



# Red Hat Hyperconverged Infrastructure 1.0

## Red Hat Hyperconverged Infrastructure のデプロイ

---

Red Hat Hyperconverged Infrastructure のデプロイ手順

Laura Bailey



# Red Hat Hyperconverged Infrastructure 1.0 Red Hat Hyperconverged Infrastructure のデプロイ

---

## Red Hat Hyperconverged Infrastructure のデプロイ手順

Laura Bailey  
lbailey@redhat.com

## 法律上の通知

Copyright © 2017 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

本ガイドでは、Red Hat Gluster Storage 3.2 および Red Hat Enterprise Virtualization 4.1 を使用して、3 台の物理マシンにまたがって Red Hat Hyperconverged Infrastructure (RHHI) をデプロイする方法について説明します。これにより、リモートオフィス/ブランチオフィス (ROBO: Remote Office Branch Office) 環境で使用するための分離されたクラスターが作成されます。このクラスターでは、リモートオフィスのデータが中央データセンターと定期的に同期されますが、中央データセンターへの接続が切断されても完全に機能する状態を維持することができます。

## 目次

第1章 既知の問題 .....	3
パート I. プラン .....	5
第2章 アーキテクチャー .....	6
第3章 サポート要件 .....	7
3.1. オペレーティングシステム	7
3.2. 物理マシン	7
3.3. 仮想マシン	7
3.4. ネットワーク	8
3.5. ストレージ	8
3.5.1. アーキテクチャー	8
3.5.2. RAID	8
3.5.3. JBOD	9
3.5.4. ボリュームタイプ	9
3.6. サポートの制限事項	9
3.7. 障害復旧	10
パート II. デプロイ .....	11
第4章 デプロイメントワークフロー .....	12
第5章 ホスト用物理マシンのインストール .....	13
第6章 公開鍵ベースの SSH 認証の設定 .....	14
第7章 COCKPIT UI を使用したセルフホストエンジン向けの RED HAT GLUSTER STORAGE の設定 .....	15
第8章 COCKPIT UI を使用したセルフホストエンジンのデプロイ .....	22
第9章 RED HAT GLUSTER STORAGE AS A RED HAT VIRTUALIZATION のストレージドメインの設定 ...	26
9.1. GLUSTER トラフィック用の論理ネットワークの作成	26
9.2. マスターストレージドメインの作成	27
9.3. セルフホストエンジンへの残りの仮想化ホストの追加	27
パート III. トラブルシューティング .....	29
第10章 セルフホストエンジンのデプロイメントのエラー処理 .....	30
第11章 RED HAT GLUSTER STORAGE デプロイメントエラーの自動クリーンアップ .....	31
パート IV. 確認 .....	32
第12章 デプロイメントの確認 .....	33
パート V. 参考資料 .....	34
付録A デプロイメント中の暗号化設定 .....	35
A.1. 認証局で署名された証明書を使用したデプロイメント中の TLS/SSL 設定	35
A.1.1. 作業を開始する前の注意事項	35
A.1.2. CA 署名済み証明書を使用した TLS/SSL の暗号化設定	35
A.2. 自己署名証明書を使用したデプロイメント中の TLS/SSL 暗号化設定	35
付録B GDEPLOY 向けのクリーンアップ設定ファイルの例 .....	37
付録C 生成された GDEPLOY 設定ファイルについての理解 .....	38



## 第1章 既知の問題

本項には、Red Hat Hyperconverged Infrastructure (RHHi) に影響を及ぼす予期せぬ動作が発生する既知の問題について記載します。

### BZ#1395087: ネットワークが利用できない時に仮想マシンが無期限に一時停止する

Gluster および移行トラフィック用のネットワークが利用できなくなった場合には、I/O を実行している仮想マシンにアクセスできなくなり、そのハイパーバイザーが再起動されるまで、別のノードへの移行は完了できません。これは、現在のフェンシングとマイグレーションの方法では予想される動作です。現在、この問題に対する回避策はありません。

### BZ#1401969: Arbiter ブリックが自己修復のデータソースとなってしまう

Arbiter ボリューム内のデータブリックがオフラインになった後に1つずつオンラインに戻されると、他のブリックが自己修復される際に、その Arbiter ブリックは正しいデータのソースとして誤って特定されてしまいます。その場合、Arbiter ブリックにはメタデータしか含まれていないため、仮想マシンは一時停止されます。現在、この問題に対する回避策はありません。

### BZ#1412930: TLS/SSL が有効化されている場合にストレージが利用できないと過度にログ記録される

Red Hat Gluster Storage ボリュームで Transport Layer Security (TLS/SSL) が有効化されている場合に、Red Hat Gluster Storage サーバーが利用できなくなると、Red Hat Gluster Storage サーバーが再度利用可能になるまでは、多数の接続エラーメッセージがログ記録されます。これは、再接続を試みた後に、ログ記録されたメッセージが下位のログメッセージに変更されないために発生します。現在、この問題に対する回避策はありません。

### BZ#1413845: 管理ネットワークが利用できない場合にはセルフホストエンジンは移行しない

マイグレーションの実行中に管理ネットワークが利用できなくなると、セルフホストエンジン用の仮想マシンは再起動しますが、セルフホストエンジン自体は再起動しません。この問題を回避するには、ovirt-engine サービスを手動で再起動してください。

### BZ#1425767: サニティーチェックスクリプトが失敗しない

ディスクが存在しない場合や空の場合でも、サニティーチェックのスクリプトでゼロ (成功) が返される場合があります。サニティーチェックが成功したように見えるため、gdeploy は物理ボリュームの作成を試みて失敗します。この問題を回避するには、gdeploy の設定ファイルで **disk** の値が正しいことと、そのディスクでパーティションやラベルがないことを確認してからデプロイメントを再試行してください。

### BZ#1432326: ネットワークをホストに関連付けるとネットワークが同期されなくなる

Gluster ネットワークが Red Hat Gluster Storage ノードのネットワークインターフェースに関連付けられると、Gluster ネットワークは非同期の状態に切り替わります。この問題を回避するには、ノードの対応する **管理** のタブをクリックして、**機能をリフレッシュ** をクリックしてください。

### BZ#1434105: ストレージのライブマイグレーションの失敗

移行中に I/O 操作が進行中の場合には、Gluster ベースのストレージドメインのライブマイグレーションは失敗します。現在、この問題に対する回避策はありません。

### BZ#1437799: SSH を使用した Gluster ストレージドメインへの ISO アップロードが失敗する

SSH を使用して Gluster ベースのストレージドメインに ISO をアップロードする場合に、**ovirt-iso-uploader** ツールが誤ったパスを使用するために、以下のようなエラー

が表示されて操作が失敗します。

```
OSError: [Errno 2] No such file or directory: '/ISO_Volume'  
ERROR: Unable to copy RHGSS-3.1.3-RHEL-7-20160616.2-RHGSS-x86_64-  
dvd1.iso to ISO storage domain on ISODomain1.  
ERROR: Error message is "unable to test the available space on  
/ISO_Volume"
```

この問題を回避するには、Gluster ボリューム上の NFS アクセスを有効にして、SSH の代わりに NFS を使用してアップロードしてください。

#### **BZ#1439069:** ノードを再インストールすると、`/etc/ovirt-hosted-engine/hosted-engine.conf` が上書きされる

Red Hat Gluster Storage のプライマリサーバーが置き換えられた後にノードを再インストールすると、`/etc/ovirt-hosted-engine/hosted-engine.conf` ファイルの内容は、古いプライマリホストの情報で上書きされるため、クラスターで非稼働状態となってしまいます。

この問題を回避するには、再インストールしたノードをメンテナンスモードに切り替えて `/etc/ovirt-hosted-engine/hosted-engine.conf` 内容を更新し、置き換えたプライマリサーバーをポイントするようにします。次に再インストールしたノードを再起動し、再度アクティブ化してオンライン状態にし、全ボリュームをマウントします。

