



# Red Hat OpenStack Platform 13

## Fujitsu ETERNUS 后端指南

在 Red Hat OpenStack Platform 环境中使用 Fujitsu ETERNUS 后端指南



# Red Hat OpenStack Platform 13 Fujitsu ETERNUS 后端指南

---

在 Red Hat OpenStack Platform 环境中使用 Fujitsu ETERNUS 后端指南

OpenStack Team  
rhos-docs@redhat.com

## 法律通告

Copyright © 2023 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, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

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

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

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

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

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

The OpenStack<sup>®</sup> 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.

## 摘要

本文档论述了如何将 Red Hat OpenStack Platform 13 配置为使用 Fujitsu ETERNUS Disk Storage System 作为后端。

---

## 目录

使开源包含更多 .....	3
第1章 简介 .....	4
第2章 配置 FUJITSU ETERNUS 设备 .....	5
第3章 准备 FUJITSU ETERNUS HEAT 模板 .....	7
3.1. 为每个 FUJITSU ETERNUS 后端创建驱动程序定义 .....	8
3.2. FUJITSU ETERNUS HEAT 模板示例 .....	10
第4章 创建 FUJITSU ETERNUS 环境文件 .....	13
第5章 部署配置的 FUJITSU ETERNUS 后端 .....	16
第6章 测试您的 FUJITSU ETERNUS 配置 .....	17



## 使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。有关更多详情，请参阅[我们的首席技术官 Chris Wright 提供的消息](#)。

# 第1章 简介

本文档论述了如何将 Red Hat OpenStack Platform 配置为使用 Fujitsu ETERNUS Disk Storage System 作为块存储服务的后端。本文档论述了如何定义 overcloud 部署上 ETERNUS 设备提供的光纤通道和 iSCSI 后端。此过程涉及将两个后端定义为块存储服务的**自定义后端**。默认情况下，Controller 节点包含块存储服务。

## 前提条件

- 通过带有容器 [registry.connect.redhat.com/fujitsu/rhosp13-fujitsu-cinder-volume-161](https://registry.connect.redhat.com/fujitsu/rhosp13-fujitsu-cinder-volume-161) 的 director 部署的 Red Hat OpenStack Platform overcloud。如需更多信息，请参阅 [部署供应商插件](#)。
- 您要只使用 Fujitsu ETERNUS Disk Storage System devices 和 Block Storage 后端的驱动程序。
- 您可以使用 *director 安装用户*，通过 overcloud 部署创建。有关更多信息，请参阅从 [Director 安装和使用](#) [创建 Stack 用户](#)。
- 您可以通过 ETERNUS Web GUI 或 CLI 访问 ETERNUS 设备上的管理员帐户。

红帽支持使用光纤通道或 iSCSI 接口以及相应的驱动程序和设置，并带有 Fujitsu ETERNUS 设备。



### 注意

有关定义自定义后端的更多信息，[请参阅自定义块存储后端部署指南](#)。



## 第 2 章 配置 FUJITSU ETERNUS 设备

在将 Fujitsu ETERNUS 设备定义为块存储后端前，在设备上配置存储池和端口。有关每个步骤的详情，请参考您的设备文档：

### 流程

1. 配置托管块存储服务的 Controller 节点和 ETERNUS 设备的 MNT 端口的 LAN 连接。
2. 在 ETERNUS 设备的 Compute 节点和 CA 端口之间配置 SAN 连接。
3. 使用具有 Admin 角色的帐户登录 ETERNUS 设备。
4. 启用 ETERNUS DX 的 SMI-S。
5. 设置 SSH 密钥
6. 使用软件角色添加用户帐户。
7. 注册高级复制功能许可证并配置复制表大小。
8. 为卷创建存储池。您稍后在 [第 3.1 节“为每个 Fujitsu ETERNUS 后端创建驱动程序定义”](#) 中的 EternusPool 设置中使用这个池。
9. 可选：为卷快照创建单独的存储池。这个池代表 [第 3.1 节“为每个 Fujitsu ETERNUS 后端创建驱动程序定义”](#) 中的 EternusSnapPool 设置。
10. 可选：创建一个 Snap 数据仓库卷(SDPV)，为创建快照功能启用 Snap 数据仓库(SDP)。

