



Red Hat Ceph Storage 7

安装指南

在 Red Hat Enterprise Linux 上安装 Red Hat Ceph Storage

Red Hat Ceph Storage 7 安装指南

在 Red Hat Enterprise Linux 上安装 Red Hat Ceph Storage

法律通告

Copyright © 2024 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[®] 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 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.

摘要

本文档提供有关在 AMD64 和 Intel 64 架构上运行的 Red Hat Enterprise Linux 上安装 Red Hat Ceph Storage 的说明。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。详情请查看 CTO Chris Wright 信息。

目录

第 1 章 RED HAT CEPH STORAGE	3
第 2 章 RED HAT CEPH STORAGE 注意事项和建议	4
2.1. RED HAT CEPH STORAGE 的基本注意事项	4
2.2. RED HAT CEPH STORAGE 工作负载注意事项	5
2.3. RED HAT CEPH STORAGE 的网络注意事项	8
2.4. 在 OSD 主机中使用 RAID 控制器的注意事项	9
2.5. 在运行 CEPH 时调整 LINUX 内核的注意事项	9
2.6. COLOCATION 如何工作及其优点	9
2.7. RED HAT CEPH STORAGE 的操作系统要求	15
2.8. RED HAT CEPH STORAGE 的最低硬件注意事项	16
第 3 章 RED HAT CEPH STORAGE 安装	19
3.1. CEPHADM 实用程序	19
3.2. CEPHADM 的工作原理	20
3.3. CEPHADM-ANSIBLE PLAYBOOK	20
3.4. 将 RED HAT CEPH STORAGE 节点注册到 CDN 并附加订阅	21
3.5. 配置 ANSIBLE 清单位置	23
3.6. 在 RED HAT ENTERPRISE LINUX 9 中以 ROOT 用户身份启用 SSH 登录	24
3.7. 创建具有 SUDO 访问权限的 ANSIBLE 用户	25
3.8. 为 ANSIBLE 启用免密码 SSH	27
3.9. 运行 PREFLIGHT PLAYBOOK	28
3.10. 引导新存储集群	30
3.11. 分发 SSH 密钥	51
3.12. 启动 CEPHADM SHELL	52
3.13. 验证集群安装	53
3.14. 添加主机	54
3.15. 标记主机	61
3.16. 添加 MONITOR 服务	65
3.17. 在现有集群中设置自定义 SSH 密钥	67
3.18. 设置管理节点	69
3.19. 添加管理器服务	72
3.20. 添加 OSD	73
3.21. 运行 CEPHADM-CLIENTS PLAYBOOK	74
3.22. 清除 CEPH 存储集群	75
3.23. 部署客户端节点	76
第 4 章 使用 CEPHADM-ANSIBLE 模块管理红帽 CEPH 存储集群	79
4.1. CEPHADM-ANSIBLE 模块	79
4.2. CEPHADM-ANSIBLE 模块选项	79
4.3. 使用 CEPHADM_BOOTSTRAP 和 CEPHADM_REGISTRY_LOGIN 模块引导存储集群	83
4.4. 使用 CEPH_ORCH_HOST 模块添加或删除主机	85
4.5. 使用 CEPH_CONFIG 模块设置配置选项	90
4.6. 使用 CEPH_ORCH_APPLY 模块应用服务规格	92
4.7. 使用 CEPH_ORCH_DAEMON 模块管理 CEPH 守护进程状态	94
第 5 章 接下来该怎么办？第 2 天	96
附录 A. CEPH ANSIBLE 和 CEPHADM 之间的比较	97
附录 B. CEPHADM 命令	99

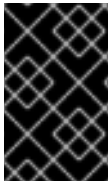
第 1 章 RED HAT CEPH STORAGE

Red Hat Ceph Storage 是一款可扩展、开放、软件定义型存储平台，结合了 Ceph 存储系统的企业级强化版本和 Ceph 管理平台、部署实用程序和支持服务。

Red Hat Ceph Storage 存储专为云基础架构和 Web 规模对象存储而设计。Red Hat Ceph Storage 集群由以下类型的节点组成：

Ceph monitor

每一 Ceph 监控 (Monitor) 节点会运行 **ceph-mon** 守护进程，它会维护存储集群映射的一个主 (master) 副本。存储集群映射包含存储集群拓扑。连接 Ceph 存储集群的客户端从 Ceph monitor 检索存储集群映射的当前副本，这使得客户端能够从存储集群读取和写入数据。



重要

存储群集只能使用一个 Ceph monitor 运行；但是，为了确保在生产存储群集中实现高可用性，红帽将仅支持具有至少三个 Ceph 监控节点的部署。红帽建议为超过 750 个 Ceph OSD 的存储群集部署总计 5 个 Ceph 监控器。

Ceph Manager

Ceph 管理器 (Manager) 守护进程 **ceph-mgr** 与 Ceph 监控节点上运行的 Ceph monitor 守护进程共同存在，以提供额外的服务。Ceph 管理器利用 Ceph 管理器模块为其他监控和管理系统提供接口。运行 Ceph 管理器守护进程对于普通存储群集操作是必需的。

Ceph OSD

每个 Ceph 对象存储设备 (OSD) 节点运行 **ceph-osd** 守护进程，该守护进程与附加到节点的逻辑卷交互。存储群集在这些 Ceph OSD 节点上存储数据。

Ceph 可在只有很少 OSD 节点的环境中运行，默认为三个。但对于生产环境，只有从中等范围环境开始才可能看到其在性能方面的优势。例如，存储群集中的 50 个 Ceph OSD。理想情况下，Ceph 存储群集具有多个 OSD 节点，可以通过相应地配置 CRUSH map 来隔离故障域。

Ceph MDS

每个 Ceph 元数据服务器 (MDS) 节点运行 **ceph-mds** 守护进程，它管理与 Ceph 文件系统 (CephFS) 中存储的文件相关的元数据。Ceph MDS 守护进程也协调对共享存储群集的访问。

Ceph 对象网关

Ceph 对象网关节点运行 **ceph-radosgw** 守护进程，它是基于 **librados** 构建的对象存储接口，为应用提供 Ceph 存储群集的 RESTful 访问点。Ceph 对象网关支持两个接口：

- S3
通过与 Amazon S3 RESTful API 的大子集兼容的接口提供对象存储功能。
- Swift
通过与 OpenStack Swift API 的大集兼容的接口提供对象存储功能。

其它资源

- 有关 Ceph 架构的详细信息，请参阅 [Red Hat Ceph Storage 架构指南](#)。
- 有关最低硬件建议，请参阅 [Red Hat Ceph Storage Hardware Selection Guide](#)。

第 2 章 RED HAT CEPH STORAGE 注意事项和建议

作为存储管理员，您可以在运行 Red Hat Ceph Storage 集群对其有一定的了解。了解诸如硬件和网络要求等因素，了解哪种类型的工作负载与 Red Hat Ceph Storage 集群配合工作以及红帽的建议。Red Hat Ceph Storage 可根据特定业务需求或一组要求，用于不同的工作负载。在安装 Red Hat Ceph Storage 之前，进行必要的规划是高效运行 Ceph 存储集群以满足业务需求的关键。



注意

您是否想要获得针对特定用例规划 Red Hat Ceph Storage 集群的帮助？请联络您的红帽代表以获得帮助。

2.1. RED HAT CEPH STORAGE 的基本注意事项

使用 Red Hat Ceph Storage 的第一个考虑因素是为数据制定存储策略。存储策略是一种存储服务特定用例的数据的方法。如果您需要为 OpenStack 等云平台存储卷和镜像，可以选择将数据存储在带有 Solid State Drives (SSD) 的快速 Serial Attached SCSI (SAS) 驱动器上。相反，如果您需要存储 S3 或 Swift 兼容网关的对象数据，您可以选择使用更经济的方式，如传统的 SATA 驱动器。Red Hat Ceph Storage 可以在同一存储集群中同时容纳这两种场景，但您需要一种方式为云平台提供快速存储策略，并为对象存储提供更传统的存储方式。

一个成功的 Ceph 部署中的最重要的一个步骤是，找出一个适合存储集群的用例和工作负载的性价比配置集。为用例选择正确的硬件非常重要。例如，为冷存储应用程序选择 IOPS 优化的硬件会不必要地增加硬件成本。然而，在 IOPS 密集型工作负载中，选择容量优化的硬件使其更具吸引力的价格点可能会导致用户对性能较慢的抱怨。

Red Hat Ceph Storage 可以支持多种存储策略。用例、成本与好处性能权衡以及数据持久性是帮助开发合理存储策略的主要考虑因素。

使用案例

Ceph 提供大量存储容量，它支持许多用例，例如：

- Ceph 块设备客户端是云平台的领先存储后端，可为具有写时复制（copy-on-write）克隆等高性能功能的卷和镜像提供无限存储。
- Ceph 对象网关客户端是云平台的领先存储后端，为音频、位映射、视频和其他数据等对象提供 RESTful S3 兼容和 Swift 兼容对象存储。
- 传统文件存储的 Ceph 文件系统。

成本比较性能优势

越快越好。越大越好。越耐用越好。但是，每种出色的质量、相应的成本与收益权衡都有价格。从性能角度考虑以下用例：SSD 可以为相对较小的数据和日志量提供非常快速的存储。存储数据库或对象索引可以从非常快的 SSD 池中受益，但对于其他数据而言成本过高。带有 SSD 日志的 SAS 驱动器以经济的价格为卷和图像提供快速性能。没有 SSD 日志的 SATA 驱动器可提供低成本存储，同时整体性能也较低。在创建 OSD 的 CRUSH 层次结构时，您需要考虑用例和可接受的成本与性能权衡。

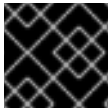
数据持续时间

在大型存储集群中，硬件故障是预期的，而非例外。但是，数据丢失和服务中断仍然不可接受。因此，数据的持久性非常重要。Ceph 通过对象的多个副本解决数据持久性问题，或使用纠删代码和多个编码区块来解决数据持久性。多个副本或多个编码区块会带来额外的成本与好处权衡：存储更少的副本或编码区块会更便宜，但可能会导致在降级状态中为写入请求提供服务。通常，一个具有两个额外副本的对象（或两个编码区块）可以允许存储集群在存储集群恢复时服务降级状态的写入。

在出现硬件故障时，复制存储在故障域中的一个或多个数据冗余副本。但是，冗余的数据副本规模可能会变得昂贵。例如，要存储 1 PB 字节并带有三倍复制的数据，将需要至少具有 3 PB 存储容量的集群。

纠删代码将数据存储为数据区块和编码区块。如果数据区块丢失，纠删代码可以使用剩余的数据区块和编码区块来恢复丢失的数据区块。纠删代码比复制更经济。例如，使用带有 8 个数据区块和 3 个编码区块的纠删代码提供与 3 个数据副本相同的冗余。但是，与复制相比（使用 3 倍的初始数据），此类编码方案使用约 1.5 倍的初始数据。

CRUSH 算法通过确保 Ceph 将额外的副本或编码区块存储在存储集群内的不同位置来协助这个过程。这样可以确保单个存储设备或主机的故障不会丢失防止数据丢失所需的所有副本或编码区块。您可以规划一个成本取舍存储策略，以及数据持久性，然后将它作为存储池呈现给 Ceph 客户端。



重要

数据存储池可以使用纠删代码。存储服务数据和存储桶索引的池使用复制。



重要

与 Ceph 的对象复制或编码区块相比，RAID 解决方案已变得过时。不要使用 RAID，因为 Ceph 已经处理数据持久性，降级的 RAID 对性能有负面影响，并且使用 RAID 恢复数据比使用深度副本或纠删代码区块要慢得多。

其它资源

- 如需了解更多详细信息，请参阅 [Red Hat Ceph Storage 安装指南中的 Red Hat Ceph Storage 的最低硬件注意事项](#) 部分。