#### **BZ#1443169:** ブリッジの設定中にセルフホストエンジンのデプロイが失敗する

ブリッジネットワーク設定を使用するシステム上にセルフホストエンジンをデプロイすると、`firewalld` サービスの再起動後に設定が失敗します。この問題を回避するには、セルフホストエンジンをデプロイする前に、`/etc/sysconfig/network-scripts/` ディレクトリから `*.bak` ファイルを削除してください。



## パート I. プラン

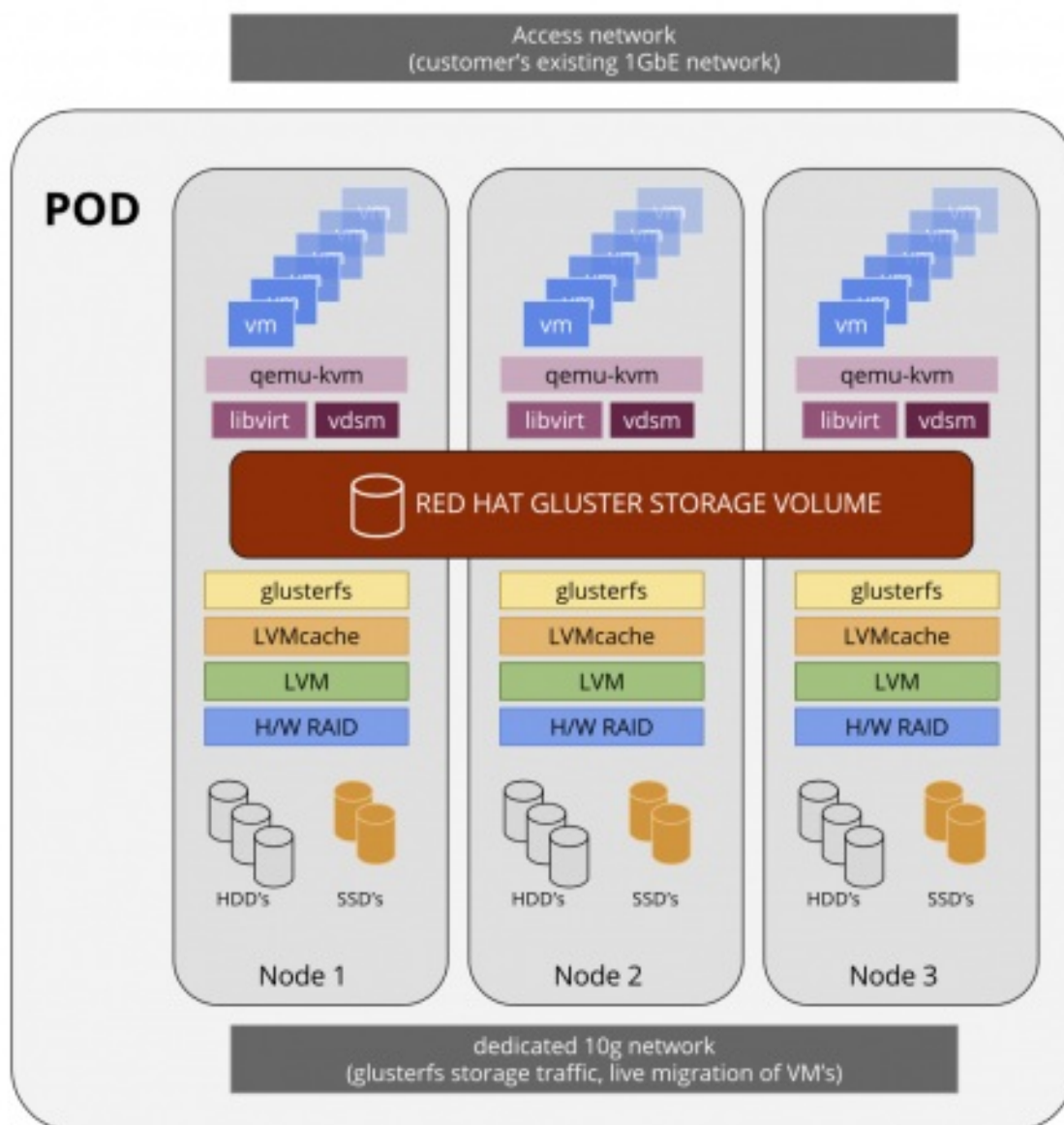
## 第2章 アーキテクチャー

Red Hat Hyperconverged Infrastructure (RHHI) はコンピュータ、ストレージ、ネットワーク、管理の機能を単一のデプロイメント内に統合します。

RHHI は、3 台の物理マシンにまたがってデプロイして、Red Hat Gluster Storage 3.2 および Red Hat Virtualization 4.1 を使用する分離されたクラスターまたは **pod** を作成します。

このデプロイメントの主要なユースケースは、リモートオフィス/ブランチオフィス (ROBO: Remote Office Branch Office) 環境で、リモートオフィスは中央データセンターと定期的にデータを同期しますが、中央データセンターに接続されていないなくても機能します。

以下の図には、単一クラスターの基本的なアーキテクチャーを示しています。



## 第3章 サポート要件

本項の記載内容をチェックして、計画中のデプロイメントが Red Hat のサポート要件を満たしていることを確認してください。

### 3.1. オペレーティングシステム

Red Hat Hyperconverged Infrastructure (RHHI) は Red Hat Virtualization Host 4.1 でのみサポートされています。その他の設定にはすべて Red Hat Virtualization Host 4.1 をベースとして使用してください。

Red Hat Virtualization の要件については、Red Hat Virtualization 『インストールガイド』を参照してください: [https://access.redhat.com/documentation/en-us/red\\_hat\\_virtualization/4.1/html/installation\\_guide/chap-system\\_requirements](https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.1/html/installation_guide/chap-system_requirements)

### 3.2. 物理マシン

Red Hat Hyperconverged Infrastructure (RHHI) には、**少なくとも 3 台の物理マシン**が必要です。物理マシン 6 台または 9 台へのスケーリングもサポートされています。

各物理マシンには、以下の機能が必要です。

- ※ データトラフィックと管理トラフィックを分離するために、物理マシン 1 台につき少なくとも 2 つの NIC が搭載されていること (詳しくは「[ネットワーク](#)」を参照)。
- ※ 小型のデプロイメントの場合:
  - 最小 12 コア
  - 最小 64 GB のメモリー
  - 最大 48 TB のストレージ
- ※ 中型のデプロイメントの場合:
  - 最小 12 コア
  - 最小 128 GB のメモリー
  - 最大 64 TB のストレージ
- ※ 大型のデプロイメントの場合:
  - 最小 16 コア
  - 最小 256 GB のメモリー
  - 最大 80 TB のストレージ

### 3.3. 仮想マシン

各仮想マシンには、最大で 4 つの仮想 CPU と 2 TB の仮想ディスク容量を割り当てることができます。

サポートされている仮想マシン数はそれらのサイズとリソースの使用状況によって異なります。

## 3.4. ネットワーク

クラスター内のクライアントトラフィックと管理トラフィックを分けるために2つの分離したネットワークが必要です。

### フロントエンドネットワーク

このネットワークは、ovirt の管理用ネットワークブリッジとして使用されます。

- ※ フロントエンドの IP アドレスは、同じサブネットから割り当てられる必要があるが、バックエンドの IP アドレスとは異なるサブネットである必要があります。

### バックエンドネットワーク

このネットワークはストレージトラフィックと、ストレージピア間の移行トラフィックに使用されます。

- ※ Red Hat は、バックエンドトラフィックに 10 Gbps のネットワークを推奨しています。
- ※ Red Hat Gluster Storage では、ピア間の最大レイテンシーが 5 ミリ秒必要です。
- ※ バックエンドの IP アドレスは、同じサブネットから割り当てられる必要があり、フロントエンドの IP アドレスとは異なるサブネットである必要があります。