### 提示

如果将精简调配池用于 EternusSnapPool，请跳过这一步。

11. 配置块存储服务要使用的 存储端口。
12. 将存储端口端口 设置为 CA 模式。
13. 要为存储端口启用 主机反关联性，请在每个端口的 ETERNUS CLI 中输入以下命令：

```
set <PROTO>-parameters -host-affinity enable -port <CM#> <CA#> <PORT>
```

- 将 <PROTO> 替换为存储协议，如 fc 或 iscsi。
- 将 <CM#> 和 <CA#> 替换为端口所在控制器外方的名称。
- 将 <PORT> 替换为端口号。

14. 使用以下命令在 undercloud 上生成 SSH 密钥，并将 eternus.ietf 文件上传到 ETERNUS 设备。

```
ssh-keygen -t rsa -N "" -f ./eternus  
ssh-keygen -e -f ./eternus.pub > ./eternus.ietf
```



#### 注意

保存 eternus 文件供以后使用。为安全起见，请不要复制或将其从 undercloud 或 overcloud 中移出。

### 第 3 章 准备 FUJITSU ETERNUS HEAT 模板

为确保以后在 Red Hat OpenStack Platform overcloud 更新过程中保留您的设置，请通过 **director** 在部署期间执行所有服务配置。

在托管块存储服务的 ETERNUS 后端的 Controller 节点上包括以下配置：

- 您可以为每个后端驱动程序设置包含 XML 配置文件。
- 包括一个与 ETERNUS 设备通信的 SSH 服务器密钥。

您可以使用 heat 模板通过 **director** 编配这两个任务。有关 **director** heat 模板语法的更多信息，请参阅高级 *Overcloud 指南* 中的 [了解 Heat 模板](#)。

以下模板 `eternus-temp.yaml` 概述了所需 heat 模板的基本语法。

#### eternus-temp.yaml

```
heat_template_version: 2014-10-16

description: >
  Add XML configuration file for the driver settings of each back end

parameters:
  server:
    type: string

resources:
  EternusSetup: # 1
    type: OS::Heat::SoftwareConfig
    properties:
      group: script
      config: | # 2
        #!/bin/bash
        # 3

  ExtraPreDeployment:
    type: OS::Heat::SoftwareDeployment
    properties:
```

```
config: {get_resource: EternusSetup}
server: {get_param: server}
actions: ['CREATE','UPDATE']
```

**1**

**EternusSetup** 部分包含编配 **Controller** 节点上的任务的资源。

**2**

**config** 部分包含在 **Controller** 节点上运行的命令。

**3**

将私钥信息复制到托管块存储服务的每个 **Controller** 节点，并添加命令为 [第 3.1 节 “为每个 Fujitsu ETERNUS 后端创建驱动程序定义”](#) 中每个后端驱动程序设置创建 XML 配置文件的命令。

将此文件存储在 **director** 节点上的自定义 **heat** 模板目录中，即 `/home/stack/templates/`。

### 3.1. 为每个 FUJITSU ETERNUS 后端创建驱动程序定义

在单独的 XML 文件上为每个 **ETERNUS** 后端定义驱动程序设置，而不是块存储配置文件 `/etc/cinder/cinder.conf`。确定每个后端都有一个 XML 文件，但有以下设置：

#### EternusIP

**ETERNUS** 设备的 **SMI-S** 连接的 IP 地址。使用设备 **MNT** 端口的 IP 地址。

#### EternusPort

**ETERNUS** 设备的 **SMI-S** 连接端口的端口号。

#### EternusUser

连接 **EternusIP** 的软件角色的用户名。

#### EternusPassword

**EternusUser** 的对应密码( **EternusIP** )。

#### EternusPool

从第 2 章 [配置 Fujitsu ETERNUS 设备](#) 中卷的存储池的名称。使用 ETERNUS 设备中的池 RAID 组名称或 TPP 名称。

### EternusSnapPool

从第 2 章 [配置 Fujitsu ETERNUS 设备](#) 的卷快照的存储池名称。使用 ETERNUS 设备中的池 RAID 组名称。如果没有为快照创建不同的池，请使用与 EternusPool 相同的值。

