Red Hat Virtualization 4.4

障害復旧ガイド

障害復旧を目的とした Red Hat Virtualization 4.4 の設定

Last Updated: 2023-08-22

Red Hat Virtualization 4.4 障害復旧ガイド

障害復旧を目的とした Red Hat Virtualization 4.4 の設定

Red Hat Virtualization Documentation Team
Red Hat Customer Content Services
rhev-docs@redhat.com

法律上の通知

概要

Red Hat Virtualization は、障害が発生した場合でも環境が機能し続けるように設定できます。このドキュメントでは、障害復旧を目的とした Red Hat Virtualization の環境設定に関する情報と手順について説明します。

目次

第1章 障害復旧ソリューション

Red Hat Virtualization では、サイト停止時に環境を確実に復旧できるように、2 種類の障害復旧ソリューションがサポートされています。どちらのソリューションも 2 つのサイトに対応しており、どちらにもレプリケートされたストレージが必要です。

Active-Active 障害復旧

このソリューションは、ストレッチクラスター設定を使用して実装されます。これは、プライマリーサイトとセカンダリーサイトで必要な仮想マシンを実行できるホストを含むクラスターが存在する単一の RHV 環境があることを意味します。停止すると、仮想マシンはセカンダリーサイトのホストに自動的に移行します。ただし、この環境はレイテンシーとネットワークの要件を満たす必要があります。詳細は、Active-Active の概要 を参照してください。

Active-Passive 障害復旧

サイト間フェイルオーバーとも呼ばれるこの障害復旧ソリューションは、アクティブプライマリー環境とパッシブセカンダリー (バックアップ) 環境の 2 つの別個の RHV 環境を設定することによって実装されます。サイト間のフェイルオーバーおよびフェイルバックは手動で実行する必要があり、Ansible で管理されます。詳細は、Active-Passive の概要 を参照してください。

第2章 ACTIVE-ACTIVE 障害復旧

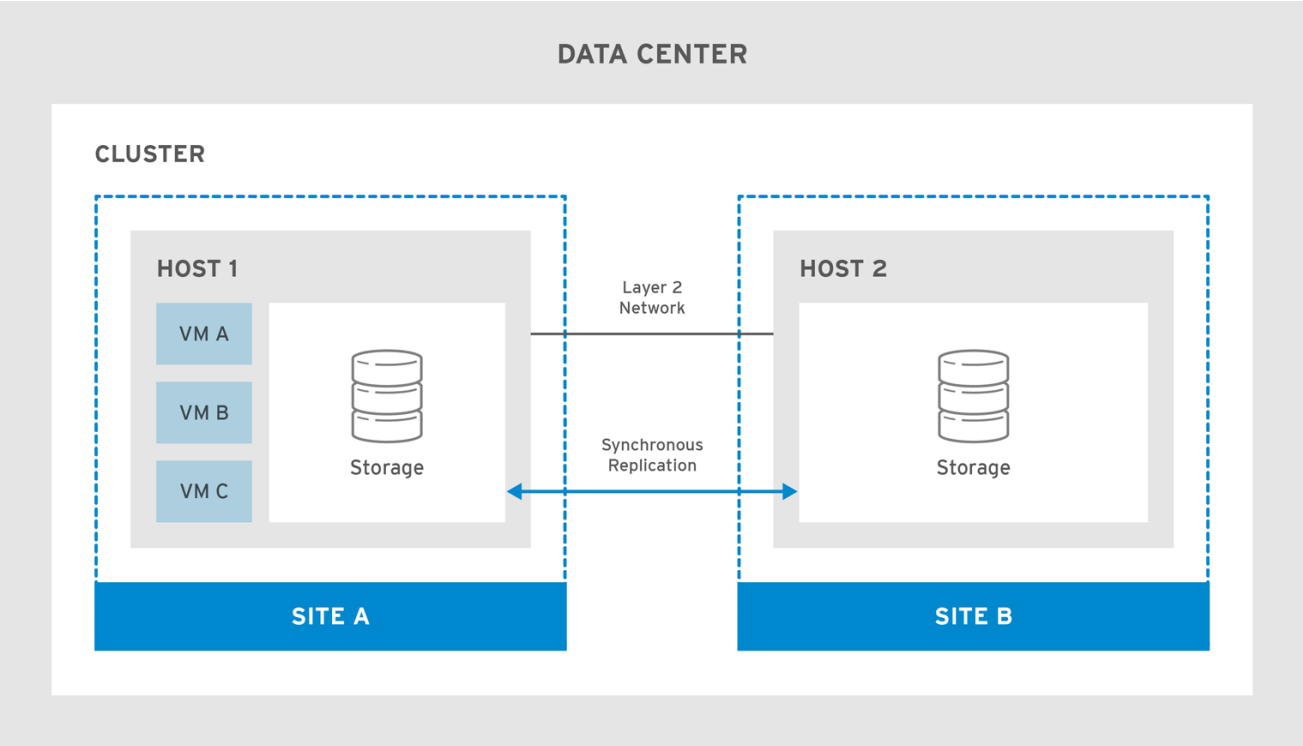
2.1. ACTIVE-ACTIVE の概要

Active-Active 障害復旧フェイルオーバーの設定は、2 つのサイトにまたがる可能性があります。両方のサイトがアクティブであり、プライマリーサイトが使用できなくなった場合でも、Red Hat Virtualization 環境はセカンダリーサイトで引き続き動作し、ビジネス継続性を確保します。

Active-Active フェイルオーバーの設定には、仮想マシンを実行できるストレッチクラスターが含まれています。このクラスターは、プライマリーサイトとセカンダリーサイトの両方にあります。すべてのホストは、同じ Red Hat Virtualization クラスターに属します。

この設定には、両方のサイトでレプリケートされた書き込み可能ストレージが必要です。これにより、仮想マシンは 2 つのサイト間で移行でき、両サイトのストレージで引き続き実行されます。