全ホストの FQDN とセルフホストエンジン用仮想マシンの FQDN は、DNS で正引き/逆引きの名前解決可能である必要があります。

DHCP サーバーは、セルフホストエンジン用の仮想マシンの DHCP ネットワーク設定を選択する際に必要です。

## 3.5. ストレージ

### 3.5.1. アーキテクチャー

単一の Red Hat Gluster Storage クラスターには 3 つから 4 つのボリュームを使用することができます。

- ※ セルフホストエンジン用の **engine** ボリュームを 1 つ
- ※ 仮想マシンのブートディスクイメージ用の **vmstore** ボリュームを 1 つ
- ※ その他の仮想マシンディスクイメージ用にオプションの **data** ボリュームを 1 つ
- ※ ジオレプリケーションのメタデータ用の **shared\_storage** ボリュームを 1 つ

Red Hat Gluster Storage クラスターに使用できるジオレプリケーション用ボリュームは最大で 1 つです。

Red Hat はさらに、各サーバーのローカルにホットスペアドライブを少なくとも 1 つ用意することを推奨します。

### 3.5.2. RAID

RAID 設定の制限は、使用されているテクノロジーによって異なります。

- ※ SAS/SATA 7k のディスクは、RAID6 でサポートされています (最大で 10+2)。

- ※ SAS 10 k および 15 k のディスクは、以下でサポートされています。
  - RAID10
  - RAID5 (最大 7+1)
  - RAID6 (最大 10+2)

RAID カードは、Flash Backed Write Cache を使用する必要があります。

### 3.5.3. JBOD

JBOD の設定には、アーキテクチャーのレビューが必要です。詳しくは、Red Hat の担当者までお問い合わせください。

### 3.5.4. ボリュームタイプ

Red Hat Hyperconverged Infrastructure (RHII) は **replicated** または **Arbitrated Replicated** のボリュームタイプのみをサポートしています。

Arbitrated Replicated のボリュームタイプには、以下のような追加の制限事項があります。

- ※ Arbitrated Replicated ボリュームは、最初の 3 ノードでのみサポートされます。
- ※ Arbitrated Replicated ボリュームは **replica 3 arbiter 1** の構成を使用する必要があります。
- ※ Arbiter ブリックはファイルのデータは保管せず、ファイル名、構造、およびメタデータのみを保管します。

これは、3 方向の Arbitrated Replicated ボリュームが同程度の整合性を実現するために、3 方向の replicated ボリュームに必要なストレージ容量の約 75% が必要であることを意味します。

ただし、Arbiter ブリックはメタデータのみを保管するため、3 方向の Arbitrated Replicated ボリュームが提供するのは 2 方向の replicated ボリュームの可用性のみです。

詳しい情報は、Red Hat Gluster Storage 3.2 『**Administration Guide**』を参照してください：  
[https://access.redhat.com/documentation/en-us/red\\_hat\\_gluster\\_storage/3.2/html-single/administration\\_guide/#Creating\\_Arbitrated\\_Replicated\\_Volumes](https://access.redhat.com/documentation/en-us/red_hat_gluster_storage/3.2/html-single/administration_guide/#Creating_Arbitrated_Replicated_Volumes)

## 3.6. サポートの制限事項

- ※ Red Hat Hyperconverged Infrastructure (RHII) の初回のデプロイメントでは、1 つの Arbitrated Replicated ボリュームがサポートされています。この Arbitrated Replicated ボリュームの拡張はサポートされていません。また、Arbitrated Replicated ボリュームの追加はサポートされていません。
- ※ Arbitrated Replicated ボリュームは、現在は **replica 3 arbiter 1** 構成でのみサポートされています。

ボリュームのタイプについての詳しい情報は、Red Hat Gluster Storage 『**Administration Guide**』を参照してください。

- ※ [https://access.redhat.com/documentation/en-us/red\\_hat\\_gluster\\_storage/3.2/html/administration\\_guide/sect-creating\\_replicated\\_volumes](https://access.redhat.com/documentation/en-us/red_hat_gluster_storage/3.2/html/administration_guide/sect-creating_replicated_volumes)

- ※ [https://access.redhat.com/documentation/en-us/red\\_hat\\_gluster\\_storage/3.2/html/administration\\_guide/creating\\_arbitrated\\_replicated\\_volumes](https://access.redhat.com/documentation/en-us/red_hat_gluster_storage/3.2/html/administration_guide/creating_arbitrated_replicated_volumes)

### 3.7. 障害復旧

Red Hat では、障害復旧ソリューションを設定することを強く推奨します。ジオレプリケーションを障害復旧ソリューションとして設定する方法についての詳しい情報は、『**Maintaining Red Hat Hyperconverged Infrastructure**』を参照してください:

[https://access.redhat.com/documentation/en-us/red\\_hat\\_hyperconverged\\_infrastructure/1.0/html/maintaining\\_red\\_hat\\_hyperconverged\\_infrastructure/](https://access.redhat.com/documentation/en-us/red_hat_hyperconverged_infrastructure/1.0/html/maintaining_red_hat_hyperconverged_infrastructure/)

ジオレプリケーションを設定する際には、以下のようなサポートの制限事項がある点に注意してください。

- ※ Red Hat Hyperconverged Infrastructure (RHAI) は 1 ボリュームのみでのジオレプリケーション設定をサポートしています。Red Hat は、仮想マシンのデータを保管するボリュームをバックアップすることを推奨します。
- ※ ジオレプリケーションの複製元と複製先のボリュームは、Red Hat Virtualization Manager の異なるインスタンスで管理する必要があります。

## パート II. デプロイ

## 第4章 デプロイメントワークフロー

Red Hat Hyperconverged Infrastructure (RHHi) のデプロイメントのワークフローは以下のとおりです。

1. 計画中のデプロイメントが「[3章 サポート要件](#)」を満たしているかどうかを確認します。
2. 仮想ホストとして機能する「[5章 ホスト用物理マシンのインストール](#)」を実行します。
3. 「[6章 公開鍵ベースの SSH 認証の設定](#)」でホストの自動設定を有効します。
4. Cockpit UI を使用して、ホストで「[7章 Cockpit UI を使用したセルフホストエンジン向けの Red Hat Gluster Storage の設定](#)」を実行します。
5. Cockpit UI を使用して「[8章 Cockpit UI を使用したセルフホストエンジンのデプロイ](#)」を実行します。
6. Red Hat Virtualization の管理 UI を使用して、Red Hat Gluster Storage ノードから「[9章 Red Hat Gluster Storage as a Red Hat Virtualization のストレージドメインの設定](#)」を実行します。



## 第5章 ホスト用物理マシンのインストール

Red Hat Virtualization Host 4.1 を 3 台の物理マシンにインストールします。仮想化ホストのインストールについての説明は、次のリンク先を参照してください:

<https://access.redhat.com/documentation/en/red-hat-virtualization/4.1/paged/installation-guide/chapter-6-red-hat-virtualization-hosts>

## 第6章 公開鍵ベースの SSH 認証の設定

最初の仮想化ホストから、全仮想化ホストに対して、root ユーザーの公開鍵認証ベースの SSH を設定します。

バックエンドの管理ネットワークに関連付けられたホスト名または IP アドレスを必ず使用してください。

詳しくは、Red Hat Enterprise Linux 7 『インストールガイド』を参照してください:

[https://access.redhat.com/documentation/en-](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/s1-ssh-configuration.html#s2-ssh-configuration-keypairs)

[US/Red\\_Hat\\_Enterprise\\_Linux/7/html/System\\_Administrators\\_Guide/s1-ssh-configuration.html#s2-ssh-configuration-keypairs](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/s1-ssh-configuration.html#s2-ssh-configuration-keypairs)