2.2. RED HAT CEPH STORAGE 工作负载注意事项

Ceph 存储集群的一个关键优势在于能够使用性能域支持同一存储集群中的不同类型的工作负载。不同的硬件配置可以与每个性能域关联。存储管理员可以在适当的性能域中部署存储池，为应用提供专为特定性能和成本配置文件量身定制的存储。为这些性能域选择适当的大小和优化的服务器是设计 Red Hat Ceph Storage 集群的一个重要方面。

在读取和写入数据的 Ceph 客户端接口中，Ceph 存储集群显示为一个客户端存储数据的简单池。但是，存储集群以对客户端接口完全透明的方式执行许多复杂的操作。Ceph 客户端和 Ceph 对象存储守护进程（称为 Ceph OSD）或只是 OSD，都使用可扩展哈希下的受控复制 (CRUSH) 算法来存储和检索对象。Ceph OSD 可以在存储集群内的容器中运行。

CRUSH map 描述了集群资源的拓扑结构，并且 map 存在于客户端主机和集群中的 Ceph 监控主机中。Ceph 客户端和 Ceph OSD 都使用 CRUSH map 和 CRUSH 算法。Ceph 客户端直接与 OSD 通信，消除了集中式对象查找和潜在的性能瓶颈。利用 CRUSH map 并与其对等方通信，OSD 可以处理复制、回填和恢复，从而实现动态故障恢复。

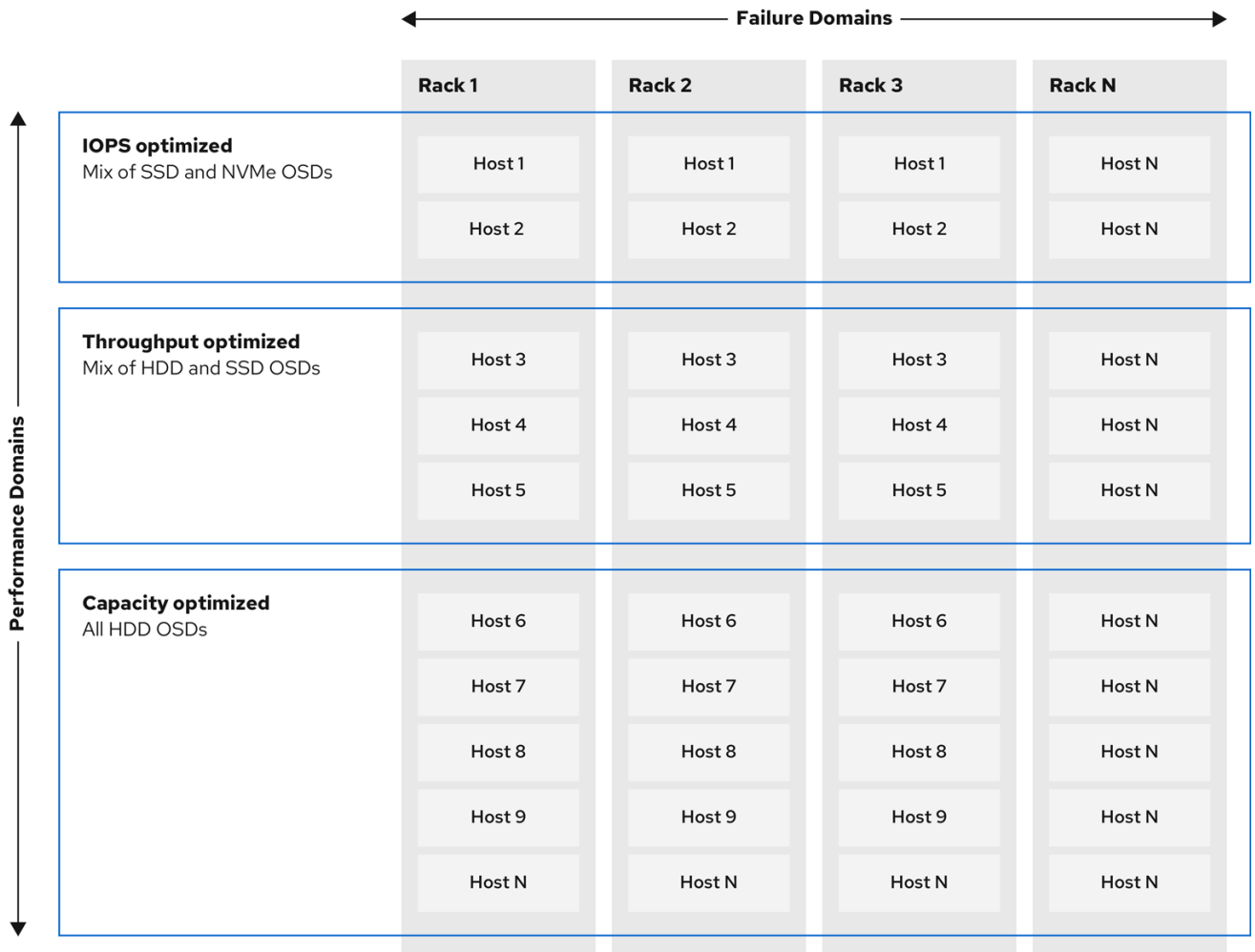
Ceph 使用 CRUSH map 来实施故障域。Ceph 还使用 CRUSH map 实施性能域，这只需将底层硬件的性能配置文件纳入考量。CRUSH map 描述了 Ceph 存储数据的方式，它作为简单的层次结构（特别是圆环图和规则集）实施。CRUSH map 可以支持多种层次结构，将一种类型的硬件性能配置集与另一类分隔开。Ceph 实施具有设备“类”的性能域。

例如，您可以让这些性能域共存在同一 Red Hat Ceph Storage 集群中：

- 硬盘 (HDD) 通常适合以成本和容量为导向的工作负载。
- 吞吐量敏感的工作负载通常使用 HDD，在固态硬盘 (SSD) 上 Ceph 写入日志。

- MySQL 和 MariaDB 等 IOPS 密集型工作负载通常使用 SSD。

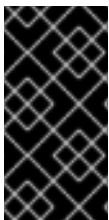
图 2.1. 性能和故障域



336_Ceph_0523

工作负载

Red Hat Ceph Storage 针对三种主要工作负载进行了优化：



重要

在购买硬件前，请仔细考虑由 Red Hat Ceph Storage 运行的工作负载，因为它可能会显著影响存储集群的价格和性能。例如，如果工作负载是容量优化的，并且硬件更适合通过吞吐量优化的工作负载，则硬件的成本将超过必要成本。相反，如果工作负载被优化吞吐量，且硬件更适合容量优化的工作负载，则存储集群的性能会受到影响。

- **优化 IOPS：** IOPS (Input, output per second) 优化部署适合云计算操作，例如将 MySQL 或 MariaDB 实例作为 OpenStack 上的虚拟机运行。优化 IOPS 部署需要更高的性能存储，如 15k RPM SAS 驱动器和单独的 SSD 日志，以处理频繁的写入操作。一些高 IOPS 情景使用所有闪存存储来提高 IOPS 和总吞吐量。
IOPS 优化存储集群具有以下属性：
 - 每个 IOPS 的成本最低。
 - 每 GB 的 IOPS 最高。

- 99 个百分点延迟一致性.

IOPS 优化存储集群的用例：

- 典型的块存储.
- 用于硬盘 (HDD) 或 2x 复制的 3 倍复制，用于固态硬盘 (SSD)。
- OpenStack 云上的 MySQL.
- **优化吞吐量：**使用优化吞吐量的部署适合服务大量数据，如图形、音频和视频内容。优化吞吐量的部署需要高带宽网络硬件、控制器和硬盘，具有快速顺序的读写特征。如果要求快速数据访问，则使用吞吐量优化存储策略。此外，如果要求快速写入性能，将 Solid State Disk (SSD) 用于日志将显著提高写入性能。

吞吐量优化存储集群具有以下属性：

- 每 MBps 成本最低（吞吐量）。
- 每个 TB 的 MBps 最高。
- 每个 BTU 的 MBps 最高。
- 每个 Watt 的 MBps 最高。
- 97% 的延迟一致性.

优化吞吐量的存储集群用例：

- 块或对象存储。
- 3 倍复制。
- 面向视频、音频和图像的主动性能存储.
- 流媒体，如 4k 视频.
- **优化容量：**容量优化部署适合以尽可能低的成本存储大量数据。容量优化的部署通常会以更具吸引力的价格点来换取性能。例如，容量优化部署通常使用速度较慢且成本更低的 SATA 驱动器和共同定位日志，而不是使用 SSD 进行日志。

成本和容量优化的存储集群具有以下属性：

- 每 TB 成本最低。
- 每 TB 的 BTU 最低。
- 每 TB 的 Watts 最低.

用于成本和容量优化的存储集群有：

- 典型的对象存储.
- 纠删代码，以最大程度地提高可用容量
- 对象存档。
- 视频、音频和图像对象存储库.

2.3. RED HAT CEPH STORAGE 的网络注意事项

云存储解决方案的一个重要方面是存储集群可能会因为网络延迟及其他因素而耗尽 IOPS。另外，存储集群可能会因为带宽限制而无法在存储集群用尽存储容量前耗尽吞吐量。这意味着网络硬件配置必须支持所选工作负载，以满足价格与性能要求。

存储管理员希望存储集群尽快恢复。仔细考虑存储集群网络的带宽要求、通过订阅的网络链接，以及隔离客户端到集群流量的集群内部流量。在考虑使用 Solid State Disks(SSD)、闪存、NVMe 和其他高性能存储设备时，还需要考虑到网络性能变得越来越重要。

Ceph 支持公共网络和存储集群网络。公共网络处理客户端流量以及与 Ceph 监控器的通信。存储集群网络处理 Ceph OSD 心跳、复制、回填和恢复流量。**至少**，存储硬件应使用 10 Gb/s 的以太网链接，您可以为连接和吞吐量添加额外的 10 Gb/s 以太网链接。



重要

红帽建议为存储集群网络分配带宽，以便它是使用 `osd_pool_default_size` 作为复制池多个池基础的公共网络的倍数。红帽还建议在单独的网卡中运行公共和存储集群网络。



重要

红帽建议在生产环境中使用 10 Gb/s 以太网部署 Red Hat Ceph Storage。1 Gb/s 以太网网络不适用于生产环境的存储集群。

如果出现驱动器故障，在 1 Gb/s 网络中复制 1 TB 数据需要 3 小时，在 10 Gb/s 网络中复制 10 TB 数据需要 30 小时。使用 10 TB 是典型的驱动器配置。与之相反，使用 10 Gb/s 以太网网络，复制 1 TB 数据需要 20 分钟，复制 10 TB 需要 1 小时。请记住，当 Ceph OSD 出现故障时，存储集群将通过将其包含的数据复制到池中的其他 Ceph OSD 来进行恢复。

对于大型环境（如机架）的故障，意味着存储集群将使用的带宽要高得多。在构建由多个机架组成的存储群集（对于大型存储实施常见）时，应考虑在“树”设计中的交换机之间利用尽可能多的网络带宽，以获得最佳性能。典型的 10 Gb/s 以太网交换机有 48 个 10 Gb/s 端口和四个 40 Gb/s 端口。使用 40 Gb/s 端口以获得最大吞吐量。或者，考虑将未使用的 10 Gb/s 端口和 QSFP+ 和 SFP+ 电缆聚合到 40 Gb/s 端口，以连接到其他机架和机械路由器。此外，还要考虑使用 LACP 模式 4 来绑定网络接口。另外，使用巨型帧、最大传输单元 (MTU) 9000，特别是在后端或集群网络上。