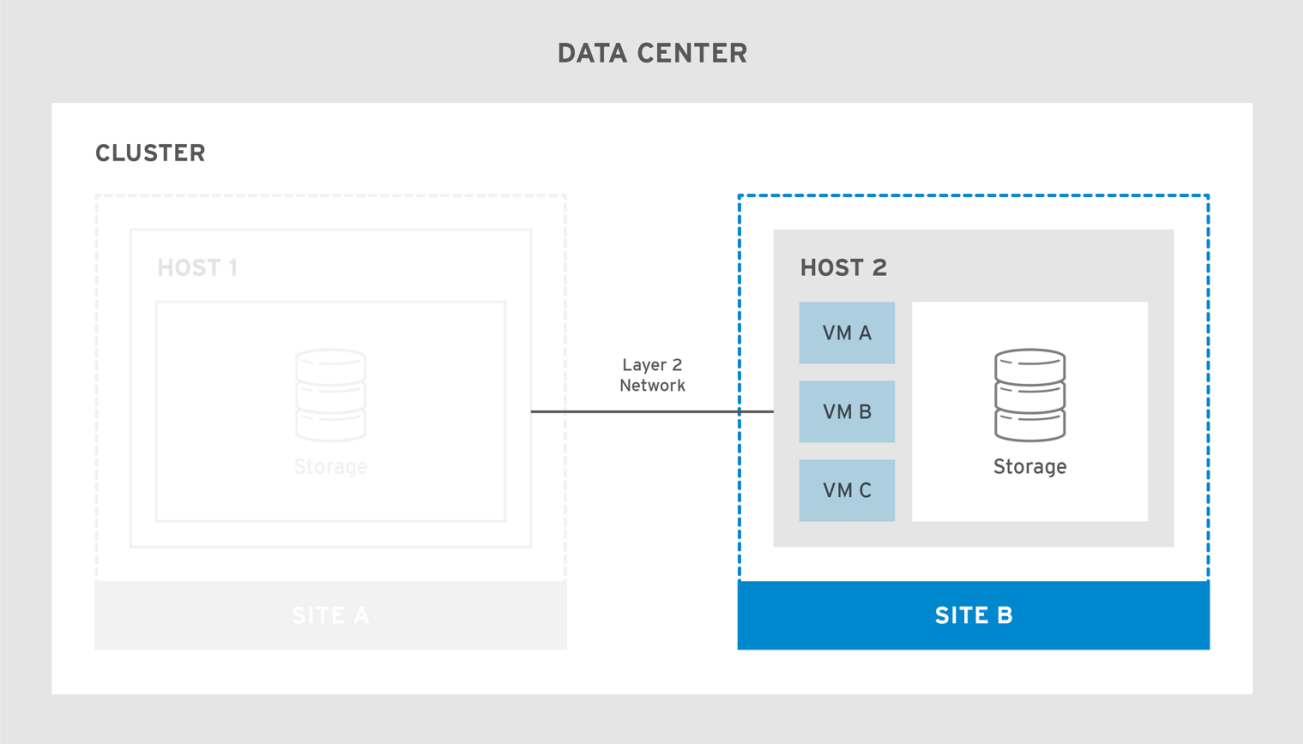
図2.1 ストレッチクラスター設定



RHV_460251_1017

プライマリーサイトが使用できなくなると、仮想マシンはセカンダリーサイトに移行します。サイトが使用可能になり、ストレージが両方のサイトでレプリケートされると、仮想マシンはプライマリーサイトに自動的にフェイルバックされます。

図2.2 ストレッチクラスターのフェイルオーバー



重要

仮想マシンのフェイルオーバーとフェイルバックが機能することを確認するには、以下を実行します。

- 仮想マシンは高可用性があるように設定する必要があります。また、各仮想マシンは、電源管理がなくても起動できるように、ターゲットストレージドメインにリースを取得している必要があります。
- ホストアフィニティーにソフト強制された仮想マシンを設定して、選択したホストでのみ仮想マシンが起動するようにする必要があります。

詳細は、**仮想マシン管理ガイド** の 仮想マシンの高可用性によるアップタイムの向上 および アフィニティーグループ を参照してください。

ストレッチクラスター設定は、セルフホストエンジン環境またはスタンドアロン Manager 環境を使用して実装できます。各種デプロイメントの詳細については、**製品ガイド** の Red Hat Virtualization のアーキテクチャー を参照してください。

2.2. ネットワークの考慮事項

クラスター内のすべてのホストは、L2 ネットワーク上の同じブロードキャストドメイン上にある必要があります。したがって、2 つのサイト間の接続は L2 である必要があります。

L2 ネットワーク全体におけるサイト間の最大レイテンシー要件は、2 つのセットアップで異なります。スタンドアロンの Manager 環境には最大 100 ミリ秒のレイテンシーが必要ですが、セルフホストエンジン環境には最大 7 ミリ秒のレイテンシーが必要です。

2.3. ストレージに関する考慮事項

Red Hat Virtualization のストレージドメインは、ブロックデバイス (SAN - iSCSI または FCP) もしくはファイルシステム (NAS - NFS、GlusterFS、または他の POSIX 準拠ファイルシステム) のいずれかで構成されます。Red Hat Virtualization ストレージの詳細は、**管理ガイド** の ストレージ を参照してください。

注記

GlusterFS Storage は非推奨になり、将来のリリースではサポートされなくなります。

サイトには、共有レイヤー 2 (L2) ネットワーク接続を備え、同時期にレプリケートされた、両方のサイトに書き込み可能なストレージが必要です。レプリケートされたストレージは、仮想マシンがサイト間で移行し、サイトのストレージで継続的に実行できるようにするために必要です。Red Hat Enterprise Linux 7 以降でサポートされているすべてのストレージレプリケーションオプションは、ストレッチクラスターで使用できます。

重要

ストレージベンダーが推奨するカスタムマルチパス設定がある場合は、SAN ベンダーのマルチパス設定のカスタマイズ の手順と重要な制限を参照してください。

プライマリーサイトのホストに SPM ロールを設定し、優先されるようにします。そのためには、プライマリーサイトホストで SPM の優先度を高く設定し、セカンダリーサイトホストでは SPM の優先度を低く設定します。停電など、プライマリーサイト内のネットワークデバイスに影響を与えるプライマリーサイトの障害が発生して SPM ホストのフェンシングデバイスに到達できない場合、セカンダリーサイトのホストは SPM のロールを引き継ぐことができません。

このようなシナリオでは、仮想マシンはフェイルオーバーを実行しますが、新しいディスクの追加、既存ディスクの拡張、仮想マシンのエクスポートなど、SPM のロールを必要とする操作は実行できません。

完全な機能を復元するには、災害の実際の性質を検出し、根本原因を修正して SPM ホストを再起動した後、SPM ホストで **Confirm 'Host has been Rebooted'** を選択します。

関連情報

管理ガイド の 反応しないホストを手動でフェンシングまたは隔離する方法。

2.4. セルフホストエンジンストレッチクラスター環境の設定

この手順では、セルフホストエンジンのデプロイメントを使用してストレッチクラスターを設定する方法を説明します。

前提条件

- L2 ネットワーク接続を持つ両方のサイトに書き込み可能なストレージサーバー。
- ストレージをレプリケートするリアルタイムストレージレプリケーションサービス。