## 第7章 COCKPIT UI を使用したセルフホストエンジン向けの RED HAT GLUSTER STORAGE の設定



### 重要

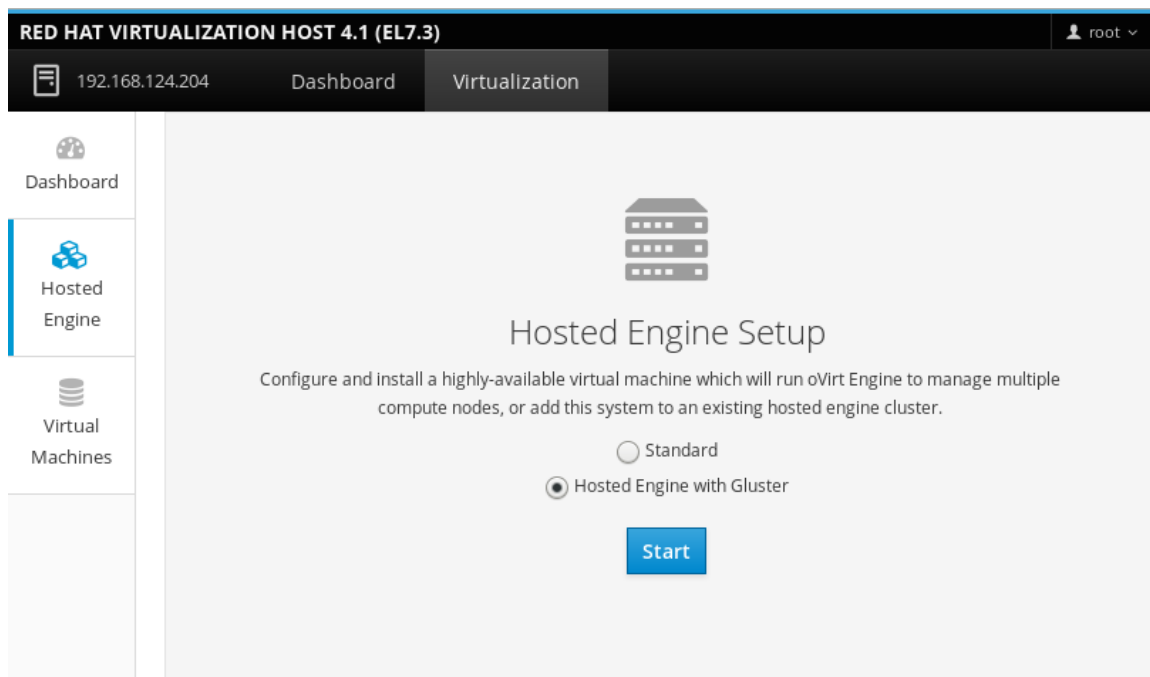
このデプロイメントプロセスの一部として指定されているディスクにはパーティションまたはラベルがないことを確認してください。

#### 1. Cockpit UI へのログイン

最初の仮想化ホストの Cockpit 管理インターフェース (例: <https://node1.example.com:9090/>) をブラウザで開いて、「**ホスト用物理マシンのインストール**」のステップで作成した認証情報を使用してログインします。

#### 2. デプロイメントウィザードの起動

Virtualization > Hosted Engine をクリックして **Hosted Engine with Gluster** を選択し、Start をクリックすると、デプロイメントウィザードが表示されます。



#### 3. デプロイメントウィザード: ストレージホストの指定

3つの仮想化ホストのバックエンド Gluster ネットワークアドレス (管理ネットワークアドレスではない) を指定します。キーペアを使用して SSH 接続が可能な仮想化ホストは、gdeploy とセルフホストエンジンを実行するホストなので、最初にリストする必要があります。



### 注記

Arbitrated Replicated ポリリュームを作成する予定の場合には、ブリックのあるホストをこの画面で **Host3** として指定するようにしてください。

Gluster Deployment
✕

Hosts      Packages      Volumes      Bricks      Review

① — ② — ③ — ④ — ⑤

---

Host1	<input type="text" value="192.168.33.101"/>	
Host2	<input type="text" value="192.168.33.102"/>	
Host3	<input type="text" value="192.168.33.103"/>	

[⊕ Add Host](#)

i gdeploy will login to gluster hosts as root user using passwordless ssh connections. Make sure, passwordless ssh is configured for all gluster hosts.

Cancel    < Back    Next >

「Next」をクリックします。

#### 4. デプロイメントウィザード: パッケージの指定

パッケージをインストールする必要はありません。これらのフィールドの値はすべて削除し、チェックボックスからはチェックを外してください。

Gluster Deployment
✕

Hosts      Packages      Volumes      Bricks      Review

① — ② — ③ — ④ — ⑤

---

CDN Username	<input type="text"/>
CDN Password	<input type="text"/>
Pool ID	<input type="text"/>
Repositories	<input type="text"/>
Packages	<input type="text"/>

Update Hosts

Cancel    < Back    Next >

「Next」をクリックします。

## 5. デプロイメントウィザード: ボリュームの指定

作成するボリュームを指定します。

Gluster Deployment
✕

Hosts      Packages      Volumes      Bricks      Review

① — ② — ③ — ④ — ⑤

---

Name	Volume Type	Arbiter	Brick Dirs	
<input type="text" value="engine"/>	<span>Replicate</span> ▼	<input type="checkbox"/>	<input type="text" value="/gluster_bricks/engine/engine"/>	
<input type="text" value="data"/>	<span>Replicate</span> ▼	<input checked="" type="checkbox"/>	<input type="text" value="/gluster_bricks/data/data"/>	
<input type="text" value="vmstore"/>	<span>Replicate</span> ▼	<input checked="" type="checkbox"/>	<input type="text" value="/gluster_bricks/vmstore/vmsto"/>	

[+ Add Volume](#)

ⓘ First volume in the list will be used for hosted-engine deployment

Cancel    < Back    Next >

### Name

作成するボリュームの名前を指定します。

### Volume Type

**Replicate** のボリュームタイプを指定します。本リリースでは、replicated ボリュームのみがサポートされています。

### Arbiter

Arbiter ブリックを使用するボリュームを作成するかどうかを指定します。このチェックボックスを選択した場合には、3 番目のディスクにはメタデータのみが保管されます。

### Brick Dirs

このボリュームのブリックが含まれているディレクトリ。

デフォルト値は、大半のインストールで適正です。

## 6. デプロイメントウィザード: ブリックの指定

作成するブリックを指定します。

Gluster Deployment
✕

Hosts      Packages      Volumes      4 Bricks      Review

① — ② — ③ — 4 — ⑤

---

**Raid Information** ⓘ

Raid Type: RAID 6

Stripe Size(KB): 256

Data Disk Count: 12

**Brick Configuration**

LV Name	Device Name	Size(GB)	Thinp	Mount Point
<input type="text" value="engine"/>	<input type="text" value="sdb"/>	<input type="text" value="100"/>	<input type="checkbox"/>	<input type="text" value="/gluster_bricks/engine"/> <span style="font-size: 0.8em;">🗑</span>
<input type="text" value="data"/>	<input type="text" value="sdb"/>	<input type="text" value="500"/>	<input checked="" type="checkbox"/>	<input type="text" value="/gluster_bricks/data"/> <span style="font-size: 0.8em;">🗑</span>
<input type="text" value="vmstore"/>	<input type="text" value="sdb"/>	<input type="text" value="500"/>	<input checked="" type="checkbox"/>	<input type="text" value="/gluster_bricks/vmstore"/> <span style="font-size: 0.8em;">🗑</span>

➕ Add Bricks

ⓘ Arbitrator bricks will be created on the third host in the host list.