在安装和测试 Red Hat Ceph Storage 集群之前，请验证网络吞吐量。Ceph 中大多数与性能相关的问题通常是因为网络问题造成的。简单的网络问题（如粒度或 Bean Cat-6 电缆）可能会导致带宽下降。至少 10 Gb/s ethernet 用于前端网络。对于大型集群，请考虑将 40 Gb/s ethernet 用于后端或集群网络。



重要

为了优化网络，红帽建议使用巨型帧来获得更高的每带宽比率的 CPU，以及一个非阻塞的网络交换机后端。Red Hat Ceph Storage 在通信路径的所有网络设备中，公共和集群网络需要相同的 MTU 值。在生产环境中使用 Red Hat Ceph Storage 集群之前，验证环境中所有主机和网络设备上的 MTU 值相同。

其它资源

- 如需了解更多详细信息，请参阅 *Red Hat Ceph Storage 配置指南* 中的 [配置专用网络](#) 部分。
- 如需了解更多详细信息，请参阅 *Red Hat Ceph Storage 配置指南* 中的 [配置公共网络](#) 部分。
- 如需了解更多详细信息，请参阅 *Red Hat Ceph Storage 配置指南* 中的 [配置多个公共网络到集群](#) 部分。

2.4. 在 OSD 主机中使用 RAID 控制器的注意事项

另外，您可以考虑在 OSD 主机上使用 RAID 控制器。以下是需要考虑的一些事项：

- 如果 OSD 主机安装了 1-2 GB 缓存的 RAID 控制器，启用回写缓存可能会导致小的 I/O 写入吞吐量增加。但是，缓存必须具有非易失性。
- 大多数现代 RAID 控制器都具有超大容量，在出现电源不足时有足够的能力为非易失性 NAND 内存排空易失性内存。务必要了解特定控制器及其固件在恢复电源后的行为。
- 有些 RAID 控制器需要手动干预。硬盘驱动器通常会向操作系统播发其磁盘缓存，无论是默认启用或禁用其磁盘缓存。但是，某些 RAID 控制器和某些固件不提供此类信息。验证磁盘级别的缓存是否已禁用，以避免文件系统损坏。
- 为启用了回写缓存的每个 Ceph OSD 数据驱动器创建一个 RAID 0 卷。
- 如果 RAID 控制器中也存在 Serial Attached SCSI(SAS)或 SATA 连接的 Solid-state Drive(SSD) 磁盘，然后调查控制器和固件是否支持透传 (*pass-through*) 模式。启用透传模式有助于避免缓存逻辑，通常会降低快速介质的延迟。

2.5. 在运行 CEPH 时调整 LINUX 内核的注意事项

生产环境的 Red Hat Ceph Storage 集群通常受益于操作系统调优，尤其是关于限值和内存分配。确保为存储集群中的所有主机进行了调整。您还可以在红帽支持下创建一个问题单，寻求其他指导。

增加文件描述符数量

如果 Ceph 对象网关缺少文件描述符，它可能会挂起。您可以修改 Ceph 对象网关主机上的 `/etc/security/limits.conf` 文件，以增加 Ceph 对象网关的文件描述符。

```
ceph soft nofile unlimited
```

调整大型存储集群的 ulimit 值

在大型存储集群上运行 Ceph 管理命令时，例如，带有 1024 个 Ceph OSD 或更多 OSD，在每个运行管理命令的主机上创建一个 `/etc/security/limits.d/50-ceph.conf` 文件，其中包含以下内容：

```
USER_NAME soft nproc unlimited
```

将 `USER_NAME` 替换为运行 Ceph 管理命令的非 root 用户帐户的名称。



注意

在 Red Hat Enterprise Linux 中，root 用户的 `ulimit` 值默认设置为 `ulimit`。

2.6. COLOCATION 如何工作及其优点

您可以在同一主机上并置容器化 Ceph 守护进程。以下是合并某些 Ceph 服务的优点：

- 以小规模显著提高总拥有成本 (TCO)
- 在最低配置的情况下，从六个主机减少到三个主机
- 更轻松地进行升级

- 更好的资源隔离

Colocation 工作方式

借助 Cephadm 编配器，您可以将以下列表中的一个守护进程与一个或多个 OSD 守护进程 (ceph-osd) 并置：

- Ceph monitor (**ceph-mon**) 和 Ceph 管理器 (**ceph-mgr**) 守护进程
- 用于 Ceph 对象网关 (**nfs-ganesha**) 的 NFS Ganesha (nfs-ganesha)
- RBD 镜像(**rbd-mirror**)
- Observability Stack (Grafana)

此外，对于 Ceph 对象网关 (**radosgw**) 和 Ceph 文件系统 (**ceph-mds**)，您可以与 OSD 守护进程以及以上列表中的守护进程并置在一起，但 RBD 镜像除外。



注意

不支持在给定节点上并置同一类守护进程。



注意

因为 **ceph-mon** 和 **ceph-mgr** 可以一起工作，所以不能把两个独立的守护进程计数为两个独立的守护进程。



注意

红帽建议将 Ceph 对象网关与 Ceph OSD 容器共存以提高性能。

通过上面共享的 colocation 规则，我们有以下符合这些规则的最低集群大小：

示例 1

1. 介质：全闪存系统 (SSD)
2. 使用案例：Block (RBD) 和 File (CephFS) 或对象(Ceph 对象网关)
3. 节点数：3 个
4. 复制方案：2

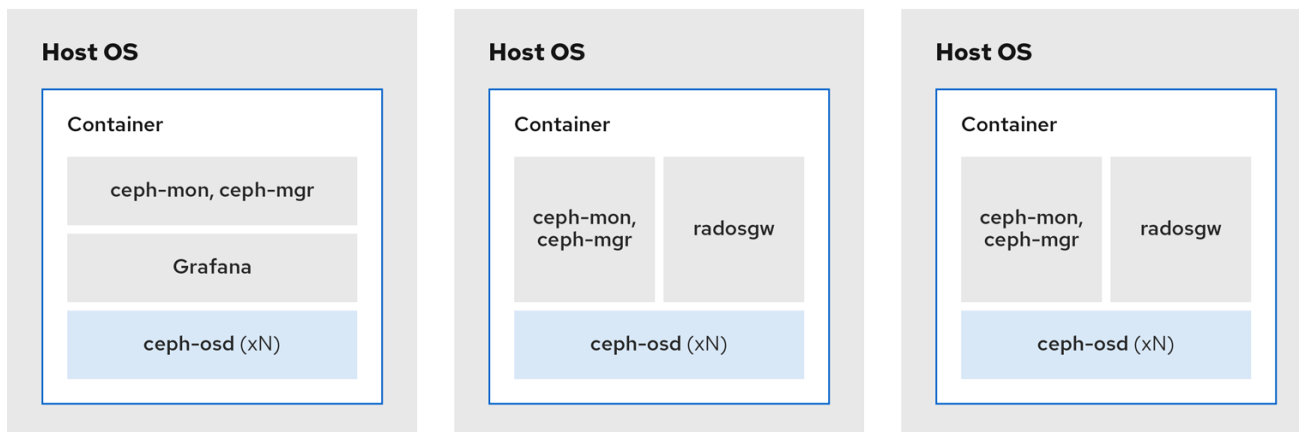
主机	Daemon	Daemon	Daemon
host1	OSD	Monitor/Manager	Grafana
host2	OSD	Monitor/Manager	RGW 或 CephFS
host3	OSD	Monitor/Manager	RGW 或 CephFS



注意

具有三个副本的存储集群的最小大小为四个节点。同样，有两个副本的存储集群的大小是三个节点集群。对于在集群中带有额外节点的复制因数需要有一定数量的节点，以避免集群在长时间内处于降级状态。

图 2.2. 并置的守护进程示例 1



336_Ceph_0423

示例 2

1. 介质：SSD 或 HDD
2. 使用案例：Block (RBD)、文件 (CephFS) 和对象 (Ceph 对象网关)
3. 节点数：4
4. 复制方案：3

主机	Daemon	Daemon	Daemon
host1	OSD	Grafana	CephFS
host2	OSD	Monitor/Manager	RGW
host3	OSD	Monitor/Manager	RGW
host4	OSD	Monitor/Manager	CephFS

图 2.3. 并置的守护进程示例 2



336_Ceph_0423

示例 3

1. 介质：SSD 或 HDD
2. 使用案例：Ceph 对象网关 (Ceph 对象网关) 和 NFS (Ceph 对象网关) 和用于 Ceph 对象网关的 NFS
3. 节点数：4
4. 复制方案：3

主机	Daemon	Daemon	Daemon
host1	OSD	Grafana	
host2	OSD	Monitor/Manager	RGW
host3	OSD	Monitor/Manager	RGW
host4	OSD	Monitor/Manager	NFS (RGW)

图 2.4. 共存的守护进程示例 3



336_Ceph_0423

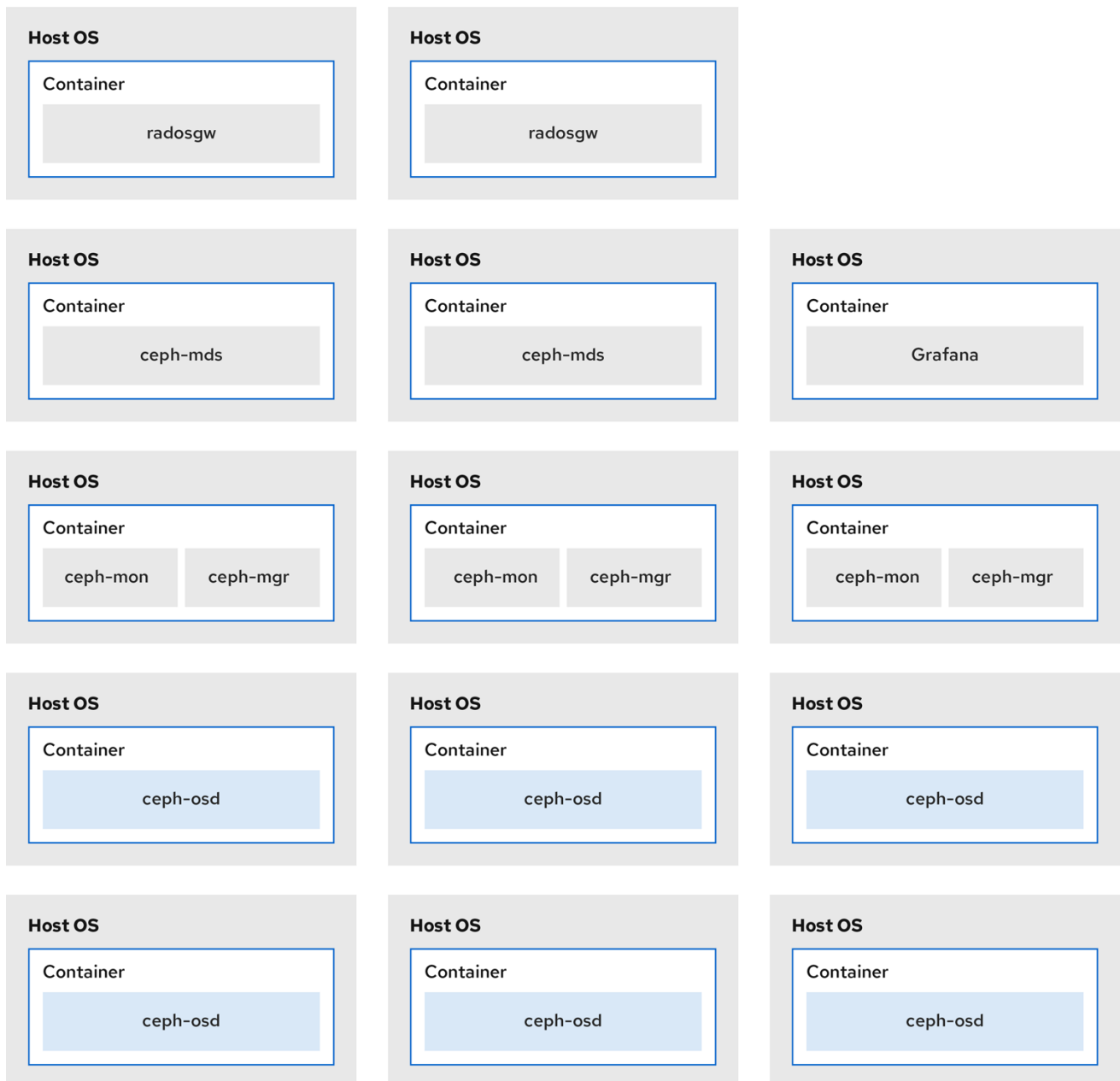
下图显示了具有并置守护进程和非并置守护进程的存储集群之间的区别。

图 2.5. colocated Daemons



336_Ceph_0423

图 2.6. 非并置守护进程



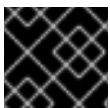
108_Ceph_0720

2.7. RED HAT CEPH STORAGE 的操作系统要求

Red Hat Enterprise Linux 权利包括在 Red Hat Ceph Storage 订阅中。Red Hat Ceph Storage 7 发行版本在 Red Hat Enterprise Linux 9.2 上被支持。