使用以下 xml 示例定义 Fibre Channel 配置：

### eternus-fc.xml

```
<?xml version='1.0' encoding='UTF-8'?>
<FUJITSU>
<EternusIP>0.0.0.0</EternusIP>
<EternusPort>5988</EternusPort>
<EternusUser>smisuser</EternusUser>
<EternusPassword>smispassword</EternusPassword>
<EternusPool>raid5_0001</EternusPool>
<EternusSnapPool>raid5_0001</EternusSnapPool>
</FUJITSU>
```

使用以下 xml 示例定义 ISCSI 配置：

### eternus-iscsi.xml

```
<?xml version='1.0' encoding='UTF-8'?>
<FUJITSU>
<EternusIP>0.0.0.0</EternusIP>
<EternusPort>5988</EternusPort>
<EternusUser>smisuser</EternusUser>
<EternusPassword>smispassword</EternusPassword>
<EternusPool>raid5_0001</EternusPool>
<EternusSnapPool>raid5_0001</EternusSnapPool>
</FUJITSU>
```

要编排这些 XML 文件的创建，请在来自 [第 3 章 准备 Fujitsu ETERNUS heat 模板的 /home/stack/templates/eternus-temp.yaml](#) 文件中的 EternusSetup 资源的 config 部分包含 bash 命令。使用以下命令编配 eternus-fc.xml 和 eternus-iscsi.xml 的创建：

```
sudo cat > /etc/cinder/eternus-fc.xml <<EOF
<?xml version='1.0' encoding='UTF-8'?>
<FUJITSU>
<EternusIP>0.0.0.0</EternusIP>
<EternusPort>5988</EternusPort>
<EternusUser>smisuser</EternusUser>
<EternusPassword>smispassword</EternusPassword>
<EternusPool>raid5_0001</EternusPool>
<EternusSnapPool>raid5_0001</EternusSnapPool>
</FUJITSU>
EOF

sudo cat > /etc/cinder/eternus-iscsi.xml <<EOF
<?xml version='1.0' encoding='UTF-8'?>
<FUJITSU>
<EternusIP>0.0.0.0</EternusIP>
<EternusPort>5988</EternusPort>
<EternusUser>smisuser</EternusUser>
<EternusPassword>smispassword</EternusPassword>
<EternusPool>raid5_0001</EternusPool>
<EternusSnapPool>raid5_0001</EternusSnapPool>
</FUJITSU>
EOF
```

使用 `sudo cat` 命令创建所需的 XML 配置文件。

为 `cinder` 用户和组设置这些 XML 文件的所有权和权限。

```
sudo chown cinder:cinder /etc/cinder/eternus-*.xml
sudo chmod 0600 /etc/cinder/eternus-*.xml
```

有关完成的 heat 模板示例，请参考 [第 3.2 节 “Fujitsu ETERNUS heat 模板示例”](#)。

### 3.2. FUJITSU ETERNUS HEAT 模板示例

以下 `/home/stack/templates/eternus-temp.yaml` 文件包含声明示例 XML 配置文件所需的参数，如 `eternus-fc.xml` 和 `eternus-iscsi.xml`：

`/home/stack/templates/eternus-temp.yaml`

heat\_template\_version: 2014-10-16

description: >

Add XML configuration file for the driver settings of each back end

parameters:

server:

type: string

resources:

EternusSetup:

type: OS::Heat::SoftwareConfig

properties:

group: script

config: |

```
#!/bin/bash
```

```
sudo cat > /etc/cinder/eternus-fc.xml <<EOF
```

```
<?xml version='1.0' encoding='UTF-8'?>
```

```
<FUJITSU>
```

```
<EternusIP>0.0.0.0</EternusIP>
```

```
<EternusPort>5988</EternusPort>
```

```
<EternusUser>smisuser</EternusUser>
```

```
<EternusPassword>smispassword</EternusPassword>
```

```
<EternusPool>raid5_0001</EternusPool>
```

```
<EternusSnapPool>raid5_0001</EternusSnapPool>
```

```
</FUJITSU>
```

```
EOF
```

```
sudo cat > /etc/cinder/eternus-iscsi.xml <<EOF
```

```
<?xml version='1.0' encoding='UTF-8'?>
```

```
<FUJITSU>
```

```
<EternusIP>0.0.0.0</EternusIP>
```

```
<EternusPort>5988</EternusPort>
```

```
<EternusUser>smisuser</EternusUser>
```

```
<EternusPassword>smispassword</EternusPassword>
```

```
<EternusPool>raid5_0001</EternusPool>
```

```
<EternusSnapPool>raid5_0001</EternusSnapPool>
```

```
</FUJITSU>
```

```
EOF
```

```
sudo cat > /etc/cinder/eternus <<EOF
```

```
-----BEGIN RSA PRIVATE KEY----- # 1
```

```
MIIEpAIBAACAQEA5yMqonpfniu+11PJ8qdWZpcf0d4UcHj2uyE7ou7vcZUQ1Cq
```

```
s5Q5pjkCgYAxITlpfOYA8jvLgc7vMEa/ZbhUgAPIYIisxbffmRsBWYJSt9gwHpcW
```

```
hvaWo6VD/iUKZ3bOcmK0buUwBdFUt5s9B8mXbYsX6bWovIVkyu8DzQfpDiPnV6C8
```

```
...
```

```
IB+46ldmCUO0DaciuEz5/KQd4AXBNdTOss2od6OzihDJXKjBwPyP1g==
```

```
-----END RSA PRIVATE KEY-----
```

```
EOF
```

