

3. 更新Terraform变量。

- a. 导航到文件夹 `aws_fsxn_variables`。
- b. 确认存在以下两个文件： `terraform.tfvars` 和 `variables.tf`。
- c. 根据环境需要更新中的值 `terraform.tfvars`。

有关详细信息、请参见 ["Terraform资源：aws\\_FSX\\_raf\\_File\\_system ONTAP"](#)。

## 第6步：部署备份解决方案

您可以部署和配置灾难恢复备份解决方案。

### 步骤

1. 导航到文件夹根(`aws_fsxN_Bck_Prov`)、然后发出配置命令。

```
docker-compose up -d
```

此命令可创建三个容器。第一个容器部署FSx for ONTAP。第二个容器将创建集群对等、SVM对等和目标卷。第三个容器将创建SnapMirror关系并启动SnapMirror传输。

2. 监控配置过程。

```
docker-compose logs -f
```

此命令可实时提供输出，但已配置为通过文件捕获日志 `deployment.log`。您可以通过编辑这些日志文件并更新变量来更改这些文件的 `DEPLOYMENT_LOGS` 名称 `.env`。

## 版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。