Red Hat Ceph Storage 7 仅在基于容器的部署中被支持。

在所有节点上使用相同的架构和部署类型。例如，不要使用带有 AMD64 和 Intel 64 架构的节点混合，或使用基于容器的部署混合节点。



重要

红帽不支持具有异构架构或部署类型的集群。

SELinux

默认情况下，SELinux 设置为 **Enforcing** 模式，并且安装了 **ceph-selinux** 软件包。有关 SELinux 的更多信息，请参阅 [数据安全性和强化指南](#)，以及使用 [SELinux 的 Red Hat Enterprise Linux 9](#)。

其它资源

- [Red Hat Enterprise Linux](#)

2.8. RED HAT CEPH STORAGE 的最低硬件注意事项

Red Hat Ceph Storage 可在非专有商用硬件上运行。通过使用适度的硬件，可在不优化性能的情况下运行小型生产集群和开发集群。



注意

磁盘空间要求基于 `/var/lib/ceph/` 目录下的 Ceph 守护进程默认路径。

表 2.1. 容器

Process	标准	最低建议
ceph-osd-container	处理器	每个 OSD 容器 1 个 AMD64 或 Intel 64 CPU 内核。
	RAM	每个 OSD 容器最少 5 GB RAM。
	节点数	至少需要 3 个节点。
	OS Disk	每个主机 1 个 OS 磁盘。
	OSD 存储	每个 OSD 容器 1X 存储驱动器。无法与 OS 磁盘共享。
	block.db	可选，但红帽建议每个守护进程 1x SSD 或 NVMe 或 Optane 分区或 lvm。对于对象、文件和混合工作负载，大小是 BlueStore 的 block.data 的 4%；对于块设备、Openstack cinder 和 Openstack cinder 工作负载，大小是 BlueStore 的 block.data 的 1%。
	block.wal	（可选）每个守护进程 1 个 SSD 或 NVMe 或 Optane 分区或逻辑卷。只有在速度比 block.db 设备快时，才使用一个小的大小值（如 10 GB）。
Network	2x 10 GB 以太网 NIC	ceph-mon-container
处理器	每个 mon-container 1 个 AMD64 或 Intel 64 CPU 内核	
RAM	每个 mon-container 3 GB	

Process	标准	最低建议
磁盘空间	每个 mon-container 10 GB, 但推荐 50 GB	
监控磁盘	另外, 1 个 SSD 磁盘用于 monitor rocksdb 数据	
Network	2 个 1 GB 以太网 NIC, 建议 10 GB	
Prometheus	在作为单独的文件系统创建的 /var/lib/ceph/ 目录下 20 GB 到 50 GB, 以保护 /var/ 目录下的内容。	ceph-mgr-container
处理器	每个 mgr-container 1 个 AMD64 或 Intel 64 CPU 内核	
RAM	每个 mgr-container 3 GB	
Network	2 个 1 GB 以太网 NIC, 建议 10 GB	ceph-radosgw-container
处理器	每个 radosgw-container 1 个 AMD64 或 Intel 64 CPU 内核	
RAM	每个守护进程 1 GB	
磁盘空间	每个守护进程 5 GB	
Network	1 个 1 GB 以太网 NIC	ceph-mds-container
处理器	每个 mds-container 1 个 AMD64 或 Intel 64 CPU 内核	

Process	标准	最低建议
RAM	<p>每个 mds-container 3 GB</p> <p>这个数字高度依赖于可配置的 MDS 缓存大小。RAM 要求通常为</p> <p>mds_cache_memory_limit 配置设置中设置的两倍。另请注意，这是守护进程的内存，而不是整体系统内存。</p>	
磁盘空间	<p>每个 mds-container 2 GB，并考虑可能调试日志所需的额外空间，20GB 是个不错的起点。</p>	

第 3 章 RED HAT CEPH STORAGE 安装

作为存储管理员，您可以使用 **cephadm** 实用程序来部署新的 Red Hat Ceph Storage 集群。

cephadm 实用程序管理 Ceph 集群的整个生命周期。安装和管理任务包含两种类型的操作：

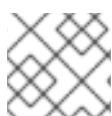
- 第一天操作涉及安装和引导在单一节点上运行的裸机最小容器化 Ceph 存储集群。第一天还包括部署 monitor 和 Manager 守护进程，以及添加 Ceph OSD。
- 第二天操作使用 Ceph 编排接口 **cephadm orch**，或 Red Hat Ceph Storage 仪表盘，通过向存储集群添加其他 Ceph 服务来扩展存储集群。

先决条件

- 至少一个正在运行的虚拟机 (VM) 或具有活跃互联网连接的裸机服务器。
- Red Hat Enterprise Linux 9.2, **ansible-core** 捆绑到 AppStream 中。
- 具有适当权利的有效的红帽订阅。
- 所有节点的根级别访问权限。
- 用于访问 Red Hat Registry 的有效红帽网络 (RHN) 或服务帐户。
- 删除了 iptables 中的可能会导致错误的配置，在刷新 iptables 服务时不会导致集群出现问题。例如，请参阅 *Red Hat Ceph Storage 配置指南* 中的 [默认 Ceph 端口部分配置的验证防火墙规则](#)。

3.1. CEPHADM 实用程序

cephadm 实用程序部署和管理 Ceph 存储集群。它与命令行界面 (CLI) 和 Red Hat Ceph Storage Dashboard Web 界面紧密集成，以便您可以从任何环境中管理存储集群。**cephadm** 使用 SSH 从管理器守护进程连接主机，以添加、移除或更新 Ceph 守护进程容器。它不依赖于外部配置或编配工具，如 Ansible 或 Rook。



注意

在主机上运行 preflight playbook 后，**cephadm** 实用程序可用。

cephadm 实用程序由两个主要组件组成：

- **cephadm** shell。
- **cephadm** 编配器。

cephadm shell

cephadm shell 在容器内启动 **bash** shell。使用 shell 完成 "Day One" 集群设置任务，如安装和引导，并使用 **ceph** 命令。

有关如何启动 **cephadm** shell 的更多信息，请参阅 [启动 cephadm shell](#)。

cephadm 编配器

使用 **cephadm** 编配器执行 "第 2 天" Ceph 功能，如扩展存储集群并调配 Ceph 守护进程和服务。您可以通过命令行界面 (CLI) 或基于 Web 的 Red Hat Ceph Storage 仪表盘来使用 **cephadm** 编配器。编配器命令采用 **ceph orch** 形式。

cephadm 脚本与 Ceph 管理器使用的 Ceph 编配模块进行交互。

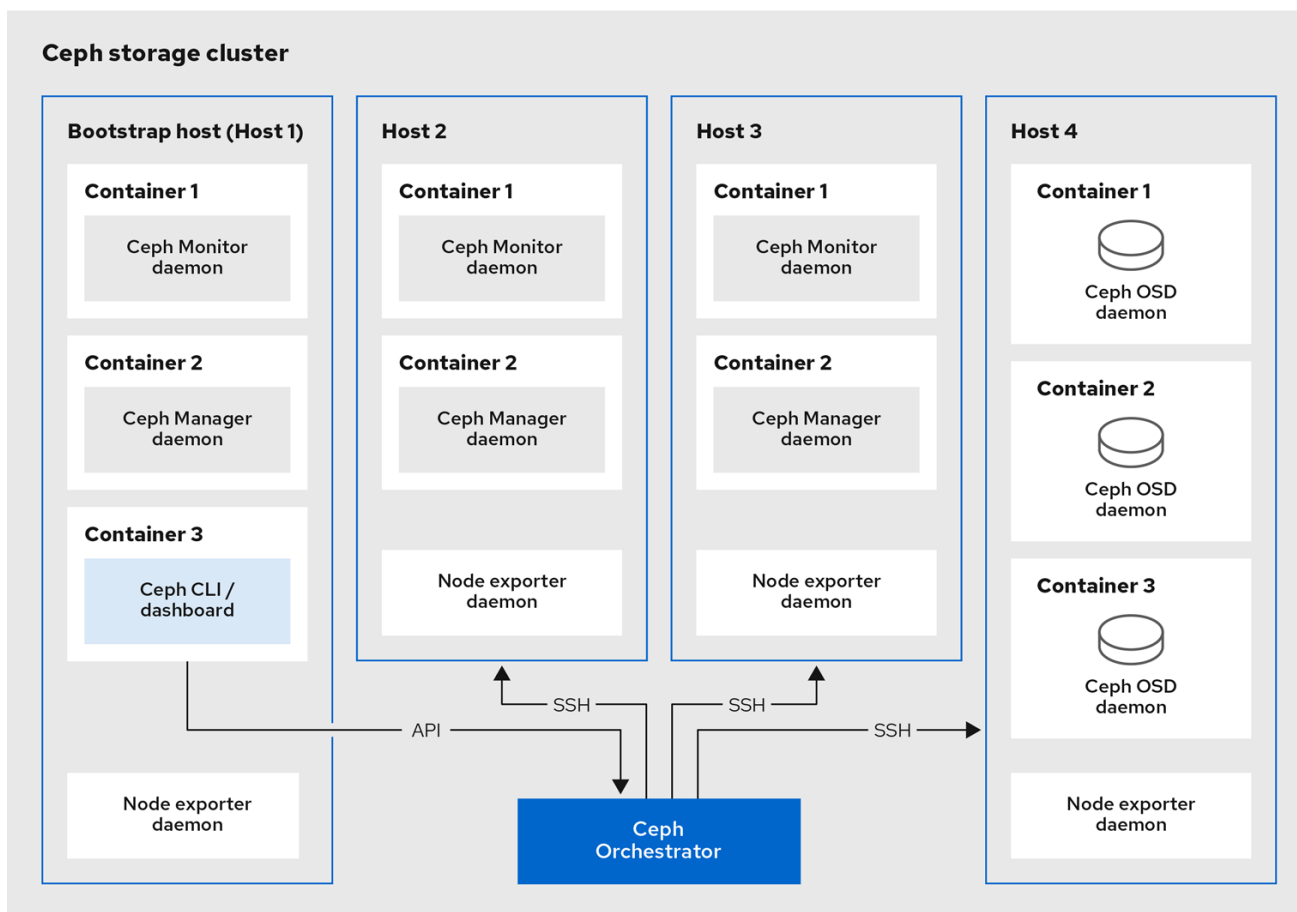
3.2. CEPHADM 的工作原理

cephadm 命令管理 Red Hat Ceph Storage 集群的完整生命周期。**cephadm** 命令可以执行以下操作：

- 引导新的 Red Hat Ceph Storage 集群。
- 启动使用 Red Hat Ceph Storage 命令行界面 (CLI) 的容器化 shell。
- 有助于调试容器化守护进程。

cephadm 命令使用 **ssh** 与存储集群中的节点通信。这样，您无需使用外部工具即可添加、删除或更新 Red Hat Ceph Storage 容器。在引导过程中生成 **ssh** 密钥对，或使用您自己的 **ssh** 密钥。

cephadm bootstrapping 过程在单一节点上创建一个小型存储集群，包含一个 Ceph monitor 和一个 Ceph 管理器，以及任何需要的依赖项。然后，您可以使用编配器 CLI 或 Red Hat Ceph Storage 仪表板来扩展存储集群，使其包含节点，并调配所有 Red Hat Ceph Storage 守护进程和服务。您可以通过 CLI 或 Red Hat Ceph Storage Dashboard Web 界面执行管理功能。