Cancel
< Back
Next >

## LV Name

作成する論理ボリュームの名前を指定します。

## Device

使用する RAW デバイスを指定します。Red Hat では、パーティション分割されていないデバイスを推奨しています。

## Size

作成する論理ボリュームのサイズを GB で指定します。単位は入力せず、数値のみを入力してください。

## Mount Point

論理ボリュームのマウントポイントを指定します。これは、ウィザードの前のページで指定したブリックディレクトリーと一致する必要があります。

## Thinp

ボリュームをシンプロビジョニングするかどうかを指定します。**engine** 用のボリュームには、シックプロビジョニングが推奨される点に注意してください。

## RAID

使用する RAID 構成。これは、ホストの RAID 構成と一致する必要があります。サポートされている値は、**raid5**、**raid6**、**raid10**、**jbod** です。

### Stripe Size

RAID のストライプサイズを KB で指定します。単位は入力せず、数値のみを入力してください。JBOD 構成では、この設定は無視することができます。

### Disk Count

RAID ボリューム内のデータディスクの数を指定します。JBOD 構成では、この設定は無視することができます。

## 7. デプロイメントウィザード: 設定のレビューと編集

- Edit** をクリックして、生成された **gdeployConfig.conf** ファイルの編集を開始します。
- (オプション) Transport Layer Security (TLS/SSL) を設定します。

これは、デプロイメントの実行中または実行後に設定することができます。デプロイメントの一環として TLS/SSL 暗号化を設定する場合には、以下のいずれかの項を参照してください。

- ※ 「[認証局で署名された証明書を使用したデプロイメント中の TLS/SSL 設定](#)」
- ※ 「[自己署名証明書を使用したデプロイメント中の TLS/SSL 暗号化設定](#)」

- 設定ファイルをレビューします。

設定の詳細に問題がない場合には、「Save」をクリックしてから「Deploy」のボタンを押します。

## 8. デプロイメントの完了待ち

生成された設定ファイルを使用して gdeploy プロセスが実行されている間に、テキストフィールドでデプロイメントの経過を確認することができます。

9. デプロイが失敗した場合には、以下の手順を実行します。

The screenshot shows the 'Gluster Deployment' interface with a progress bar indicating the 'Review' step (5) is active. Below the progress bar, a terminal window titled 'Deployment failed' displays the following error logs:

```

[{"item": "/usr/share/ansible/gdeploy/scripts/grafon-sanity-check.sh -d sdb -h a,b,c", "msg": "Failed to connect to the host via ssh: ssh: Could not resolve hostname host3.lab.eng.blr.redhat.com: Name or service not known\n", "unreachable": true}]]
failed: [host3.lab.eng.blr.redhat.com] (item=/usr/share/ansible/gdeploy/scripts/grafon-sanity-check.sh -d sdb -h a,b,c) => {"item": "/usr/share/ansible/gdeploy/scripts/grafon-sanity-check.sh -d sdb -h a,b,c", "msg": "Failed to connect to the host via ssh: ssh: Could not resolve hostname host3.lab.eng.blr.redhat.com: Name or service not known\n", "unreachable": true}
to retry, use: --limit @/tmp/tmpaoAdes/run-script.retry

PLAY RECAP *****
host1.lab.eng.blr.redhat.com : ok=0    changed=0    unreachable=1    failed=0
host2.lab.eng.blr.redhat.com : ok=0    changed=0    unreachable=1    failed=0
host3.lab.eng.blr.redhat.com : ok=0    changed=0    unreachable=1    failed=0
  
```

At the bottom of the terminal window, there are three buttons: 'Cancel', '< Back', and 'Close'. A 'Redeploy' button is also visible in the top right corner of the terminal window.

デプロイメントが失敗した場合には、**Redeploy** ボタンをクリックします。**Review and edit configuration** タブが開き、デプロイメントの再実行前に生成済みの設定ファイル内の問題を修正することができます。

再デプロイする前に、前回に試行したデプロイメントをクリーンアップする必要がある場合があります。「[11章 Red Hat Gluster Storage デプロイメントエラーの自動クリーンアップ](#)」のステップにしたがってクリーンアップしてください。

10. デプロイメントスクリプトが完了し、「**Successfully deployed gluster.**」というメッセージが表示されます。



Gluster Deployment ✕



Successfully deployed Gluster

[Continue to Hosted Engine Deployment](#)

Cancel

< Back

Close

## 第8章 COCKPIT UI を使用したセルフホストエンジンのデプロイ

本項では、Cockpit UI を使用したセルフホストエンジンのデプロイ方法について説明します。この手順に従って操作を実行すると、デプロイメント内の 1 番目の物理マシン上に仮想マシンとして稼働する Red Hat Virtualization Manager がデプロイされます。また、3 台の物理マシンで構成される Default クラスターも設定され、クラスター内の各マシンに Red Hat Gluster Storage の機能と **virtual-host tuned** パフォーマンスプロファイルが有効になります。

1. セルフホストエンジンのデプロイメントウィザードに進みます。

「[7章Cockpit UI を使用したセルフホストエンジン向けの Red Hat Gluster Storage の設定](#)」が完了した後に、**Continue to Hosted Engine Deployment**をクリックしてウィザードに進みます。

2. インストールの実行に同意します。

```
Continuing will configure this host for serving as hypervisor
and create a VM where you have to install the engine afterwards.
Are you sure you want to continue?
```

表示されたフィールドに **Yes** と入力します。**Next** をクリックして、環境が設定されるのを待ちます。

3. プロンプトに表示されるデプロイメントについての質問に対する回答

セルフホストエンジンのデプロイメントウィザードのプロンプトに回答して、セルフホストエンジンのインストールと設定を行います。**Ctrl+D** キーを押して、プロセスを随時停止することができます。

表8.1 セルフホストエンジンのデプロイメントウィザードのプロンプト

プロンプトテキスト	アクション
Do you wish to install ovirt-engine-appliance rpm?	<b>Yes</b> と入力します。
Do you want to configure this host and its cluster for gluster?	<b>Yes</b> と入力します。
iptables was detected on your computer, do you wish setup to configure it?	<b>No</b> と入力します。
Please indicate a pingable gateway IP address	フロントエンドのゲートウェイサーバーの IP アドレスを入力します。

プロンプトテキスト	アクション
Please indicate a nic to set ovirtmgmt bridge on	フロントエンドの管理ネットワークアドレスを入力します。
The following appliance have been found on your system: [1] The RHV-M Appliance image (OVA) - 4.1.20170328.1.el7ev [2] Directly select an OVA file. Please select an appliance	セルフホストエンジンのアプライアンスの適切な数を入力します (通常は <b>1</b> )。
Would you like to use cloud-init to customize the appliance on first boot	<b>Yes</b> と入力します。
Would you like to generate on-fly a cloud-init ISO image (of no-cloud type) or do you have an existing one?	<b>Generate</b> と入力します。
Please provide the FQDN you would like to use for the engine appliance. Note: This will be the FQDN of the engine VM you are now going to launch, it should not point to the base host or to any other existing machine. Engine VM FQDN:	セルフホストエンジンに使用する完全修飾ドメイン名を入力します。
Please provide the domain name you would like to use for the engine appliance. Engine VM domain:	検出されたドメイン名が正しいことを確認します。
Automatically execute engine-setup on the engine appliance on first boot	<b>Yes</b> と入力します。
Enter root password that will be used for the engine appliance	セルフホストエンジンへのリモートログインに使用するパスワードを入力します。
Confirm appliance root password	セルフホストエンジンのリモートログインに使用するパスワードを再入力します。
Enter ssh public key for the root user that will be used for the engine appliance	このフィールドは空欄のままにします。

プロンプトテキスト	アクション
Do you want to enable ssh access for the root user	<b>Yes</b> と入力します。
Please specify the memory size of the VM in MB	デフォルト値を使用します。
The following CPU types are supported by this host:  Please specify the CPU type to be used by the VM	デフォルト値を使用します。
Please specify the number of virtual CPUs for the VM	デフォルト値を使用します。
You may specify a unicast MAC address for the VM or accept a randomly generated default	セルフホストエンジンに指定した完全修飾ドメイン名に名前解決する MAC アドレスを入力します。
How should the engine VM network be configured	<b>DHCP</b> と入力します。
Add lines for the appliance itself and for this host to /etc/hosts on the engine VM? (Note: ensuring that this host could resolve the engine VM hostname is still up to you)	<b>No</b> と入力します。
Enter engine admin password (for the RHV UI)	Red Hat Virtualization Manager の管理者が使用するパスワードを入力します。
Confirm engine admin password	Red Hat Virtualization Manager の管理者が使用するパスワードを再入力します。
Please provide the name of the SMTP server through which we will send notifications	デフォルト値を使用します。

プロンプトテキスト	アクション
Please provide the TCP port number of the SMTP server	デフォルト値を使用します。
Please provide the email address from which notifications will be sent	デフォルト値を使用します。
Please provide a comma-separated list of email addresses which will get notifications	セルフホストエンジンからの通知を受信するためのメールアドレスを入力します。デフォルト (root@localhost) で十分です。

#### 4. インストールの設定を確認します。

Please confirm installation settings

設定した値を見直し、正しいことを確認します。問題がない場合には、所定のフィールドに **Yes** と入力してから **Next** をクリックします。



#### 注記

現在は、プロセスのこの時点で設定を編集する方法はありません。設定の情報が正しくない場合には、このプロセスを再び開始して修正するか、プロセスを続行して、デプロイメント中に修正する必要があります。

#### 5. デプロイメントの完了待ち

これには、多少時間がかかります (約 30 分)。

デプロイメントが完了すると、以下のようなメッセージが表示されます。

Hosted Engine Setup successfully completed!