ExtraPreDeployment:

type: OS::Heat::SoftwareDeployment

properties:

```
config: {get_resource: EternusSetup}
server: {get_param: server}
actions: [CREATE,UPDATE]
```

1

**undercloud** 中生成的私钥 信息 [第 2 章 配置 Fujitsu ETERNUS 设备](#)。



## 第 4 章 创建 FUJITSU ETERNUS 环境文件

环境文件包含您要定义的后端的设置，以及其他相关设置。有关环境文件的更多信息，[请参阅高级 Overcloud 自定义指南中的环境文件](#)。

另外，环境文件会注册您在 [第 3 章 准备 Fujitsu ETERNUS heat 模板](#) 中创建的 heat 模板。heat 模板中定义的安装和回显命令在部署期间在适当的节点上运行。

以下示例环境文件包含定义 ETERNUS 设备作为块存储后端所需的部分。它还会为 [第 3.1 节 “为每个 Fujitsu ETERNUS 后端创建驱动程序定义”](#) 和 [第 3.2 节 “Fujitsu ETERNUS heat 模板示例”](#) 中编排的每个对应 XML 文件创建后端定义。

## eternusbackend-env.yaml

```
resource_registry:
  OS::TripleO::NodeExtraConfig: /home/stack/templates/eternus-temp.yaml # 1

parameter_defaults: # 2
  CinderEnableIscsiBackend: false
  CinderEnableRbdBackend: false
  CinderEnableNfsBackend: false
  NovaEnableRbdBackend: false
  GlanceBackend: file # 3
  controllerExtraConfig: # 4
    cinder::config::cinder_config:
      FJFC/volume_driver: # 5
        value: cinder.volume.drivers.fujitsu.eternus_dx.eternus_dx_fc.FJDXFCDriver
      FJFC/cinder_eternus_config_file: # 6
        value: /etc/cinder/eternus-fc.xml
      FJFC/volume_backend_name: # 7
        value: FJFC
      FJFC/fujitsu_private_key_path:
        value: /etc/cinder/eternus
      FJISCSI/volume_driver: # 8
        value: cinder.volume.drivers.fujitsu.eternus_dx.eternus_dx_iscsi.FJDXISCSIDriver
      FJISCSI/cinder_eternus_config_file:
        value: /etc/cinder/eternus-iscsi.xml
      FJISCSI/volume_backend_name:
        value: FJISCSI
      FJISCSI/fujitsu_private_key_path:
        value: /etc/cinder/eternus
    cinder_user_enabled_backends: ['FJFC','FJISCSI'] # 9
  CinderVolumeOptVolumes: # 10
```

```
- /etc/cinder/eternus-iscsi.xml:/etc/cinder/eternus-iscsi.xml:ro  
- /etc/cinder/eternus-fc.xml:/etc/cinder/eternus-fc.xml:ro  
- /etc/cinder/eternus:/etc/cinder/eternus:ro
```

**1**

在使用 **NodeExtraConfig** 的核心 **Puppet** 配置前，为所有节点定义自定义设置。这样可确保在 **overcloud** 上部署块存储服务时，确保以下配置：

- 每个后端的 XML 配置文件都存在。
- 生成私钥。

**2**

将以下参数设置为 **false** 来禁用其他后端类型：

- **CinderEnableIscsiBackend** : 其他 iSCSI 后端。
- **CinderEnableRbdBackend**: Red Hat Ceph Storage.
- **CinderEnableNfsBackend**: NFS.
- **NovaEnableRbdBackend**: ephemeral Red Hat Ceph Storage.