252_Ceph_0522

3.3. CEPHADM-ANSIBLE PLAYBOOK

cephadm-ansible 软件包是一个 Ansible playbook 集合，可简化由 **cephadm** 所涵盖的工作流。安装后，playbook 位于 `/usr/share/cephadm-ansible/` 中。

cephadm-ansible 软件包包括以下 playbook：

- `cephadm-preflight.yml`
- `cephadm-clients.yml`
- `cephadm-purge-cluster.yml`

cephadm-preflight playbook

在引导存储集群前，使用 `cephadm-preflight` playbook 来先设置主机，并在向您的存储集群添加新节点或客户端前。此 playbook 配置 Ceph 存储库并安装一些前提条件，如 `podman`、`lvm2`、`chronyd` 和 `cephadm`。

cephadm-clients playbook

使用 `cephadm-clients` playbook 设置客户端主机。此 playbook 将配置和密钥环文件分发给一组 Ceph 客户端。

cephadm-purge-cluster playbook

使用 `cephadm-purge-cluster` playbook 来移除 Ceph 集群。此 playbook 清除使用 `cephadm` 管理的 Ceph 集群。

其它资源

- 有关 `cephadm-preflight` playbook 的更多信息，请参阅 [运行 preflight playbook](#)。
- 有关 `cephadm-clients` playbook 的更多信息，请参阅 [运行 cephadm-clients playbook](#)。
- 有关 `cephadm-purge-cluster` playbook 的更多信息，请参阅 [填充 Ceph 存储集群](#)。

3.4. 将 RED HAT CEPH STORAGE 节点注册到 CDN 并附加订阅

Red Hat Ceph Storage 7.1 支持 Red Hat Enterprise Linux 8.10 和 9.4。



重要

使用 Red Hat Enterprise Linux 8.x 时，必须运行您的 Red Hat Ceph Storage 支持的 Red Hat Enterprise Linux 9.x 版本。

有关完整的兼容性信息，请参阅 [兼容性指南](#)。

先决条件

- 至少一个正在运行的虚拟机 (VM) 或具有活跃互联网连接的裸机服务器。
- Red Hat Enterprise Linux 8.10 或 9.4，将 `ansible-core` 捆绑到 AppStream 中。
- 具有适当权利的有效红帽订阅。
- 所有节点的根级别访问权限。

流程

1. 注册该节点，并在提示时输入您的红帽客户门户网站凭证：

语法

```
subscription-manager register
```

2. 从 CDN 拉取最新的订阅数据：

语法

```
subscription-manager refresh
```

3. 列出 Red Hat Ceph Storage 的所有可用订阅：

语法

```
subscription-manager list --available --matches 'Red Hat Ceph Storage'
```

4. 确定适当的订阅并检索其池 ID。
5. 附加池 ID 以获取软件权利的访问权限。使用您在上一步中确定的池 ID。

语法

```
subscription-manager attach --pool=POOL_ID
```

6. 禁用默认软件存储库，并在相应版本的 Red Hat Enterprise Linux 中启用服务器和附加软件仓库：

Red Hat Enterprise Linux 9

```
subscription-manager repos --disable=*  
subscription-manager repos --enable=rhel-9-for-x86_64-baseos-rpms  
subscription-manager repos --enable=rhel-9-for-x86_64-appstream-rpms
```

7. 更新系统以接收 Red Hat Enterprise Linux 的最新软件包：

语法

```
# dnf update
```

8. 订阅 Red Hat Ceph Storage 7 内容。按照[如何通过 Red Hat Satellite 6 注册 Ceph](#) 中的说明进行操作。
9. 启用 **ceph-tools** 存储库：

Red Hat Enterprise Linux 9

```
subscription-manager repos --enable=rhceph-7-tools-for-rhel-9-x86_64-rpms
```

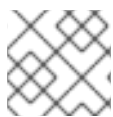
10. 在您要添加到集群中的所有节点上重复上述步骤。
11. 安装 **cephadm-ansible**：

语法

```
dnf install cephadm-ansible
```

3.5. 配置 ANSIBLE 清单位置

您可以为 **cephadm-ansible** 暂存和生产环境配置清单位置文件。Ansible 清单主机文件包含属于存储集群的所有主机。您可以在清单主机文件中单独列出节点，也可以创建 **[mons]**、**[osds]** 和 **[rgws]** 等组，以提供有关您的清单，并简化在运行 playbook 时将 **--limit** 选项用于目标组或节点的组。



注意

如果部署客户端，必须在专用 **[clients]** 组中定义客户端节点。

先决条件

- Ansible 管理节点。
- 对 Ansible 管理节点的根级别访问权限。
- **cephadm-ansible** 软件包安装在节点上。

流程

1. 进入 **/usr/share/cephadm-ansible/** 目录：

```
[root@admin ~]# cd /usr/share/cephadm-ansible
```

2. 可选：为 staging 和 production 创建子目录：

```
[root@admin cephadm-ansible]# mkdir -p inventory/staging inventory/production
```

3. 可选：编辑 **ansible.cfg** 文件并添加以下行来分配默认清单位置：

```
[defaults]
inventory = ./inventory/staging
```

4. 可选：为每个环境创建一个清单主机文件：

```
[root@admin cephadm-ansible]# touch inventory/staging/hosts
[root@admin cephadm-ansible]# touch inventory/production/hosts
```

5. 打开并编辑每个 **hosts** 文件，并添加节点和 **[admin]** 组：

```
NODE_NAME_1
NODE_NAME_2

[admin]
ADMIN_NODE_NAME_1
```

- 将 **NODE_NAME_1** 和 **NODE_NAME_2** 替换为 Ceph 节点，如 monitor、OSD、MDS 和网关节点。
- 将 **ADMIN_NODE_NAME_1** 替换为存储了 admin 密钥环的节点的名称。

示例

```
host02
host03
host04

[admin]
host01
```



注意

如果将 **ansible.cfg** 文件中的清单位置设置为 staging，则需要在暂存环境中运行 playbook，如下所示：

语法

```
ansible-playbook -i inventory/staging/hosts PLAYBOOK.yml
```

在生产环境中运行 playbook：

语法

```
ansible-playbook -i inventory/production/hosts PLAYBOOK.yml
```

3.6. 在 RED HAT ENTERPRISE LINUX 9 中以 ROOT 用户身份启用 SSH 登录

Red Hat Enterprise Linux 9 不支持以 root 用户身份 SSH 登录，即使 **/etc/ssh/sshd_config** 文件中将 **PermitRootLogin** 参数设置为 **yes**。您会收到以下错误：

示例

```
[root@host01 ~]# ssh root@myhostname
root@myhostname password:
Permission denied, please try again.
```

您可以使用以下方法之一启用以 root 用户身份登录：

- 在安装 Red Hat Enterprise Linux 9 的过程中，使用 "Allow root SSH login with password" 标记。
- 在 Red Hat Enterprise Linux 9 安装后手动设置 **PermitRootLogin** 参数。

本节论述了 **PermitRootLogin** 参数的手动设置。

先决条件

- 所有节点的根级别访问权限。

流程

1. 打开 **etc/ssh/sshd_config** 文件，并将 **PermitRootLogin** 设置为 **yes**：

示例

```
[root@admin ~]# echo 'PermitRootLogin yes' >> /etc/ssh/sshd_config.d/01-permitrootlogin.conf
```

2. 重启 **SSH** 服务：

示例

```
[root@admin ~]# systemctl restart sshd.service
```

3. 以 **root** 用户身份登录节点：

语法

```
ssh root@HOST_NAME
```

将 *HOST_NAME* 替换为 Ceph 节点的主机名。

示例

```
[root@admin ~]# ssh root@host01
```

出现提示时，输入 **root** 密码。

其它资源

- 如需更多信息，请参阅在 [RHEL 9 服务器知识库解决方案中通过 ssh 以 root 用户身份登录](#)。

3.7. 创建具有 **sudo** 访问权限的 **ANSIBLE** 用户

您可以在存储集群的所有节点上创建一个具有无密码 **root** 访问权限的 Ansible 用户，以运行 **cephadm-ansible** playbook。Ansible 用户必须能够以具有 **root** 权限的用户身份登录所有 Red Hat Ceph Storage 节点，以安装软件并创建配置文件而不提示输入密码。

先决条件

- 所有节点的根级别访问权限。
- 对于 Red Hat Enterprise Linux 9，要以 root 用户身份登录，请参阅在 [Red Hat Enterprise Linux 9 中以 root 用户身份启用 SSH 登录](#)

流程

1. 以 **root** 用户身份登录节点：

语法

```
ssh root@HOST_NAME
```

将 *HOST_NAME* 替换为 Ceph 节点的主机名。

示例

```
[root@admin ~]# ssh root@host01
```

出现提示时，输入 **root** 密码。

2. 创建一个新的 Ansible 用户：

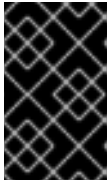
语法

```
adduser USER_NAME
```

将 *USER_NAME* 替换为 Ansible 用户的新用户名。

示例

```
[root@host01 ~]# adduser ceph-admin
```



重要

不要使用 **ceph** 作为用户名。**ceph** 用户名保留用于 Ceph 守护进程。整个集群中的统一用户名可以提高易用性，但避免使用明显的用户名，因为入侵者通常使用它们进行暴力攻击。

3. 为这个用户设置一个新密码：

语法

```
passwd USER_NAME
```

将 *USER_NAME* 替换为 Ansible 用户的新用户名。

示例

```
[root@host01 ~]# passwd ceph-admin
```

出现提示时，输入新密码两次。

4. 为新创建的用户配置 **sudo** 访问权限：

语法

```
cat << EOF >/etc/sudoers.d/USER_NAME
USER_NAME ALL = (root) NOPASSWD:ALL
EOF
```

将 *USER_NAME* 替换为 Ansible 用户的新用户名。

示例

```
[root@host01 ~]# cat << EOF >/etc/sudoers.d/ceph-admin
ceph-admin ALL = (root) NOPASSWD:ALL
EOF
```

5. 为新文件分配正确的文件权限：

语法

```
chmod 0440 /etc/sudoers.d/USER_NAME
```

将 *USER_NAME* 替换为 Ansible 用户的新用户名。

示例

```
[root@host01 ~]# chmod 0440 /etc/sudoers.d/ceph-admin
```

6. 在存储集群中的所有节点上重复上述步骤。

其它资源

- 有关创建用户帐户的更多信息，请参阅 Red Hat Enterprise Linux 9 指南中的 [配置基本系统设置](#) 一章中的 [管理用户帐户](#) 部分。