制限

- サイト間のレイテンシーは最大 7 ミリ秒。

セルフホストエンジン用ストレッチクラスターの設定

1. セルフホストエンジンをデプロイします。コマンドラインを使用して Red Hat Virtualization をセルフホストエンジンとしてインストール を参照してください。

2. 各サイトに追加のセルフホストエンジンノードをインストールし、それらをクラスターに追加します。**コマンドラインを使用して Red Hat Virtualization をセルフホストエンジンとしてインストール** の Red Hat Virtualization Manager へのセルフホストエンジンノードの追加 を参照してください。

3. オプションで、追加の標準ホストをインストールします。**コマンドラインを使用して Red Hat Virtualization をセルフホストエンジンとしてインストール** の Red Hat Virtualization Manager への標準ホストの追加 を参照してください。

1. プライマリーサイトのすべてのホストで SPM の優先度を高く設定し、プライマリーサイトのすべてのホストが使用できない場合にのみセカンダリーサイトへの SPM フェイルオーバーが発生するように設定します。**管理ガイド** の SPM の優先度 を参照してください。

2. フェイルオーバーが必要なすべての仮想マシンを高可用性として設定し、仮想マシンがターゲットストレージドメインにリースを持っていることを確認します。**仮想マシン管理ガイド** の 高可用性仮想マシンの設定 を参照してください。

3. ソフトアフィニティーをホストするように仮想マシンを設定し、アフィニティーグループに期待する動作を定義します。**仮想マシン管理ガイド** の アフィニティーグループ、および **管理ガイド** の スケジューリングポリシー を参照してください。

Active-Active フェイルオーバーは、メインサイトのホストをメンテナンスモードに切り替えることで手動で実行できます。

2.5. スタンドアロン MANAGER ストレッチクラスター環境の設定

この手順では、スタンドアロン Manager デプロイメントを使用してストレッチクラスターを設定する方法を説明します。

前提条件

- L2 ネットワーク接続を持つ両方のサイトに書き込み可能なストレージサーバー。
- ストレージをレプリケートするリアルタイムストレージレプリケーションサービス。

制限

- サイト間のレイテンシーは最大 100 ミリ秒。

重要

Manager は、仮想マシンがサイト間でフェイルオーバーおよびフェイルバックできるように高可用性を備えている必要があります。Manager がサイトでダウンした場合、仮想マシンはフェイルオーバーしません。

スタンドアロン Manager は、外部で管理されている場合にのみ高可用性があります。以下に例を示します。

- Red Hat の High Availability Add-On を使用している。
- 別の仮想化環境での高可用性仮想マシンとして管理されている。
- Red Hat Enterprise Linux Cluster Suite を使用している。
- パブリッククラウド内で管理されている。

手順

1. Red Hat Virtualization Manager をインストールして設定します。**Red Hat Virtualization をローカルデータベースが設定されたスタンドアロン Manager としてインストール** を参照してください。
2. 各サイトにホストをインストールし、それらのホストをクラスターに追加します。**Red Hat Virtualization をローカルデータベースが設定されたスタンドアロン Manager としてインストール** の Red Hat Virtualization のホストのインストール を参照してください。

1. プライマリーサイトのすべてのホストで SPM の優先度を高く設定し、プライマリーサイトのすべてのホストが使用できない場合にのみセカンダリーサイトへの SPM フェイルオーバーが発生するように設定します。**管理ガイド** の SPM の優先度 を参照してください。
2. フェイルオーバーが必要なすべての仮想マシンを高可用性として設定し、仮想マシンがターゲットストレージドメインにリースを持っていることを確認します。**仮想マシン管理ガイド** の 高可用性仮想マシンの設定 を参照してください。
3. ソフトアフィニティーをホストするように仮想マシンを設定し、アフィニティーグループに期待する動作を定義します。**仮想マシン管理ガイド** の アフィニティーグループ、および **管理ガイド** の スケジューリングポリシー を参照してください。

Active-Active フェイルオーバーは、メインサイトのホストをメンテナンスモードに切り替えることで手動で実行できます。

第3章 ACTIVE-PASSIVE 障害復旧

3.1. ACTIVE-PASSIVE の概要

Red Hat Virtualization は、2 つのサイトにまたがる Active-Passive 障害復旧ソリューションをサポートしています。プライマリーサイトが使用できなくなった場合、Red Hat Virtualization 環境を強制的にセカンダリー (バックアップ) サイトにフェイルオーバーできます。

フェイルオーバーは、セカンダリーサイトで Red Hat Virtualization 環境を設定することで実現されます。これには以下が必要になります。

- アクティブな Red Hat Virtualization Manager。
- データセンターおよびクラスター。
- プライマリーサイトと同じ一般的な接続を持つネットワーク。
- フェイルオーバー後に重要な仮想マシンを実行できるアクティブなホスト。