これで Red Hat Virtualization Manager にログインして設定を完了できる状態になりました。

デプロイメントが完了しなかった場合には、「[10章セルフホストエンジンのデプロイメントのエラー処理](#)」を参照してください。

## 第9章 RED HAT GLUSTER STORAGE AS A RED HAT VIRTUALIZATION のストレージドメインの設定

### 9.1. GLUSTER トラフィック用の論理ネットワークの作成

1. engine にログインします。

ブラウザで engine を開き、「[8章 Cockpit UI を使用したセルフホストエンジンのデプロイ](#)」で設定した管理者の認証情報を使用してログインします。

2. Gluster トラフィック用の論理ネットワークを作成します。
  - a. ネットワーク タブをクリックして、**新規作成** をクリックします。新規論理ネットワーク ウィンドウが表示されます。
  - b. このウィンドウの **全般** タブで、新規論理ネットワークの **名前** を指定し、**仮想マシンネットワーク** のチェックボックスからチェックを外します。
  - c. このウィンドウの **クラスター** タブで **必須** のチェックボックスを外します。
  - d. **OK** をクリックして新規論理ネットワークを作成します。
3. Gluster 用の新規論理ネットワークを有効化します。
  - a. ネットワーク タブをクリックして、新規論理ネットワークを選択します。
  - b. クラスターサブタブをクリックして、**ネットワークを管理** をクリックします。ネットワークの管理 ダイアログが表示されます。
  - c. ネットワークの管理 ダイアログで、**移行ネットワーク** と **Gluster ネットワーク** のチェックボックスにチェックを付けます。
  - d. **OK** をクリックして保存します。
4. Gluster ネットワークをホストにアタッチします。
  - a. ホスト タブをクリックして、対象のホストを選択します。
  - b. ネットワークインターフェースサブタブを選択して、**ホストネットワークを設定** をクリックします。
  - c. 新規作成したネットワークを正しいインターフェースにドラッグアンドドロップします。
  - d. ホストと **Engine** 間の**接続を検証** のチェックボックスに必ずチェックを入れます。
  - e. **ネットワーク設定を保存** のチェックボックスに必ずチェックを入れます。
  - f. **OK** をクリックして保存します。
5. ネットワークの正常性をチェックします。

ホスト タブをクリックして、対象のホストを選択します。ネットワークインターフェースサブタブをクリックして、ホストのネットワークの状態をチェックします。

ネットワークインターフェースが「非同期」状態に切り替わった場合や、IPv4 アドレスがない場合には、対応するホストの **管理** ドロップダウンメニューから **機能をリフレッシュ** を選択します。

## 9.2. マスターストレージドメインの作成

1. ストレージタブをクリックして、**新規ドメイン** をクリックします。
2. ストレージタイプに **GlusterFS** を選択し、そのドメインの**名前** を指定します。
3. **管理対象の Gluster ボリュームを使用する** オプションにチェックを入れます。

クラスター内で利用可能なボリュームの一覧が表示されます。

**vmstore** ボリュームを選択して、以下の行を **マウントオプション** に追加します。

```
backup-volfile-servers=server2:server3
```

4. **OK** をクリックして保存します。

マスターストレージドメインの作成後には、セルフホストエンジンのストレージドメインが自動的にインポートされます。



### 重要

セルフホストエンジン用の仮想マシンとそのストレージドメインが利用可能になるまで待ってから、他のタスクを続行します。こうすることにより、セルフホストエンジンのタブを使用できるようになります。

## 9.3. セルフホストエンジンへの残りの仮想化ホストの追加

他の各ホストに対して、Red Hat Virtualization Manager で以下の手順を実行します。

1. **ホスト** タブをクリックして、**新規作成** をクリックすると、**新規ホスト** ダイアログが開きます。
2. 新規ホストの **名前**、**アドレス**、**パスワード** を指定します。
3. ファイアウォールルールは **gdeploy** ですすでに設定済みなので、**ホストのファイアウォールを自動設定する** のチェックボックスからチェックを外します。
4. **新規ホスト** ダイアログの**セルフホストエンジン**タブで、**セルフホストエンジンのデプロイメントアクションの選択** の値を **デプロイ** に設定します。
5. **デプロイ** をクリックします。
6. Gluster ネットワークをホストにアタッチします。
  - a. **ホスト** タブをクリックして、対象のホストを選択します。
  - b. **ネットワークインターフェース** サブタブを選択して、**ホストネットワークを設定** をクリックします。
  - c. 新規作成したネットワークを正しいインターフェースにドラッグアンドドロップし

ます。

- d. **ホストと Engine 間の接続を検証** のチェックボックスに必ずチェックを入れます。
  - e. **ネットワーク設定を保存** のチェックボックスに必ずチェックを入れます。
  - f. **OK** をクリックして保存します。
7. ネットワークの正常性をチェックします。

**ホスト** タブをクリックして、対象のホストを選択します。**ネットワークインターフェース** サブタブをクリックして、ホストのネットワークの状態をチェックします。

ネットワークインターフェースが「非同期」状態に切り替わった場合や、IPv4 アドレスがない場合には、対応するホストの **管理** ドロップダウンメニューから **機能をリフレッシュ** を選択します。

詳しい情報は、Red Hat Virtualization 4.1 『セルフホストエンジンガイド』を参照してください:

[https://access.redhat.com/documentation/ja-jp/red\\_hat\\_virtualization/4.1/html/self-hosted\\_engine\\_guide/chap-installing\\_additional\\_hosts\\_to\\_a\\_self-hosted\\_environment](https://access.redhat.com/documentation/ja-jp/red_hat_virtualization/4.1/html/self-hosted_engine_guide/chap-installing_additional_hosts_to_a_self-hosted_environment)



## パート III. トラブルシュート

## 第10章 セルフホストエンジンのデプロイメントのエラー処理

セルフホストエンジンのデプロイメント中にエラーが発生した場合には、デプロイメントは一時停止され、以下のようなメッセージが表示されます。

```
Please check Engine VM configuration
Make a selection from the options below:
(1) Continue setup - Engine VM configuration has been fixed
(2) Abort setup
```

次にエラーの詳細が赤で表示されます。セルフホストエンジンにログインしてエラーを修正してから設定を続行することができます。**ovirt-engine** サービスは、修正を適用する前に停止し、**Continue setup** を選択する前に再起動する必要がある点に注意してください。

## 第11章 RED HAT GLUSTER STORAGE デプロイメントエラーの自動クリーンアップ

1. 「[付録B gdeploy 向けのクリーンアップ設定ファイルの例](#)」の `volume_cleanup.conf` ファイルをベースにして、`volume_cleanup.conf` ファイルを作成します。
2. `volume_cleanup.conf` ファイルを使用して `gdeploy` を実行します。

```
# gdeploy -c volume_cleanup.conf
```

3. 「[付録B gdeploy 向けのクリーンアップ設定ファイルの例](#)」の `lv_cleanup.conf` ファイルをベースにして、`lv_cleanup.conf` ファイルを作成します。
4. `lv_cleanup.conf` ファイルを使用して `gdeploy` を実行します。

```
# gdeploy -c lv_cleanup.conf
```