3.8. 为 ANSIBLE 启用免密码 SSH

在 Ansible 管理节点上生成 SSH 密钥对，并将公钥分发到存储集群中的每个节点，以便 Ansible 可以在不提示输入密码的情况下访问节点。

先决条件

- 访问 Ansible 管理节点。
- 具有 sudo 对存储集群中所有节点的访问权限的 Ansible 用户。
- 对于 Red Hat Enterprise Linux 9，要以 root 用户身份登录，[请参阅在 Red Hat Enterprise Linux 9 中以 root 用户身份启用 SSH 登录](#)

流程

1. 生成 SSH 密钥对，接受默认文件名并将密语留空：

```
[ceph-admin@admin ~]$ ssh-keygen
```

2. 将公钥复制到存储集群中的所有节点：

```
ssh-copy-id USER_NAME@HOST_NAME
```

将 *USER_NAME* 替换为 Ansible 用户的新用户名。将 *HOST_NAME* 替换为 Ceph 节点的主机名。

示例

```
[ceph-admin@admin ~]$ ssh-copy-id ceph-admin@host01
```

3. 创建用户的 SSH **config** 文件：

```
[ceph-admin@admin ~]$ touch ~/.ssh/config
```

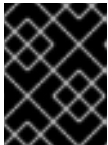
4. 打开并编辑 **config** 文件。为存储集群中每个节点的 **Hostname** 和 **User** 选项设置值：

```
Host host01
  Hostname HOST_NAME
  User USER_NAME
Host host02
  Hostname HOST_NAME
  User USER_NAME
...
```

将 *HOST_NAME* 替换为 Ceph 节点的主机名。将 *USER_NAME* 替换为 Ansible 用户的新用户名。

示例

```
Host host01
  Hostname host01
  User ceph-admin
Host host02
  Hostname host02
  User ceph-admin
Host host03
  Hostname host03
  User ceph-admin
```



重要

通过配置 `~/.ssh/config` 文件，您不必在每次执行 `ansible-playbook` 命令时指定 `-u USER_NAME` 选项。

5. 为 `~/.ssh/config` 文件设置正确的文件权限：

```
[ceph-admin@admin ~]$ chmod 600 ~/.ssh/config
```

其它资源

- [ssh_config\(5\) 手册页面](#)。
- 请参阅 [使用 OpenSSH 的两个系统间的安全通信](#)。

3.9. 运行 PREFLIGHT PLAYBOOK

此 Ansible playbook 配置 Ceph 存储库，并准备用于引导的存储集群。它还会安装一些前提条件，如 **podman**、**lvm2**、**chronyd** 和 **cephadm**。**cephadm-ansible** 和 **cephadm-preflight.yml** 的默认位置为 `/usr/share/cephadm-ansible`。

preflight playbook 使用 **cephadm-ansible** 清单文件来识别存储集群中的所有 admin 和节点。

清单文件的默认位置为 `/usr/share/cephadm-ansible/hosts`。以下示例显示了典型的清单文件的结构：

示例

```
host02
host03
```



```
host04
```

```
[admin]
host01
```

清单文件中的 **[admin]** 组包含存储了 admin 密钥环的节点的名称。在新存储集群中，**[admin]** 组中的节点将是 bootstrap 节点。要在引导集群后添加额外的 *admin 主机*，[请参阅安装指南中的设置管理节点](#)。



注意

在引导初始主机前运行 preflight playbook。



重要

如果要执行断开连接的安装，[请参阅运行 preflight playbook 以断开连接安装](#)。

先决条件

- 对 Ansible 管理节点的根级别访问权限。
- 具有 sudo 的 Ansible 用户，对存储集群中所有节点的 **ssh** 访问和免密码访问。



注意

在以下示例中，host01 是 bootstrap 节点。

流程

1. 进入 `/usr/share/cephadm-ansible` 目录。
2. 打开并编辑 `hosts` 文件并添加节点：

示例

```
host02
host03
host04

[admin]
host01
```

3. 运行 preflight playbook：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars "ceph_origin=rhcs"
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --extra-vars "ceph_origin=rhcs"
```

安装完成后，`cephadm` 驻留在 `/usr/sbin/` 目录中。

- 使用 `--limit` 选项在存储集群中的一组主机上运行 `preflight` playbook :

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars
"ceph_origin=rhcs" --limit GROUP_NAME|NODE_NAME
```

将 `GROUP_NAME` 替换为清单文件中的组名称。将 `NODE_NAME` 替换为清单文件中的特定节点名称。



注意

另外，您可以根据组名称，如 `[mons]`、`[osds]` 和 `[mgrs]` 等对节点进行分组。但是，必须将管理节点添加到 `[admin]` 组和客户端，必须将客户端添加到 `[clients]` 组中。

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml
--extra-vars "ceph_origin=rhcs" --limit clients
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml
--extra-vars "ceph_origin=rhcs" --limit host01
```

- 运行 `preflight` playbook 时，`cephadm-ansible` 会自动在客户端节点上安装 `chronyd` 和 `ceph-common`。
`preflight` playbook 安装 `chronyd`，但为单个 NTP 源进行配置。如果要配置多个源，或者有断开连接的环境，请参阅以下文档以了解更多信息：
 - [如何配置 chrony ?](#)
 - [NTP 的最佳实践。](#)
 - [基本 chrony NTP 故障排除。](#)

3.10. 引导新存储集群

`cephadm` 实用程序在 `bootstrap` 过程中执行以下任务：

- 为本地节点上的新 Red Hat Ceph Storage 集群安装并启动 Ceph 监控守护进程和 Ceph 管理器守护进程，作为容器。
- 创建 `/etc/ceph` 目录。
- 将公钥的副本写入 Red Hat Ceph Storage 集群的 `/etc/ceph/ceph.pub`，并将 SSH 密钥添加到 `root` 用户的 `/root/.ssh/authorized_keys` 文件中。
- 将 `_admin` 标签应用到 `bootstrap` 节点。
- 编写与新集群通信所需的最小配置文件到 `/etc/ceph/ceph.conf`。
- 将 `client.admin` 管理 `secret` 密钥的副本写入 `/etc/ceph/ceph.client.admin.keyring`。
- 使用 `prometheus`、`grafana` 和其他工具（如 `node-exporter` 和 `alert-manager`）部署基本的监控堆栈。

**重要**

如果要执行断开连接的安装，请参阅 [执行断开连接的安装](#)。

**注意**

如果您要使用新存储集群运行 prometheus 服务，或者您使用 Rook 运行 Ceph，请将 **--skip-monitoring-stack** 选项与 **cephadm bootstrap** 命令搭配使用。这个选项绕过基本的监控堆栈，以便稍后手动配置。

**重要**

如果要部署监控堆栈，请参阅 Red Hat Ceph Storage Operations Guide 中的 [使用 Ceph Orchestrator 部署监控堆栈](#)。

**重要**

bootstrapping 提供初始登录控制板的默认用户名和密码。Bootstrap 要求在登录后更改密码。

**重要**

在开始 bootstrap 过程前，请确保要使用的容器镜像与 **cephadm** 具有相同的 Red Hat Ceph Storage 版本。如果两个版本不匹配，bootstrapping 会在创建初始 **admin** 用户阶段失败。

**注意**

在开始 bootstrap 过程前，您必须为 **registry.redhat.io** 容器 registry 创建用户名和密码。有关红帽容器 registry 身份验证的更多信息，请参阅知识库文章 [Red Hat Container Registry 身份验证](#)

先决条件

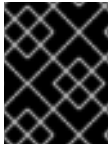
- 第一个 Ceph 监控容器的 IP 地址，也是存储集群中第一个节点的 IP 地址。
- 登录到 **registry.redhat.io**。
- 至少 10 GB 的可用空间用于 **/var/lib/containers/**。
- 所有节点的根级别访问权限。

**注意**

如果存储集群包含多个网络和接口，请确定选择一个可供使用存储集群的任何节点访问的网络。

**注意**

如果本地节点使用完全限定域名 (FQDN)，则将 **--allow-fqdn-hostname** 选项添加到命令行上的 **cephadm bootstrap**。



重要

在您要作为集群中初始监控节点的节点上运行 **cephadm bootstrap**。IP_ADDRESS 选项应该是您用于运行 **cephadm bootstrap** 的节点的 IP 地址。



注意

如果要使用 IPV6 地址部署存储集群，则使用 **--mon-ip IP_ADDRESS** 选项的 IPV6 地址格式。例如：**cephadm bootstrap --mon-ip 2620:52:0:880:225:90ff:fe6c:2536 --registry-json /etc/mylogin.json**

流程

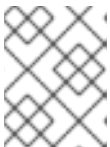
1. 引导一个存储集群：

语法

```
cephadm bootstrap --cluster-network NETWORK_CIDR --mon-ip IP_ADDRESS --registry-url registry.redhat.io --registry-username USER_NAME --registry-password PASSWORD --yes-i-know
```

示例

```
[root@host01 ~]# cephadm bootstrap --cluster-network 10.10.128.0/24 --mon-ip 10.10.128.68 --registry-url registry.redhat.io --registry-username myuser1 --registry-password mypassword1 --yes-i-know
```



注意

如果要通过公共网络路由的内部集群流量，可以省略 **--cluster-network NETWORK_CIDR** 选项。

完成该脚本需要几分钟时间。脚本完成后，会提供 Red Hat Ceph Storage Dashboard URL 提供凭据、用于访问 Ceph 命令行界面 (CLI) 的命令，以及启用遥测的请求。

Ceph Dashboard is now available at:

```
URL: https://host01:8443/
User: admin
Password: i8nhu7zham
```

Enabling client.admin keyring and conf on hosts with "admin" label
You can access the Ceph CLI with:

```
sudo /usr/sbin/cephadm shell --fsid 266ee7a8-2a05-11eb-b846-5254002d4916 -c /etc/ceph/ceph.conf -k /etc/ceph/ceph.client.admin.keyring
```

Please consider enabling telemetry to help improve Ceph:

```
ceph telemetry on
```

For more information see:

<https://docs.ceph.com/docs/master/mgr/telemetry/>

Bootstrap complete.

其它资源

- 有关推荐的 bootstrap 命令选项的更多信息，请参阅[推荐的 cephadm bootstrap 命令选项](#)。
- 有关 bootstrap 命令可用选项的更多信息，请参阅[Bootstrap 命令选项](#)。
- 有关使用 JSON 文件包含 bootstrap 过程的登录凭证的详情，请参考[使用 JSON 文件来保护登录信息](#)。

3.10.1. 推荐的 cephadm bootstrap 命令选项

cephadm bootstrap 命令具有多个选项，供您指定文件位置，配置 ssh 设置、设置密码，以及执行其他初始配置任务。

红帽建议您为 **cephadm bootstrap** 使用一组基本命令选项。您可以在初始集群启动并运行后配置附加选项。

以下示例演示了如何指定推荐的选项。

语法

```
cephadm bootstrap --ssh-user USER_NAME --mon-ip IP_ADDRESS --allow-fqdn-hostname --registry-json REGISTRY_JSON
```

示例

```
[root@host01 ~]# cephadm bootstrap --ssh-user ceph --mon-ip 10.10.128.68 --allow-fqdn-hostname --registry-json /etc/mylogin.json
```

其它资源

- 有关 **--registry-json** 选项的更多信息，请参阅[使用 JSON 文件保护登录信息](#)。
- 有关所有可用 **cephadm bootstrap** 选项的更多信息，请参阅[Bootstrap 命令选项](#)。
- 有关以非 root 用户身份引导存储集群的更多信息，请参阅[以非 root 用户身份引导存储集群](#)。

3.10.2. 使用 JSON 文件保护登录信息

作为存储管理员，您可以选择添加登录和密码信息到 JSON 文件，然后引用 JSON 文件以进行引导。这样可防止登录凭据被暴露。



注意

您还可以将 JSON 文件与 **cephadm --registry-login** 命令搭配使用。

先决条件

- 第一个 Ceph 监控容器的 IP 地址，也是存储集群中第一个节点的 IP 地址。

- 登录到 registry.redhat.io。
- 至少 10 GB 的可用空间用于 `/var/lib/containers/`。
- 所有节点的根级别访问权限。

流程

1. 创建 JSON 文件。在本例中，该文件名为 `mylogin.json`。

语法