重要

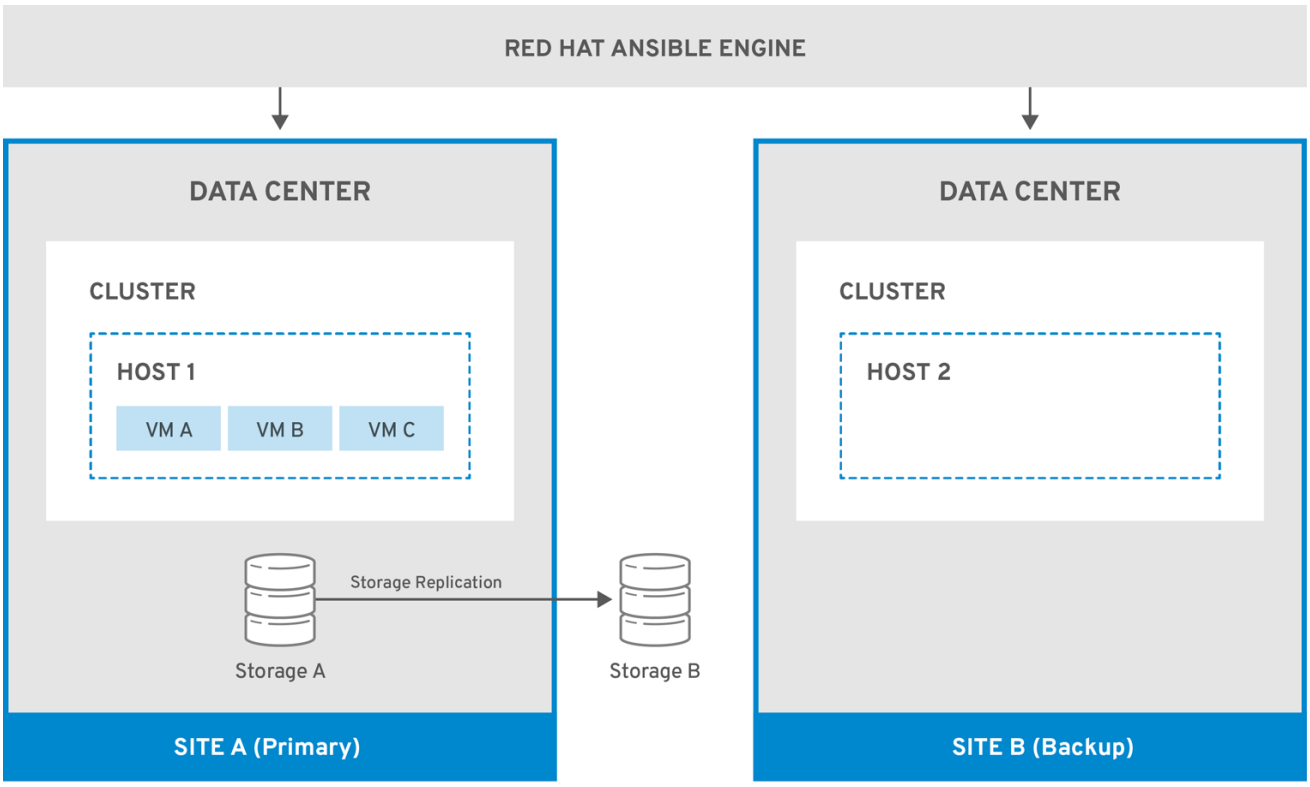
セダンダリー環境に、フェイルオーバーされた仮想マシンを実行するのに十分なリソースがあること、プライマリー環境とセカンダリー環境の両方に同一の Manager バージョン、データセンターとクラスターの互換性レベル、PostgreSQL バージョンがあることを確認する必要があります。サポートされる最小互換性レベルは 4.2 です。

プライマリーサイトに仮想マシンディスクおよびテンプレートを含むストレージドメインをレプリケートする必要があります。これらのレプリケートされたストレージドメインは、セカンダリーサイトにアタッチしないでください。

フェイルオーバーとフェイルバックのプロセスは手動で実行する必要があります。そのためには、Ansible Playbook を作成してサイト間でエンティティーをマッピングし、フェイルオーバーとフェイルバックのプロセスを管理する必要があります。マッピングファイルは、ターゲットサイトのどこでフェイルオーバーまたはフェイルバックするかを Red Hat Virtualization コンポーネントに指示します。

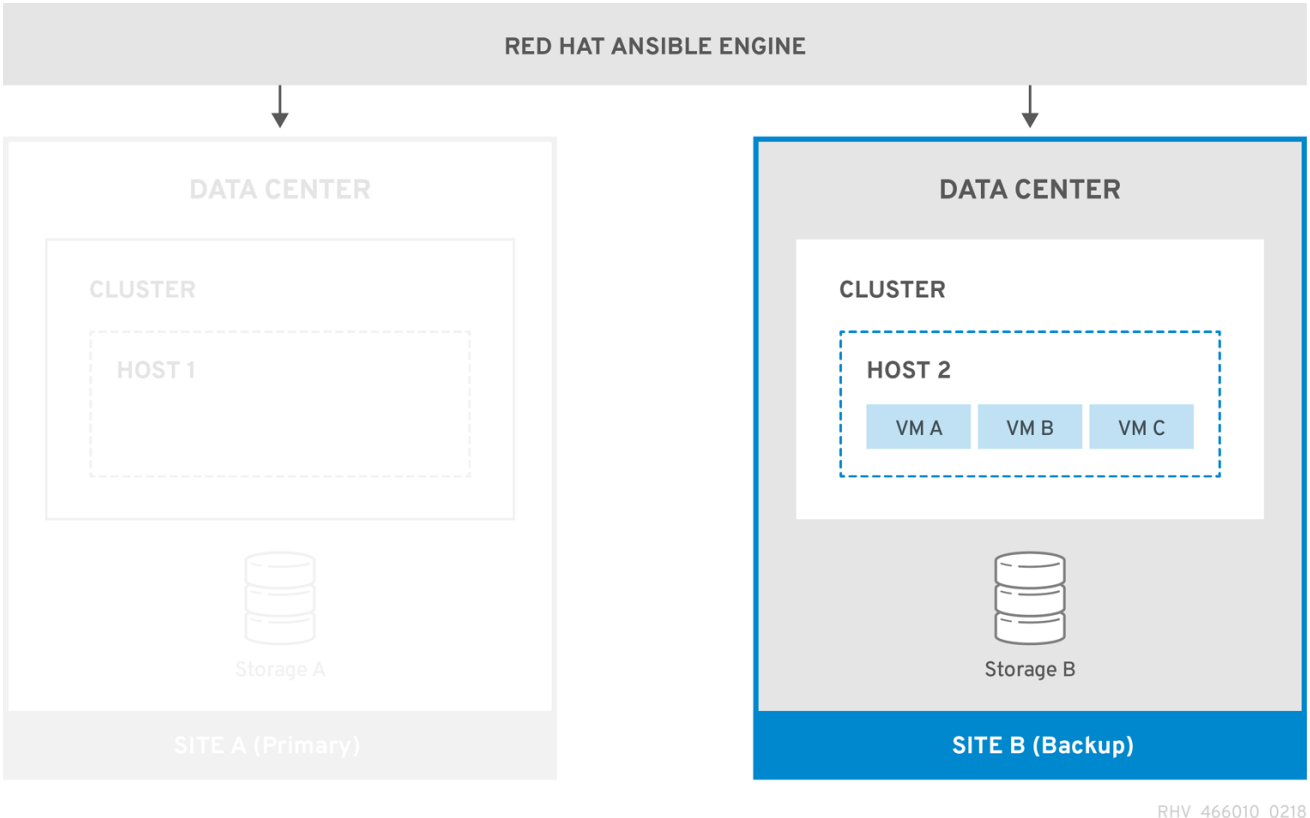
次の図は、Red Hat Ansible Engine を実行しているマシンが高可用性であり、**oVirt.disaster-recovery** Ansible ロール、設定済み Playbook、およびマッピングファイルにアクセスできる、Active-Passive セットアップを示しています。仮想マシンディスクをサイト A に保存するストレージドメインがレプリケートされます。サイト B に、仮想マシンやアタッチされたストレージドメインはありません。

図3.1 Active-Passive 設定



環境がサイト B にフェイルオーバーすると、最初にストレージドメインがアタッチされ、サイト B のデータセンターでアクティブ化されてから、仮想マシンが登録されます。高可用性の仮想マシンは最初にフェイルオーバーします。