5. 全ホスト上でマウントの設定をチェックします。

全ホストで `/etc/fstab` ファイルをチェックして、自動作成されたブリックの XFS マウントの行をすべて削除します。

## パート IV. 確認

## 第12章 デプロイメントの確認

デプロイメントの完了時には、Red Hat Virtualization Manager を使用してデプロイメントのステータスを確認します。

仮想マシンを作成して、デプロイメントが想定通りに機能していることを確認します。詳しくは、Red Hat Virtualization 『**仮想マシン管理ガイド**』を参照してください:

[https://access.redhat.com/documentation/ja-jp/red\\_hat\\_virtualization/4.1/html/virtual\\_machine\\_management\\_guide/](https://access.redhat.com/documentation/ja-jp/red_hat_virtualization/4.1/html/virtual_machine_management_guide/)

## パート V. 参考資料

## 付録A デプロイメント中の暗号化設定

### A.1. 認証局で署名された証明書を使用したデプロイメント中の TLS/SSL 設定

#### A.1.1. 作業を開始する前の注意事項

開始する前に認証局による署名済みの適切な証明書があることを確認してください。証明書の取得方法については、本ガイドの対象範囲外です。

#### A.1.2. CA 署名済み証明書を使用した TLS/SSL の暗号化設定

1. 以下のファイルが全ノードの所定の場所に存在することを確認します。

**/etc/ssl/glusterfs.key**

ノードの秘密鍵

**/etc/ssl/glusterfs.pem**

認証局によって署名された証明書。これがノードの証明書となります。

**/etc/ssl/glusterfs.ca**

認証局の証明書。自己署名の設定の場合は、このファイルに全ノードの連結された証明書が含まれます。

2. 管理の暗号化を有効にします。

各ノードで **/var/lib/glusterd/secure-access** ファイルを作成します。

```
# touch /var/lib/glusterd/secure-access
```

3. 暗号化を設定します。

「[7章Cockpit UI を使用したセルフホストエンジン向けの Red Hat Gluster Storage の設定](#)」で生成された設定ファイルにリストされている各ボリュームに以下の行を追加します。これにより、デプロイメントプロセス中に CA 署名済み証明書を使用して Gluster ボリューム間の TLS/SSL ベースの暗号化が作成および設定されます。

```
key=client.ssl,server.ssl,auth.ssl-allow  
value=on,on,"host1;host2;host3"
```

生成されたファイルは、編集後に必ず保存してください。

### A.2. 自己署名証明書を使用したデプロイメント中の TLS/SSL 暗号化設定

「[7章Cockpit UI を使用したセルフホストエンジン向けの Red Hat Gluster Storage の設定](#)」で生成された設定ファイルに以下の行を追加し、デプロイメントプロセスで自己署名済み証明書を使用して Gluster ボリューム間の TLS/SSL ベースの暗号化を作成および設定します。gdeploy によって生成される証明書は 1 年間有効です。

最初のボリュームの設定で、**enable\_ssl** および **ssl\_clients** パラメーターそれらの値を指定する行を追加します。

```
[volume1]
enable_ssl=yes
ssl_clients=<Gluster_Network_IP1>,<Gluster_Network_IP2>,
<Gluster_Network_IP3>
```

その後のボリュームでは、以下の行を追加して **client.ssl**、**server.ssl**、**auth.ssl-allow** パラメーターの値を定義します。

```
[volumeX]
key=client.ssl,server.ssl,auth.ssl-allow
value=on,on,"<Gluster_Network_IP1>;<Gluster_Network_IP2>;
<Gluster_Network_IP3>"
```



## 付録B GDEPLOY 向けのクリーンアップ設定ファイルの例

デプロイメントが失敗した場合には、デプロイメントを再試行する前に、その前に試みたデプロイメントをクリーンアップする必要があります。以下の2つのサンプルファイルを `gdeploy` で実行して、以前に試みて失敗したデプロイメントをクリーンアップして再試行することができます。

### `volume_cleanup.conf`

```
[hosts]
<Gluster_Network_NodeA>
<Gluster_Network_NodeB>
<Gluster_Network_NodeC>

[volume1]
action=delete
volname=engine

[volume2]
action=delete
volname=vmstore

[volume3]
action=delete
volname=data

[peer]
action=detach
```

### `lv_cleanup.conf`

```
[hosts]
<Gluster_Network_NodeA>
<Gluster_Network_NodeB>
<Gluster_Network_NodeC>

[backend-reset]
pvs=sdb, sdc
unmount=yes
```

## 付録C 生成された GDEPLOY 設定ファイルについての理解

gdeploy は、設定ファイルに基づいて Red Hat Gluster Storage で 1 台または複数のマシンを自動でプロビジョニングします。

Cockpit UI は、Red Hat Hyperconverged Infrastructure のベースレベルデプロイメントを実行するのに適した gdeploy 設定ファイルをユーザーが生成することのできるウィザードを提供します。

本項では、Cockpit UI で以下の設定情報が指定されている場合に生成される gdeploy の設定ファイルについて説明します。

- ✦ IP アドレスが 192.168.0.101、192.168.0.102、192.168.0.103 のホスト 3 台
- ✦ 追加のパッケージやリポジトリはなし
- ✦ 全ボリュームに Arbiter 設定
- ✦ ストライプサイズが 256 KB の RAID 6 で設定されている 12 のブリック

これにより、以下のセクションが含まれた gdeploy の設定ファイルが生成されます。

本項で定義しているセクションについての情報は、Red Hat Gluster Storage 『**Administration Guide**』を参照してください:[https://access.redhat.com/documentation/en-us/red\\_hat\\_gluster\\_storage/3.2/html/administration\\_guide/chap-red\\_hat\\_storage\\_volumes#chap-Red\\_Hat\\_Storage\\_Volumes-gdeploy\\_configfile](https://access.redhat.com/documentation/en-us/red_hat_gluster_storage/3.2/html/administration_guide/chap-red_hat_storage_volumes#chap-Red_Hat_Storage_Volumes-gdeploy_configfile)

### [hosts] section

```
[hosts]
192.168.0.101
192.168.0.102
192.168.0.103
```

[hosts] セクションでは、この設定ファイルに従って設定される3台の物理マシンの IP アドレスを定義します。

### [script1] section

```
[script1]
action=execute
ignore_script_errors=no
file=/usr/share/ansible/gdeploy/scripts/grafton-sanity-check.sh -d sdb
-h 192.168.0.101,192.168.0.102,192.168.0.103
```

[script1] セクションでは、gdeploy がエラーなしで実行できるように、全ホストが正しく設定されていることを確認するために実行するスクリプトを指定します。

### 下層のストレージ設定

```
[disktype]
raid6

[diskcount]
```

12

```
[stripesize]
256
```

**[disktype]** セクションでは、全ホストの下層のストレージのハードウェア設定を指定します。

**[diskcount]** セクションでは、RAID ストレージ内のディスク数を指定します。JBOD 構成の場合には、この設定は省略することができます。

**[stripesize]** セクションでは、RAID ストレージのストライプサイズをキロバイト単位で指定します。JBOD 構成の場合には、この設定は省略することができます。

### NTPD の有効化と再起動

```
[service1]
action=enable
service=ntpd

[service2]
action=restart
service=ntpd
```

これらのサービスのセクションでは、全ホストで Network Time Protocol デーモン (NTPD) を有効化し、再起動します。

### 全ホスト上での物理ボリューム作成

```
[pv1]
action=create
devices=sdb
ignore_pv_errors=no
```

**[pv1]** セクションでは、全ホストの **sdb** デバイスに物理ボリュームを作成します。

### 全ホスト上でのボリュームグループ作成

```
[vg1]
action=create
vgname=gluster_vg_sdb
pvname=sdb
ignore_vg_errors=no
```

**[vg1]** セクションでは、前のステップで全ホスト上に作成した物理ボリュームにボリュームグループを作成します。

### 論理ボリュームのシンプールの作成

```
[lv1:{192.168.0.101,192.168.0.102}]
action=create
poolname=gluster_thinpool_sdb
ignore_lv_errors=no
```

```

vgname=gluster_vg_sdb
lvtype=thinpool
poolmetadatasize=16GB
size=1000GB

[lv2:192.168.0.103]
action=create
poolname=gluster_thinpool_sdb
ignore_lv_errors=no
vgname=gluster_vg_sdb
lvtype=thinpool
poolmetadatasize=16GB
size=20GB