```
{
  "url": "REGISTRY_URL",
  "username": "USER_NAME",
  "password": "PASSWORD"
}
```

示例

```
{
  "url": "registry.redhat.io",
  "username": "myuser1",
  "password": "mypassword1"
}
```

2. 引导一个存储集群：

语法

```
cephadm bootstrap --mon-ip IP_ADDRESS --registry-json /etc/mylogin.json
```

示例

```
[root@host01 ~]# cephadm bootstrap --mon-ip 10.10.128.68 --registry-json /etc/mylogin.json
```

3.10.3. 使用服务配置文件引导存储集群

要引导存储集群并使用服务配置文件配置额外的主机和守护进程，请使用 `cephadm bootstrap` 命令的 `--apply-spec` 选项。配置文件是一个 `.yaml` 文件，其中包含您要部署的服务类型、放置和指定节点。



注意

如果要为多站点等应用使用非默认域或区域，请在引导存储集群后配置 Ceph 对象网关守护进程，而不是使用 `--apply-spec` 选项。这可让您在部署 Ceph Object Gateway 前创建 Ceph 对象网关守护进程所需的 realm 或 zone。如需更多信息，请参阅 [Red Hat Ceph Storage 操作指南](#)。



注意

如果部署 NFS-Ganesha 网关或元数据服务器(MDS)服务, 请在引导存储集群后配置它们。

- 若要部署 Ceph NFS-Ganesha 网关, 您必须首先创建一个 RADOS 池。
- 要部署 MDS 服务, 您必须首先创建一个 CephFS 卷。

如需更多信息, 请参阅 [Red Hat Ceph Storage 操作指南](#)。

先决条件

- 至少一个正在运行的虚拟机 (VM) 或服务器。
- Red Hat Enterprise Linux 8.10 或 9.4, 将 **ansible-core** 捆绑到 AppStream 中。
- 所有节点的根级别访问权限。
- 登录到 registry.redhat.io。
- 在存储集群的所有主机上设置免密码 **ssh**。
- **cephadm** 安装在您要作为存储集群中初始监控节点的节点上。

流程

1. 登录到 bootstrap 主机。
2. 为您的存储集群创建服务配置 **.yaml** 文件。示例文件指示 **cephadm bootstrap** 配置初始主机和两个额外的主机, 并且它指定将在所有可用的磁盘上创建 OSD。

示例

```

service_type: host
addr: host01
hostname: host01
---
service_type: host
addr: host02
hostname: host02
---
service_type: host
addr: host03
hostname: host03
---
service_type: host
addr: host04
hostname: host04
---
service_type: mon
placement:
  host_pattern: "host[0-2]"
---
service_type: osd
service_id: my_osds

```

```
placement:
  host_pattern: "host[1-3]"
data_devices:
  all: true
```

3. 使用 **--apply-spec** 选项引导存储集群：

语法

```
cephadm bootstrap --apply-spec CONFIGURATION_FILE_NAME --mon-ip
MONITOR_IP_ADDRESS --registry-url registry.redhat.io --registry-username USER_NAME
--registry-password PASSWORD
```

示例

```
[root@host01 ~]# cephadm bootstrap --apply-spec initial-config.yaml --mon-ip 10.10.128.68 -
--registry-url registry.redhat.io --registry-username myuser1 --registry-password
mypassword1
```

完成该脚本需要几分钟时间。脚本完成后，会提供 Red Hat Ceph Storage Dashboard URL 提供凭据、用于访问 Ceph 命令行界面 (CLI) 的命令，以及启用遥测的请求。

4. 在您的存储集群启动并运行后，请参阅 [Red Hat Ceph Storage Operations Guide](#) 以了解有关配置其他守护进程和服务的更多信息。

其它资源

- 有关 bootstrap 命令可用选项的更多信息，请参阅 [Bootstrap 命令选项](#)。

3.10.4. 以非 root 用户身份引导存储集群

如果您有免密码 sudo 权限，则可以以非 root 用户身份引导存储集群。

要在 bootstrap 节点上以非 root 用户身份引导 Red Hat Ceph Storage 集群，在 **cephadm bootstrap** 命令中使用 **--ssh-user** 选项。**--ssh-user** 指定 SSH 连接到集群节点的用户。

非 root 用户必须具有免密码 **sudo** 访问权限。

先决条件

- 第一个 Ceph Monitor 容器的 IP 地址，这也是存储集群中初始 monitor 节点的 IP 地址。
- 登录到 **registry.redhat.io**。
- 至少 10 GB 的可用空间用于 **/var/lib/containers/**。
- 可选：SSH 公钥和私钥。
- 对 bootstrap 节点进行免密码 **sudo** 访问。
- 非 root 用户在所有旨在作为集群一部分的节点上具有免密码 **sudo** 访问权限。
- **cephadm** 安装在您要作为存储集群中初始监控节点的节点上。

流程

1. 更改为 bootstrap 节点上的 **sudo** :

语法

```
su - SSH_USER_NAME
```

示例

```
[root@host01 ~]# su - ceph
Last login: Tue Sep 14 12:00:29 EST 2021 on pts/0
```

2. 检查到 bootstrap 节点的 SSH 连接 :

示例

```
[ceph@host01 ~]$ ssh host01
Last login: Tue Sep 14 12:03:29 EST 2021 on pts/0
```

3. 可选 : 指示 **cephadm bootstrap** 命令。



注意

使用私钥和公钥是可选的。如果之前尚未创建 SSH 密钥，则可以在此步骤中创建这些密钥。

包含 **--ssh-private-key** 和 **--ssh-public-key** 选项 :

语法

```
sudo cephadm bootstrap --ssh-user USER_NAME --mon-ip IP_ADDRESS --ssh-private-key PRIVATE_KEY --ssh-public-key PUBLIC_KEY --registry-url registry.redhat.io --registry-username USER_NAME --registry-password PASSWORD
```

示例

```
sudo cephadm bootstrap --ssh-user ceph --mon-ip 10.10.128.68 --ssh-private-key /home/ceph/.ssh/id_rsa --ssh-public-key /home/ceph/.ssh/id_rsa.pub --registry-url registry.redhat.io --registry-username myuser1 --registry-password mypassword1
```

其它资源

- 有关所有可用 **cephadm bootstrap** 选项的更多信息，请参阅 [Bootstrap 命令选项](#)。
- 有关使用 Ansible 自动引导无根集群的更多信息，请参阅使用 [ansible 临时命令的知识库文章 Red Hat Ceph Storage 6 rootless 部署](#)。
- 有关 sudo 权限的更多信息，请参阅 [管理 sudo 访问](#)。

3.10.5. bootstrap 命令选项

cephadm bootstrap 命令在本地主机上引导 Ceph 存储集群。它在本地主机上部署 MON 守护进程和 MGR 守护进程，自动在本地主机上部署监控堆栈，并调用 **ceph orch host add HOSTNAME**。

下表列出了 **cephadm bootstrap** 的可用选项。

cephadm bootstrap 选项	描述
<code>--config CONFIG_FILE, -c CONFIG_FILE</code>	<code>CONFIG_FILE</code> 是要与 bootstrap 命令一起使用的 ceph.conf 文件
<code>--cluster-network NETWORK_CIDR</code>	将 <code>NETWORK_CIDR</code> 定义的子网用于内部集群流量。以 CIDR 表示法指定。例如： 10.10.128.0/24 。
<code>--mon-id MON_ID</code>	在名为 <code>MON_ID</code> 的主机上引导。默认值为本地主机。
<code>--mon-addrv MON_ADDRV</code>	mon IPs (例如 [v2:localipaddr:3300,v1:localipaddr:6789])
<code>--mon-ip IP_ADDRESS</code>	用于运行 cephadm bootstrap 的节点的 IP 地址。
<code>--mgr-id MGR_ID</code>	安装 MGR 节点的主机 ID。默认：随机生成。
<code>--fsid FSID</code>	集群 FSID。
<code>--output-dir OUTPUT_DIR</code>	使用此目录编写配置、密钥环和公钥文件。
<code>--output-keyring OUTPUT_KEYRING</code>	使用新的集群管理员和 mon 密钥生成的密钥环文件要写入的位置。
<code>--output-config OUTPUT_CONFIG</code>	写入用于连接到新集群的配置文件的写入位置。
<code>--output-pub-ssh-key OUTPUT_PUB_SSH_KEY</code>	用于集群的公共 SSH 公钥的写入位置。
<code>--skip-ssh</code>	跳过本地主机上 ssh 密钥的设置。
<code>--initial-dashboard-user INITIAL_DASHBOARD_USER</code>	仪表板的初始用户。
<code>--initial-dashboard-password INITIAL_DASHBOARD_PASSWORD</code>	仪表板初始用户的初始密码。
<code>--ssl-dashboard-port SSL_DASHBOARD_PORT</code>	用于使用 SSL 与仪表板连接的端口号。
<code>--dashboard-key DASHBOARD_KEY</code>	仪表板密钥。
<code>--dashboard-crt DASHBOARD_CRT</code>	仪表板证书。
<code>--ssh-config SSH_CONFIG</code>	SSH 配置。

cephadm bootstrap 选项	描述
<code>--ssh-private-key SSH_PRIVATE_KEY</code>	SSH 私钥。
<code>--ssh-public-key SSH_PUBLIC_KEY</code>	SSH 公钥。
<code>--ssh-user SSH_USER</code>	设置用于到集群主机的 SSH 连接的用户。非 root 用户需要免密码 sudo。
<code>--skip-mon-network</code>	根据 bootstrap mon ip 设置 mon public_network。
<code>--skip-dashboard</code>	不要启用 Ceph 仪表板。
<code>--dashboard-password-noupdate</code>	禁用强制仪表板密码更改。
<code>--no-minimize-config</code>	不要模拟和最小化配置文件。
<code>--skip-ping-check</code>	不验证 mon IP 是否可 ping 通。
<code>--skip-pull</code>	在 bootstrapping 前不要拉取最新的镜像。
<code>--skip-firewalld</code>	不配置 firewalld。
<code>--allow-overwrite</code>	允许覆盖现有的 -output-* config/keyring/ssh 文件。
<code>--allow-fqdn-hostname</code>	允许完全限定主机名。
<code>--skip-prepare-host</code>	不准备主机。
<code>--orphan-initial-daemons</code>	不创建初始 mon、mgr 和崩溃服务规格。
<code>--skip-monitoring-stack</code>	不自动置备监控堆栈] (prometheus、grafana、alertmanager、node-exporter) 。
<code>--apply-spec APPLY_SPEC</code>	在 bootstrap 后应用集群 spec 文件 (复制 ssh 密钥、添加主机和应用服务) 。
<code>--registry-url REGISTRY_URL</code>	指定要登录的自定义 registry 的 URL。例如： registry.redhat.io 。
<code>--registry-username REGISTRY_USERNAME</code>	到自定义 registry 的登录帐户的用户名。
<code>--registry-password REGISTRY_PASSWORD</code>	到自定义 registry 的登录帐户的密码。
<code>--registry-json REGISTRY_JSON</code>	包含 registry 登录信息的 JSON 文件。

其它资源

- 有关 `--skip-monitoring-stack` 选项的更多信息，请参阅 [添加主机](#)。
- 有关使用 `registry-json` 选项登录 registry 的更多信息，请参阅 `registry-login` 命令的帮助信息。
- 如需有关 `cephadm` 选项的更多信息，请参阅 `cephadm` 的帮助。

3.10.6. 为断开连接的安装配置私有 registry

您可以使用断开连接的安装过程在专用网络上安装 `cephadm` 和 bootstrap 您的存储集群。断开连接的安装使用私有 registry 进行安装。在部署过程中，Red Hat Ceph Storage 节点无法访问互联网时，请使用这个步骤。

按照以下步骤，使用身份验证和自签名证书设置安全私有 registry。在可以访问互联网以及可以访问本地集群的节点上执行这些步骤。



注意

不建议将不安全的 registry 用于生产环境。

先决条件

- 至少一个正在运行的虚拟机(VM)或带有活跃互联网连接的服务器。
- Red Hat Enterprise Linux 8.10 或 9.4，将 `ansible-core` 捆绑到 AppStream 中。
- 登录到 `registry.redhat.io`。
- 所有节点的根级别访问权限。

流程

1. 登录到可访问公共网络和集群节点的节点。
2. 注册该节点，并在提示时输入适当的红帽客户门户网站凭证：

示例