図3.2 バックアップサイトへのフェイルオーバー



再実行中の場合は、プライマリーサイト (サイト A) に手動でフェイルバックする必要があります。

3.2. ネットワークの考慮事項

プライマリーサイトとセカンダリーサイトに、同じ一般的な接続があることを確認する必要があります。

複数のネットワークまたは複数のデータセンターがある場合は、マッピングファイルで空のネットワークマッピングを使用して、フェイルオーバー時にすべてのエンティティーがターゲットに登録されるようにする必要があります。詳細については、マッピングファイルの属性 を参照してください。

3.3. ストレージに関する考慮事項

Red Hat Virtualization のストレージドメインは、ブロックデバイス (SAN - iSCSI または FCP) もしくはファイルシステム (NAS - NFS、GlusterFS、または他の POSIX 準拠ファイルシステム) のいずれかで構成されます。Red Hat Virtualization ストレージの詳細は、**管理ガイド** の ストレージ を参照してください。

注記

GlusterFS Storage は非推奨になり、将来のリリースではサポートされなくなります。

重要

ローカルストレージドメインは、障害復旧ではサポートされていません。

プライマリーおよびセカンダリーストレージのレプリカが必要です。プライマリーストレージドメインのブロックデバイス、もしくは仮想マシンディスクまたはテンプレートを含む共有をレプリケートする必要があります。セカンダリーストレージは、データセンターにアタッチしないでください。フェイルオーバー時にバックアップサイトのデータセンターに追加されます。

セルフホストエンジンを使用して障害復旧を実装する場合は、Manager 仮想マシンが使用するストレージドメインに仮想マシンディスクが含まれていないことを確認します。含まれている場合、ストレージドメインはフェイルオーバーされません。

Red Hat Enterprise Linux 7 以降でサポートされているレプリケーションオプションを持つすべてのストレージソリューションを使用できます。

3.4. 必要な ANSIBLE PLAYBOOK の作成

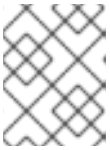
Ansible は、障害復旧フェイルオーバーとフェイルバックの開始および管理に使用されます。したがって、これを容易にするために Ansible Playbook を作成する必要があります。Ansible Playbook の作成に関する詳細は、Ansible ドキュメント を参照してください。

前提条件

- プライマリーサイトで完全に機能する Red Hat Virtualization 環境。
- プライマリー環境と同じデータセンターとクラスターの互換性レベルを持つセカンダリーサイトのバックアップ環境。バックアップ環境には以下が必要です。
  - Red Hat Virtualization Manager。
  - 仮想マシンを実行し、レプリケートされたストレージドメインに接続できるアクティブホスト。

- クラスターのあるデータセンター。
- プライマリーサイトと同じ一般的な接続を持つネットワーク。

- レプリケートされたストレージ。詳細については、ストレージに関する考慮事項 を参照してください。

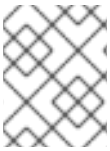


注記

仮想マシンとテンプレートを含むレプリケートされたストレージは、セカンダリーサイトにアタッチしないでください。

- **oVirt.disaster-recovery** パッケージが、フェイルオーバーおよびフェイルバックを自動化する高可用性 Red Hat Ansible Engine マシンにインストールされている必要があります。
- Red Hat Ansible Engine を実行しているマシンは、SSH を使用してプライマリーおよびセカンダリーサイトの Manager に接続できる必要があります。

また、アフィニティーグループ、アフィニティーラベル、ユーザーなど、プライマリーサイトに存在する環境プロパティーをセカンダリーサイトに作成することが推奨されます。



注記

Ansible Playbook のデフォルト動作は、**/usr/share/ansible/roles/oVirt.disaster-recovery/defaults/main.yml** ファイルで設定できます。

次の Playbook を作成する必要があります。

- プライマリーサイトとセカンダリーサイトのエンティティーをマップするファイルを作成する Playbook。
- フェイルオーバー Playbook。
- フェイルバック Playbook。

フェイルバックする前にプライマリーサイトをクリーンアップする Playbook もオプションで作成できます。

フェイルオーバーおよびフェイルバックを管理している Ansible マシンの **/usr/share/ansible/roles/oVirt.disaster-recovery/** に Playbook と関連ファイルを作成します。それを管理できる複数の Ansible マシンがある場合は、それらすべてにファイルをコピーするようにしてください。

Testing the Active-Passive Configuration のテスト手順を 1 つ以上使用して、設定をテストできます。

3.4.1. Ansible タスク用の ovirt-dr スクリプト

ovirt-dr スクリプトは、次の Ansible タスクを簡素化します。

- プライマリーサイトおよびセカンダリーサイトのフェイルオーバーおよびフェイルバック用 **var** マッピングファイルの生成
- **var** マッピングファイルの検証
- ターゲットサイトでのフェイルオーバーの実行

- ターゲットサイトからソースサイトへのフェイルバックの実行

このスクリプトは、**/usr/share/ansible/roles/oVirt.disaster-recovery/files** にあります。

使用方法

```
# ./ovirt-dr generate/validate/failover/failback
            [--conf-file=dr.conf]
            [--log-file=ovirt-dr-log_number.log]
            [--log-level=DEBUG/INFO/WARNING/ERROR]
```

スクリプトの動作のパラメーターは、設定ファイル **/usr/share/ansible/roles/oVirt.disaster-recovery/files/dr.conf** に設定できます。

--conf-file オプションを使用して、設定ファイルの場所を変更できます。

--log-file および **--log-level** オプションを使用して、ログの詳細の場所とレベルを設定できます。

3.4.2. マッピングファイルを生成する Playbook の作成

マッピングファイルの生成に使用する Ansible Playbook は、ターゲット (プライマリー) サイトのエンティティーをファイルに事前入力します。次に、IP アドレス、クラスター、アフィニティーグループ、アフィニティーラベル、外部 LUN ディスク、承認ドメイン、ロール、vNIC プロファイルなどのバックアップサイトのエンティティーを、ファイルに手動で追加する必要があります。

重要

セルフホストエンジンのストレージドメインに仮想マシンディスクがある場合は、マッピングファイルの生成に失敗します。また、フェイルオーバーしてはならないため、マッピングファイルにはこのストレージドメインの属性は含まれません。

この例では、Ansible Playbook は **dr-rhv-setup.yml** という名前で、プライマリーサイトの Manager マシンで実行されます。