```

**[lv1:\*]** セクションでは、最初の 2 つのホストに、メタデータプールサイズを 16 GB に指定して、1000 GB のシンプールを作成します。

**[lv2:\*]** セクションでは、3 番目のホストにメタデータプールサイズを 16 GB に指定して、20 GB のシンプールを作成します。これは、Arbiter ブリックに使用する論理ボリュームです。

**chunksize** の変数も利用可能ですが、使用する場合には注意が必要です。**chunksize** はスナップショット、キャッシュプール、シンプールに使用されるチャンクサイズを定義します。この値はデフォルトではキロバイト単位で指定されます。RAID 5 および 6 ボリュームの場合には、**gdeploy** がストライプサイズとディスク数を乗算してデフォルトのチャンクサイズを計算します。RAID 10 の場合には、デフォルトのチャンクサイズは 256 KB です。

### 警告

Red Hat では、少なくともデフォルトのチャンクサイズを使用することを推奨します。チャンクサイズが小さすぎてメタデータ用のボリューム容量が足りなくなった場合には、そのボリュームはデータを作成できなくなります。Red Hat は、メタデータボリュームが完全に満杯になる前に論理ボリュームを拡張するかストレージを追加できるように論理ボリュームを監視することをお勧めします。

## 下層の engine ストレージの作成

```

[lv3:{192.168.0.101,192.168.0.102}]
action=create
lvname=gluster_lv_engine
ignore_lv_errors=no
vgname=gluster_vg_sdb
mount=/gluster_bricks/engine
size=100GB
lvtype=thick

[lv4:192.168.0.103]
action=create
lvname=gluster_lv_engine
ignore_lv_errors=no
vgname=gluster_vg_sdb
mount=/gluster_bricks/engine
size=10GB
lvtype=thick

```

**[lv3:\*]** セクションでは、最初の 2 つのホスト上に **gluster\_lv\_engine** という名前の 100 GB のシックプロビジョニング論理ボリュームを作成します。このボリュームは、**/gluster\_bricks/engine** にマウントするように設定されます。

**[lv4:\*]** セクションでは、3 番目のホストに engine 用の 10 GB のシックプロビジョニング論理ボリュームを作成します。このボリュームは **/gluster\_bricks/engine** にマウントするように設定されています。

## 下層のデータおよび仮想マシンのブートディスク用ストレージ

```
[lv5:{192.168.0.101,192.168.0.102}]
action=create
lvname=gluster_lv_data
ignore_lv_errors=no
vgname=gluster_vg_sdb
mount=/gluster_bricks/data
lvtype=thinlv
poolname=gluster_thinpool_sdb
virtualsize=500GB
```

```
[lv6:192.168.0.103]
action=create
lvname=gluster_lv_data
ignore_lv_errors=no
vgname=gluster_vg_sdb
mount=/gluster_bricks/data
lvtype=thinlv
poolname=gluster_thinpool_sdb
virtualsize=10GB
```

```
[lv7:{192.168.0.101,192.168.0.102}]
action=create
lvname=gluster_lv_vmstore
ignore_lv_errors=no
vgname=gluster_vg_sdb
mount=/gluster_bricks/vmstore
lvtype=thinlv
poolname=gluster_thinpool_sdb
virtualsize=500GB
```

```
[lv8:192.168.0.103]
action=create
lvname=gluster_lv_vmstore
ignore_lv_errors=no
vgname=gluster_vg_sdb
mount=/gluster_bricks/vmstore
lvtype=thinlv
poolname=gluster_thinpool_sdb
virtualsize=10GB
```

**[lv5:\*]** および **[lv7:\*]** のセクションでは、最初の 2 つのホストのデータおよび vmstore ボリューム用に 500 GB の論理ボリュームをブリックとして作成します。

**[lv6:\*]** および **[lv8:\*]** のセクションでは、3 番目のホストのデータおよび vmstore ボリューム用に 10 GB の論理ボリュームを Arbiter ブリックとして作成します。

データブリックは `/gluster_bricks/data` に、vmstore ブリックは `/gluster_bricks/vmstore` にマウントされるように設定されます。

## SELinux ファイルシステムラベルの設定

```
[selinux]
yes
```

**[selinux]** のセクションでは、作成されるストレージが Gluster Storage 用に適切な SELinux ファイルシステムラベルで設定される必要があることを指定しています。

## glusterd の起動

```
[service3]
action=start
service=glusterd
slice_setup=yes
```

**[service3]** のセクションでは、glusterd サービスが起動して、コントロールグループを設定し、glusterd が全システムリソースを使い切ることができないようにします。詳しくは、Red Hat Enterprise Linux 『リソース管理ガイド』を参照してください:

[https://access.redhat.com/documentation/ja-JP/Red\\_Hat\\_Enterprise\\_Linux/7/html/Resource\\_Management\\_Guide/index.html](https://access.redhat.com/documentation/ja-JP/Red_Hat_Enterprise_Linux/7/html/Resource_Management_Guide/index.html)

## ファイアウォールの設定

```
[firewalld]
action=add
ports=111/tcp,2049/tcp,54321/tcp,5900/tcp,5900-6923/tcp,5666/tcp,16514/tcp
services=glusterfs
```

**[firewalld]** のセクションでは、Gluster のトラフィックを許可するのに必要なポートを開放します。

## Gluster フックの無効化

```
[script2]
action=execute
file=/usr/share/ansible/gdeploy/scripts/disable-gluster-hooks.sh
```

**[script2]** セクションでは、ハイパーコンバージドインフラストラクチャーに干渉する可能性のある Gluster フックを無効化します。

## Gluster ボリュームの作成

```
[volume1]
action=create
volname=engine
transport=tcp
replica=yes
```

```
replica_count=3
key=group,storage.owner-uid,storage.owner-gid,network.ping-
timeout,performance.strict-o-direct,network.remote-
dio,cluster.granular-entry-heal
value=virt,36,36,30,on,off,enable
brick_dirs=192.168.0.101:/gluster_bricks/engine/engine,192.168.0.102:/g
luster_bricks/engine/engine,192.168.0.103:/gluster_bricks/engine/engine
ignore_volume_errors=no
arbiter_count=1

[volume2]
action=create
volname=data
transport=tcp
replica=yes
replica_count=3
key=group,storage.owner-uid,storage.owner-gid,network.ping-
timeout,performance.strict-o-direct,network.remote-
dio,cluster.granular-entry-heal
value=virt,36,36,30,on,off,enable
brick_dirs=192.168.0.101:/gluster_bricks/data/data,192.168.0.102:/glust
er_bricks/data/data,192.168.0.103:/gluster_bricks/data/data
ignore_volume_errors=no
arbiter_count=1

[volume3]
action=create
volname=vmstore
transport=tcp
replica=yes
replica_count=3
key=group,storage.owner-uid,storage.owner-gid,network.ping-
timeout,performance.strict-o-direct,network.remote-
dio,cluster.granular-entry-heal
value=virt,36,36,30,on,off,enable
brick_dirs=192.168.0.101:/gluster_bricks/vmstore/vmstore,192.168.0.102:/
gluster_bricks/vmstore/vmstore,192.168.0.103:/gluster_bricks/vmstore/v
mstore
ignore_volume_errors=no
arbiter_count=1
```

[**volume\***] セクションでは、**engine**、**data**、**vmstore** の 3 つの Arbitrated Replicated の Red Hat Gluster Storage ボリュームを設定します。3 番目のホスト上の各ボリュームに Arbiter ブリックが 1 つずつ含まれます。

**key** と **value** のパラメーターは、以下のオプションの設定に使用されます。

- ※ **group=virt**
- ※ **storage.owner-uid=36**
- ※ **storage.owner-gid=36**
- ※ **network.ping-timeout=30**
- ※ **performance.strict-o-direct=on**

✘ `network.remote-dio=off`

✘ `cluster.granular-entry-heal=enable`