**3**

使用 **GlanceBackend** 参数定义镜像服务镜像存储设置。支持以下值：

- 文件 在每个 **Controller** 节点上的 **/var/lib/glance/images** 上存储镜像。
- **Swift** 使用对象存储服务进行镜像存储。

- Cinder 使用块存储服务进行镜像存储。

**4**

使用 `controllerExtraConfig` 为所有 Controller 节点定义自定义设置。 `cinder::config::cinder_config` 类用于块存储服务。 `director` 在每个节点的 `/etc/cinder/cinder.conf` 文件中存储这些后端设置。

**5**

使用 `FJFC/` 字符串配置名为 `FJFC/` 字符串的后端定义，并在该后端定义下声明 `volume_driver` 参数。使用 `volume_driver` 参数为后端设置 Fibre Channel ETERNUS 驱动程序，如 `cinder.volume.drivers.fujitsu.eternus_dx.eternus_dx_fc.FJDXFCDriver`。

**6**

使用 `cinder_eternus_config_file` 设置驱动程序用于后端的 XML 配置文件的路径。通过 `heat` 模板编排 `/etc/cinder/eternus-fc.xml` 创建，如 `/home/stack/templates/eternus-temp.yaml`。

**7**

`volume_backend_name` 是块存储服务用来启用后端的名称。

**8**

使用 `FJISCSI/` 字符串配置新后端定义。使用 `volume_driver` 参数为后端设置 iSCSI ETERNUS 驱动程序，如 `cinder.volume.drivers.fujitsu.eternus_dx.eternus_dx_iscsi.FJDXISCSIDriver`。

**9**

使用 `cinder_user_enabled_backends` 类设置并启用自定义后端。仅将这个类用于支持用户的后端，如 `cinder::config::cinder_config` 类中定义的后端。

**10**

使主机上的自定义配置文件可供使用 `Cinder-volume` 服务在 `CinderVolumeOptVolumes` 容器中运行。

创建环境文件后，您可以部署配置。有关环境文件 `/home/stack/templates/eternusbackend-env.yaml` 的详情请参考 [第 5 章 部署配置的 Fujitsu ETERNUS 后端](#)。

## 第 5 章 部署配置的 FUJITSU ETERNUS 后端

在 `/home/stack/templates/` 中创建 `eternusbackend-env.yaml` 文件后，完成以下步骤：

### 流程

1. 以 `stack` 用户身份登录。
2. 使用以下命令部署后端配置：

```
$ openstack overcloud deploy --templates \  
-e [your environment files] \  
-e /home/stack/templates/eternusbackend-env.yaml
```



### 重要

如果您在创建 `overcloud` 时传递任何额外的环境文件，请使用 `-e` 选项再次传递它们，以避免对 `overcloud` 进行不必要的更改。有关更多信息，请参阅 *Director 安装和使用指南* 中的 [修改 Overcloud 环境](#)。

在 `director` 编排完成后测试后端。请参阅 [第 6 章 测试您的 Fujitsu ETERNUS 配置](#)。

## 第 6 章 测试您的 FUJITSU ETERNUS 配置

将块存储服务配置为使用新的 ETERNUS 后端后，为每个后端声明卷类型。使用卷类型在创建新卷时指定要使用的后端。

- 创建一个 Fibre Channel 后端，使用以下命令将其映射到对应的后端：

```
# cinder type-create FJFC  
# cinder type-key FJFC set volume_backend_name=FJFC
```

- 创建一个 iSCSI 后端，使用以下命令将其映射到对应的后端：

```
# cinder type-create FJISCSI  
# cinder type-key FJISCSI volume_backend_name=FJISCSI
```

有关卷类型的更多信息，请参阅 [第 4 章 创建 Fujitsu ETERNUS 环境文件](#)：

- 创建一个名为 `test_iscsi` 的 1GB iSCSI 卷，以验证您的配置：

```
# cinder create --volume_type FJISCSI --display_name test_iscsi 1
```

- 测试光纤通道后端：

```
# cinder create --volume_type FJFC --display_name test_fc 1
```