手順

1. Ansible Playbook を作成してマッピングファイルを生成します。以下に例を示します。

```
---
- name: Generate mapping
  hosts: localhost
  connection: local

  vars:
    site: https://example.engine.redhat.com/ovirt-engine/api
    username: admin@internal
    password: my_password
    ca: /etc/pki/ovirt-engine/ca.pem
    var_file: disaster_recovery_vars.yml

  roles:
    - oVirt.disaster-recovery
```

注記

セキュリティーを強化するには、**.yml** ファイルで Manager パスワードを暗号化できます。詳細は、**管理ガイド** の Ansible を使用した Red Hat Virtualization の設定 を参照してください。

2. Ansible コマンドを実行してマッピングファイルを生成します。プライマリーサイトの設定は事前に入力されます。

```
# ansible-playbook dr-rhv-setup.yml --tags "generate_mapping"
```

3. バックアップサイトの設定を使用して、マッピングファイル (この場合は **disaster_recovery_vars.yml**) を設定します。マッピングファイルの属性に関する詳細は、Mapping File Attributes を参照してください。

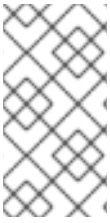
フェイルオーバーおよびフェイルバックを実行できる Ansible マシンが複数ある場合は、マッピングファイルをすべての関連マシンにコピーします。

3.4.3. フェイルオーバーおよびフェイルバック Playbook の作成

作成および設定したマッピングファイル (この場合は **disaster_recovery_vars.yml**) があることを確認してください。これは、Playbook に追加する必要があります。

パスワードファイル (**passwords.yml** など) を定義して、プライマリーおよびセカンダリーサイトの Manager パスワードを保存できます。以下に例を示します。

```
---
# This file is in plain text, if you want to
# encrypt this file, please execute following command:
#
# $ ansible-vault encrypt passwords.yml
#
# It will ask you for a password, which you must then pass to
# ansible interactively when executing the playbook.
#
# $ ansible-playbook myplaybook.yml --ask-vault-pass
#
dr_sites_primary_password: primary_password
dr_sites_secondary_password: secondary_password
```



注記

セキュリティーを強化する場合は、パスワードファイルを暗号化できます。ただし、Playbook の実行時に **--ask-vault-pass** パラメーターを使用する必要があります。詳細については、管理ガイド の **Ansible ロールを使用した Red Hat Virtualization の設定**を参照してください。

これらの例では、フェイルオーバーおよびフェイルバックする Ansible Playbook の名前は、それぞれ **dr-rhv-failover.yml** と **dr-rhv-failback.yml** です。

以下の Ansible Playbook を作成して、環境のフェイルオーバーを行います。

```
---
- name: Failover RHV
```

```
  hosts: localhost
  connection: local
  vars:
    dr_target_host: secondary
    dr_source_map: primary
  vars_files:
    - disaster_recovery_vars.yml
    - passwords.yml
  roles:
    - oVirt.disaster-recovery
```

以下の Ansible Playbook を作成して、環境のフェイルバックを行います。

```
---
- name: Failback RHV
  hosts: localhost
  connection: local
  vars:
    dr_target_host: primary
    dr_source_map: secondary
  vars_files:
    - disaster_recovery_vars.yml
    - passwords.yml
  roles:
    - oVirt.disaster-recovery
```

3.4.4. プライマリーサイトをクリーンアップするための Playbook の作成

プライマリーサイトにフェイルバックする前に、インポートするすべてのストレージドメインがプライマリーサイトから削除されていることを確認する必要があります。これは、Manager で手動で行うことも、必要に応じて Ansible Playbook を作成して実行することもできます。

この例では、プライマリーサイトをクリーンアップする Ansible Playbook の名前は **dr-cleanup.yml** で、別の Ansible Playbook によって生成されたマッピングファイルを使用します。

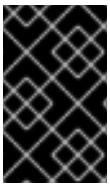
```
---
- name: clean RHV
  hosts: localhost
  connection: local
  vars:
    dr_source_map: primary
  vars_files:
    - disaster_recovery_vars.yml
  roles:
    - oVirt.disaster-recovery
```

3.5. フェイルオーバーの実行

前提条件

- セカンダリーサイトの Manager およびホストが実行中である。
- レプリケートされたストレージドメインが読み取り/書き込みモードである。

- レプリケートされたストレージドメインがセカンダリーサイトにアタッチされていない。
- SSH 経由でプライマリーサイトおよびセカンダリーサイトの Manager に接続できる Red Hat Ansible Engine を実行しているマシンと、必要なパッケージおよびファイル。
  - **oVirt.disaster-recovery** パッケージ。
  - マッピングファイルおよび必要なフェイルオーバー Playbook。



重要

Sanlock は、フェイルオーバープロセスを開始する前に、レプリケートされたストレージドメインからすべてのストレージロックを解除する必要があります。これらのロックは、障害が発生してから約 80 秒後に自動解除される必要があります。

この例では、前に作成した **dr-rhv-failover.yml** Playbook を使用しています。

手順

1. 以下のコマンドでフェイルオーバー Playbook を実行します。

```
# ansible-playbook dr-rhv-failover.yml --tags "fail_over"
```

2. プライマリーサイトがアクティブになったら、フェイルバックする前にに環境をクリーンアップしてください。詳細は、プライマリーサイトのクリーニング を参照してください。

3.6. プライマリーサイトのクリーニング

フェイルオーバーした後は、フェイルバックする前にプライマリーサイトの環境をクリーンアップする必要があります。

- プライマリーサイト内の全ホストを再起動します。
- セカンダリーサイトのストレージドメインが読み取り/書き込みモードで、プライマリーサイトのストレージドメインが読み取り専用モードであることを確認します。
- セカンダリーサイトのストレージドメインからプライマリーサイトのストレージドメインにレプリケーションを同期します。
- プライマリーサイトの、インポートするすべてのストレージドメインを消去します。これは、Manager で手動で行うか、Ansible Playbook を作成して実行できます。手動の手順については **管理ガイド** の ストレージドメインのデタッチ、Ansible Playbook を作成する方法については プライマリーサイトをクリーンアップするための Playbook の作成 を参照してください。

この例では、前に作成した **dr-cleanup.yml** Playbook を使用して環境をクリーンアップします。

手順

1. 次のコマンドを使用してプライマリーサイトをクリーンアップします。

```
# ansible-playbook dr-cleanup.yml --tags "clean_engine"
```

2. 環境をプライマリーサイトにフェイルバックできるようになりました。詳細は フェイルバックの実行 を参照してください。

3.7. フェイルバックの実行

フェイルオーバーした後、プライマリーサイトがアクティブになり、その環境をクリーンアップするために必要な手順を実行すると、プライマリーサイトにフェイルバックできます。

前提条件

- プライマリーサイトの環境が実行され、クリーンアップされている。詳細は プライマリーサイトのクリーニング を参照してください。
- セカンダリーサイトの環境が実行され、アクティブなストレージドメインがある。
- SSH 経由でプライマリーサイトおよびセカンダリーサイトの Manager に接続できる Red Hat Ansible Engine を実行しているマシンと、必要なパッケージおよびファイル。
  - **oVirt.disaster-recovery** パッケージ。
  - マッピングファイルおよび必要なフェイルバック Playbook。

この例では、前に作成した **dr-rhv-failback.yml** Playbook を使用しています。

手順

1. 以下のコマンドでフェイルバック Playbook を実行します。

```
# ansible-playbook dr-rhv-failback.yml --tags "fail_back"
```

2. プライマリーストレージドメインからセカンダリーストレージドメインへのレプリケーションを有効にします。

付録A マッピングファイルの属性

以下の表は、Active-Passive 障害復旧ソリューションの 2 つのサイト間でフェイルオーバーおよびフェイルバックを実行するために使用されるマッピングファイルの属性を示しています。

表A.1 マッピングファイルの属性

マッピングファイルのセクション	説明
サイトの詳細	プライマリーサイトおよびセカンダリーサイトの Manager の詳細をマッピングします。以下はその例です。 dr_sites_primary_url: https://manager1.example.redhat.com/ovirt-engine/api dr_sites_primary_username: admin@internal dr_sites_primary_ca_file: /etc/pki/ovirt-engine/ca.pem # Please fill in the following properties for the secondary site: dr_sites_secondary_url: https://manager2.example.redhat.com/ovirt-engine/api dr_sites_secondary_username: admin@internal dr_sites_secondary_ca_file: /etc/pki/ovirt-engine/ca.pem

ストレージドメインの詳細	プライマリーサイトとセカンダリーサイト間でストレージドメインの詳細をマッピングします。以下はその例です。
	
	dr_import_storages:
	- dr_domain_type: nfs
	  dr_primary_name: DATA
	  dr_master_domain: True
	  dr_wipe_after_delete: False
	  dr_backup: False
	  dr_critical_space_action_blocker: 5
	  dr_warning_low_space: 10
	  dr_primary_dc_name: Default
	  dr_discard_after_delete: False
	  dr_primary_path: /storage/data
	  dr_primary_address: 10.64.100.xxx
	  # Fill in the empty properties related to the secondary site
	  dr_secondary_dc_name: Default
	  dr_secondary_path: /storage/data2
	  dr_secondary_address:10.64.90.xxx
	  dr_secondary_name: DATA

マッピングファイルのセクション	説明
クラスターの詳細	プライマリーサイトとセカンダリーサイトの間でクラスター名をマッピングします。以下はその例です。 dr_cluster_mappings: - primary_name: cluster_prod secondary_name: cluster_recovery - primary_name: fc_cluster secondary_name: recovery_fc_cluster
アフィニティーグループの詳細	仮想マシンが属するアフィニティーグループをマッピングします。以下はその例です。 dr_affinity_group_mappings: - primary_name: affinity_prod secondary_name: affinity_recovery
アフィニティーラベルの詳細	仮想マシンが属するアフィニティーラベルをマッピングします。以下はその例です。 dr_affinity_label_mappings: - primary_name: affinity_label_prod secondary_name: affinity_label_recovery
ドメイン AAA の詳細	ドメイン AAA (Authentication、Authorization、Accounting) 属性は、プライマリーサイトとセカンダリーサイト間で認可の詳細をマッピングします。以下はその例です。 dr_domain_mappings: - primary_name: internal-authz secondary_name: recovery-authz - primary_name: external-authz secondary_name: recovery2-authz
ロールの詳細	ロール属性は、特定のロールのマッピングを提供します。たとえば、仮想マシンが **VmCreator** ロールを持つユーザーに登録されている場合、フェイルオーバー時に Manager は、ロールが異なる同じユーザーにその仮想マシンへのアクセス許可を登録できます。以下はその例です。 dr_role_mappings: - primary_name: VmCreator Secondary_name: NewVmCreator

マッピングファイルのセクション	説明
ネットワークの詳細	ネットワーク属性は、プライマリーサイトとセカンダリーサイト間で vNIC の詳細をマッピングします。以下はその例です。 dr_network_mappings: - primary_network_name: ovirtmgmt primary_profile_name: ovirtmgmt primary_profile_id: 0000000a-000a-000a-000a-000000000398 # Fill in the correlated vnic profile properties in the secondary site for profile 'ovirtmgmt' secondary_network_name: ovirtmgmt secondary_profile_name: ovirtmgmt secondary_profile_id: 0000000a-000a-000a-000a-000000000410 複数のネットワークまたは複数のデータセンターがある場合は、マッピングファイルで空のネットワークマッピングを使用して、フェイルオーバー時にすべてのエンティティーがターゲットに登録されるようにする必要があります。以下はその例です。 dr_network_mappings: # No mapping should be here

マッピングファイルのセクション	説明
外部 LUN ディスクの詳細	外部 LUN 属性を使用すると、フェイルオーバーおよびフェイルバック後に仮想マシンを適切な外部 LUN ディスクに登録できます。以下はその例です。 dr_lun_mappings: - primary_logical_unit_id: 460014069b2be431c0fd46c4bdce29b66 primary_logical_unit_alias: Fedora_Disk primary_wipe_after_delete: False primary_shareable: False primary_logical_unit_description: 2b66 primary_storage_type: iscsi primary_logical_unit_address: 10.35.xx.xxx primary_logical_unit_port: 3260 primary_logical_unit_portal: 1 primary_logical_unit_target: iqn.2017-12.com.prod.example:444 secondary_storage_type: iscsi secondary_wipe_after_delete: False secondary_shareable: False secondary_logical_unit_id: 460014069b2be431c0fd46c4bdce29b66 secondary_logical_unit_address: 10.35.x.xxx secondary_logical_unit_port: 3260 secondary_logical_unit_portal: 1 secondary_logical_unit_target: iqn.2017-12.com.recovery.example:444

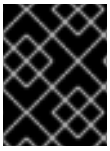
付録B ACTIVE-PASSIVE 設定のテスト

障害復旧ソリューションは、設定後にテストする必要があります。このセクションでは、Active-Passive 障害復旧の設定をテストする複数のオプションを説明します。

1. プライマリーサイトがアクティブで、プライマリーサイトのストレージドメイン上の仮想マシンと干渉しない状態で、フェイルオーバーをテストします。ディスクリートフェイルオーバーテスト を参照してください。

2. プライマリーサイトにアタッチされている特定のストレージドメインを使用してフェイルオーバーとフェイルバックをテストするため、プライマリーサイトはアクティブな状態に保たれます。フェイルオーバーとフェイルバックのディスクリートテスト を参照してください。

3. セカンダリーサイトにフェイルオーバーするための猶予期間がある、またはプライマリーサイトの計画外のシャットダウンがある、差し迫った障害のフェイルオーバーとフェイルバックをテストします。完全なフェイルオーバーとフェイルバックのテスト を参照してください。



重要

これらのテストのいずれかを実行する前に、Active-Passive 設定の手順をすべて完了してください。

B.1. ディスクリートフェイルオーバーテスト

このテストは、プライマリーサイトとそのすべてのストレージドメインがアクティブな状態のままフェイルオーバーをシミュレートします。そのため、ユーザーはプライマリーサイトで作業を継続できます。このシナリオを有効にするには、プライマリーストレージドメインとレプリケートされた (セカンダリー) ストレージドメイン間のレプリケーションを無効にする必要があります。このテスト中、プライマリーサイトはセカンダリーサイトのフェイルオーバーアクティビティーを認識しません。

このテストでは、フェイルバック機能をテストすることはできません。

重要

フェイルオーバー後に実稼働タスクが実行されていないことを確認してください。たとえば、電子メールシステムが実際のユーザーに電子メールを送信するのをブロックするか、電子メールを別の場所にリダイレクトするようにします。システムを使用して他のシステムを直接管理する場合は、システムへのアクセスを禁止するか、セカンダリーサイトの並列システムにアクセスするようにしてください。

ディスクリートフェイルオーバーテストの実行:

1. プライマリーストレージドメインと複製されたストレージドメイン間のストレージレプリケーションを無効にし、レプリケートされたすべてのストレージドメインが読み取り/書き込みモードであることを確認します。

2. 次のコマンドを実行して、セカンダリーサイトにフェイルオーバーします。

```
# ansible-playbook playbook --tags "fail_over"
```

   詳細は、フェイルバックの実行 を参照してください。

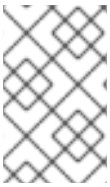
3. 関連するストレージドメイン、仮想マシン、およびテンプレートがすべて登録され、実行されていることを確認します。

環境を Active-Passive 状態に復元:

1. セカンダリーサイトからストレージドメインを切り離します。
2. プライマリーストレージドメインとセカンダリーストレージドメイン間のストレージレプリケーションを有効にします。

B.2. フェイルオーバーとフェイルバックのディスクリートテスト

このテストでは、フェイルオーバーとフェイルバックのテスト専用として使用する、テスト可能なストレージドメインを定義する必要があります。レプリケートされたストレージをセカンダリーサイトにアタッチできるように、これらのストレージドメインをレプリケートする必要があります。これにより、ユーザーがプライマリーサイトで作業を続けている間にフェイルオーバーをテストできます。



注記

プライマリーサイトで実稼働に使用するプライマリーストレージドメインに影響を与えずに、別のストレージサーバーでテスト可能なストレージドメインを定義する必要があります。

環境のフェイルオーバー、環境のクリーニング、フェイルバックの実行について、詳しくは フェイルオーバーの実行、プライマリーサイトのクリーニング、および フェイルバックの実行 を参照してください。

手順: ディスクリートフェイルオーバーテスト

1. プライマリーサイトでテストストレージドメインを停止します。これを行うには、たとえば、サーバーホストをシャットダウンするか、ファイアウォールルールでサーバーホストをブロックします。
2. テスト可能なストレージドメイン間でのストレージレプリケーションを無効にし、テストに使用するレプリケートされたすべてのストレージドメインが読み取り/書き込みモードであることを確認します。
3. テストプライマリーストレージドメインを読み取り専用モードにします。
4. 次のコマンドを実行して、セカンダリーサイトにフェイルオーバーします。

```
# ansible-playbook playbook --tags "fail_over"
```

5. 関連するストレージドメイン、仮想マシン、およびテンプレートがすべて登録され、実行されていることを確認します。

手順: ディスクリートフェイルバックテスト

1. コマンドを実行してプライマリーサイトをクリーンアップし、すべての非アクティブなストレージドメインと関連する仮想マシンおよびテンプレートを削除します。

```
# ansible-playbook playbook --tags "clean_engine"
```

2. フェイルバックコマンドを実行します。

```
# ansible-playbook playbook --tags "fail_back"
```

3. プライマリーストレージドメインからセカンダリーストレージドメインへのレプリケーションを有効にします。

4. 関連するストレージドメイン、仮想マシン、およびテンプレートがすべて登録され、実行されていることを確認します。

B.3. 完全なフェイルオーバーおよびフェイルバックのテスト

このテストは、プライマリーサイトとセカンダリーサイト間の完全なフェイルオーバーとフェイルバックをテストします。プライマリーサイトのホストをシャットダウンするか、ストレージドメインへの書き込みをブロックするファイアウォールのルールを追加して、障害をシミュレートできます。

環境のフェイルオーバー、環境のクリーニング、フェイルバックの実行について、詳しくは フェイルオーバーの実行、プライマリーサイトのクリーニング、および フェイルバックの実行 を参照してください。

手順: フェイルオーバーテスト

1. プライマリーストレージドメインと複製されたストレージドメイン間のストレージレプリケーションを無効にし、複製されたすべてのストレージドメインが読み取り/書き込みモードであることを確認します。

2. 次のコマンドを実行して、セカンダリーサイトにフェイルオーバーします。

```
# ansible-playbook playbook --tags "fail_over"
```

3. 関連するストレージドメイン、仮想マシン、およびテンプレートがすべて登録され、実行されていることを確認します。

手順: フェイルバックテスト

1. セカンダリーサイトのストレージドメインとプライマリーサイトのストレージドメイン間のレプリケーションを同期します。セカンダリーサイトのストレージドメインは読み取り/書き込みモード、プライマリーサイトのストレージドメインは読み取り専用モードである必要があります。

2. コマンドを実行してプライマリーサイトをクリーンアップし、すべての非アクティブなストレージドメインと関連する仮想マシンおよびテンプレートを削除します。

```
# ansible-playbook playbook --tags "clean_engine"
```

3. フェイルバックコマンドを実行します。

```
# ansible-playbook playbook --tags "fail_back"
```

4. プライマリーストレージドメインからセカンダリーストレージドメインへのレプリケーションを有効にします。

5. 関連するストレージドメイン、仮想マシン、およびテンプレートがすべて登録され、実行されていることを確認します。

付録C 法的通知