```
[root@admin ~]# subscription-manager register
```

3. 获取最新的订阅数据：

示例

```
[root@admin ~]# subscription-manager refresh
```

4. 列出 Red Hat Ceph Storage 的所有可用订阅：

示例

```
[root@admin ~]# subscription-manager list --available --all --matches="*Ceph*"
```

从 Red Hat Ceph Storage 可用订阅列表中复制池 ID。

5. 附加订阅以获取软件权利：

语法

```
subscription-manager attach --pool=POOL_ID
```

将 POOL_ID 替换为上一步中标识的池 ID。

- 禁用默认软件存储库，并启用服务器以及额外的存储库：

Red Hat Enterprise Linux 9

```
[root@admin ~]# subscription-manager repos --disable=*
[root@admin ~]# subscription-manager repos --enable=rhel-9-for-x86_64-baseos-rpms
[root@admin ~]# subscription-manager repos --enable=rhel-9-for-x86_64-appstream-rpms
```

- 安装 **podman** 和 **httpd-tools** 软件包：

示例

```
[root@admin ~]# dnf install -y podman httpd-tools
```

- 为私有 registry 创建文件夹：

示例

```
[root@admin ~]# mkdir -p /opt/registry/{auth,certs,data}
```

registry 将存储在 **/opt/registry** 中，目录会挂载到运行 registry 的容器中。

- **auth** 目录存储 registry 用于身份验证的 **htpasswd** 文件。
- **certs** 目录存储 registry 的证书用于身份验证。
- **data** 目录存储 registry 镜像。

- 创建用于访问私有 registry 的凭证：

语法

```
htpasswd -bBc /opt/registry/auth/htpasswd PRIVATE_REGISTRY_USERNAME
PRIVATE_REGISTRY_PASSWORD
```

- **b** 选项提供从命令行的密码。
- **B** 选项使用 **Bcrypt** 加密存储密码。
- **c** 选项创建 **htpasswd** 文件。
- 将 PRIVATE_REGISTRY_USERNAME 替换为要用于私有 registry 的用户名。
- 将 PRIVATE_REGISTRY_PASSWORD 替换为要用于私有 registry 用户名的密码。

示例

```
[root@admin ~]# htpasswd -bBc /opt/registry/auth/htpasswd myregistryusername
myregistrypassword1
```

10. 创建自签名证书：

语法

```
openssl req -newkey rsa:4096 -nodes -sha256 -keyout /opt/registry/certs/domain.key -x509 -
days 365 -out /opt/registry/certs/domain.crt -addext "subjectAltName =
DNS:LOCAL_NODE_FQDN"
```

- 将 LOCAL_NODE_FQDN 替换为私有 registry 节点的完全限定域名。



注意

系统将提示您输入证书的相应选项。**CN=** 值是节点的主机名，它应可由 DNS 或 `/etc/hosts` 文件解析。

示例

```
[root@admin ~]# openssl req -newkey rsa:4096 -nodes -sha256 -keyout
/opt/registry/certs/domain.key -x509 -days 365 -out /opt/registry/certs/domain.crt -addext
"subjectAltName = DNS:admin.lab.redhat.com"
```



注意

在创建自签名证书时，请务必使用适当的 Subject Alternative Name(SAN) 创建证书。如果 Podman 命令需要对没有包括适当 SAN 的证书的 TLS 验证，则会返回错误：**x509: certificate relies on legacy Common Name field, use SANs or temporarily enable Common Name matching with GODEBUG=x509ignoreCN=0**

11. 创建指向 `domain.crt` 的符号链接，以允许 `skopeo` 使用文件扩展名 `.cert` 查找证书：

示例

```
[root@admin ~]# ln -s /opt/registry/certs/domain.crt /opt/registry/certs/domain.cert
```

12. 将证书添加到私有 registry 节点上的可信列表中：

语法

```
cp /opt/registry/certs/domain.crt /etc/pki/ca-trust/source/anchors/
update-ca-trust
trust list | grep -i "LOCAL_NODE_FQDN"
```

将 LOCAL_NODE_FQDN 替换为私有 registry 节点的 FQDN。

示例

```
[root@admin ~]# cp /opt/registry/certs/domain.crt /etc/pki/ca-trust/source/anchors/
```

```
[root@admin ~]# update-ca-trust
[root@admin ~]# trust list | grep -i "admin.lab.redhat.com"

label: admin.lab.redhat.com
```

13. 将证书复制到将用于访问私有 registry 的任何节点中，并更新可信列表：

示例

```
[root@admin ~]# scp /opt/registry/certs/domain.crt root@host01:/etc/pki/ca-trust/source/anchors/
[root@admin ~]# ssh root@host01
[root@host01 ~]# update-ca-trust
[root@host01 ~]# trust list | grep -i "admin.lab.redhat.com"

label: admin.lab.redhat.com
```

14. 启动本地安全私有 registry：

语法

```
podman run --restart=always --name NAME_OF_CONTAINER \
-p 5000:5000 -v /opt/registry/data:/var/lib/registry:z \
-v /opt/registry/auth:/auth:z \
-v /opt/registry/certs:/certs:z \
-e "REGISTRY_AUTH=htpasswd" \
-e "REGISTRY_AUTH_HTPASSWD_REALM=Registry Realm" \
-e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd \
-e "REGISTRY_HTTP_TLS_CERTIFICATE=/certs/domain.crt" \
-e "REGISTRY_HTTP_TLS_KEY=/certs/domain.key" \
-e REGISTRY_COMPATIBILITY_SCHEMA1_ENABLED=true \
-d registry:2
```

将 NAME_OF_CONTAINER 替换为分配给容器的名称。

示例

```
[root@admin ~]# podman run --restart=always --name myprivateregistry \
-p 5000:5000 -v /opt/registry/data:/var/lib/registry:z \
-v /opt/registry/auth:/auth:z \
-v /opt/registry/certs:/certs:z \
-e "REGISTRY_AUTH=htpasswd" \
-e "REGISTRY_AUTH_HTPASSWD_REALM=Registry Realm" \
-e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd \
-e "REGISTRY_HTTP_TLS_CERTIFICATE=/certs/domain.crt" \
-e "REGISTRY_HTTP_TLS_KEY=/certs/domain.key" \
-e REGISTRY_COMPATIBILITY_SCHEMA1_ENABLED=true \
-d registry:2
```

这会在端口 5000 上启动私有 registry，并在运行 registry 的容器中挂载 registry 目录的卷。

15. 在本地 registry 节点上，验证 **registry.redhat.io** 是否在容器 registry 搜索路径中。

- a. 打开并编辑 `/etc/containers/registries.conf` 文件，如果不存在，请将 `registry.redhat.io` 添加到 `unqualified-search-registries` 列表中：

示例

```
unqualified-search-registries = ["registry.redhat.io", "registry.access.redhat.com",
"registry.fedoraproject.org", "registry.centos.org", "docker.io"]
```

16. 使用您的红帽客户门户网站凭证登录到 `registry.redhat.io`：

语法

```
podman login registry.redhat.io
```

17. 将以下 Red Hat Ceph Storage 7 镜像、Prometheus 镜像和 Dashboard 镜像从红帽客户门户网站复制到私有 registry：

表 3.1. 监控堆栈的自定义镜像详情

监控堆栈组件	镜像详情
Prometheus	registry.redhat.io/openshift4/ose-prometheus:v4.12
Grafana	registry.redhat.io/rhceph/grafana-rhel9:latest
Node-exporter	registry.redhat.io/openshift4/ose-prometheus-node-exporter:v4.12
AlertManager	registry.redhat.io/openshift4/ose-prometheus-alertmanager:v4.12
HAProxy	registry.redhat.io/rhceph/rhceph-haproxy-rhel9:latest
Keepalived	registry.redhat.io/rhceph/keepalived-rhel9:latest
SNMP Gateway	registry.redhat.io/rhceph/snmp-notifier-rhel9:latest

语法

```
podman run -v /CERTIFICATE_DIRECTORY_PATH:/certs:Z -v
/CERTIFICATE_DIRECTORY_PATH/domain.cert:/certs/domain.cert:Z --rm
registry.redhat.io/rhel9/skopeo:8.5-8 skopeo copy --remove-signatures --src-creds
RED_HAT_CUSTOMER_PORTAL_LOGIN:RED_HAT_CUSTOMER_PORTAL_PASSWORD
--dest-cert-dir=./certs/ --dest-creds
PRIVATE_REGISTRY_USERNAME:PRIVATE_REGISTRY_PASSWORD
docker://registry.redhat.io/SRC_IMAGE:SRC_TAG
docker://LOCAL_NODE_FQDN:5000/DST_IMAGE:DST_TAG
```

- 将 `CERTIFICATE_DIRECTORY_PATH` 替换为自签名证书的目录路径。

- 将 `RED_HAT_CUSTOMER_PORTAL_LOGIN` 和 `RED_HAT_CUSTOMER_PORTAL_PASSWORD` 替换为您的红帽客户门户网站凭证。
- 将 `PRIVATE_REGISTRY_USERNAME` 和 `PRIVATE_REGISTRY_PASSWORD` 替换为私有 registry 凭证。
- 将 `SRC_IMAGE` 和 `SRC_TAG` 替换为要从 `registry.redhat.io` 中复制的镜像的名称和标签。
- 将 `DST_IMAGE` 和 `DST_TAG` 替换为要复制到私有 registry 的镜像的名称和标签。
- 将 `LOCAL_NODE_FQDN` 替换为私有 registry 的 FQDN。

示例

```
[root@admin ~]# podman run -v /opt/registry/certs:/certs:Z -v
/opt/registry/certs/domain.cert:/certs/domain.cert:Z --rm registry.redhat.io/rhel9/skopeo
skopeo copy --remove-signatures --src-creds myusername:mypassword1 --dest-cert-
dir=./certs/ --dest-creds myregistryusername:myregistrypassword1
docker://registry.redhat.io/rhceph/rhceph-7-rhel9:latest
docker://admin.lab.redhat.com:5000/rhceph/rhceph-7-rhel9:latest
```

```
[root@admin ~]# podman run -v /opt/registry/certs:/certs:Z -v
/opt/registry/certs/domain.cert:/certs/domain.cert:Z --rm registry.redhat.io/rhel9/skopeo
skopeo copy --remove-signatures --src-creds myusername:mypassword1 --dest-cert-
dir=./certs/ --dest-creds myregistryusername:myregistrypassword1
docker://registry.redhat.io/openshift4/ose-prometheus-node-exporter:v4.12
docker://admin.lab.redhat.com:5000/openshift4/ose-prometheus-node-exporter:v4.12
```

```
[root@admin ~]# podman run -v /opt/registry/certs:/certs:Z -v
/opt/registry/certs/domain.cert:/certs/domain.cert:Z --rm registry.redhat.io/rhel9/skopeo
skopeo copy --remove-signatures --src-creds myusername:mypassword1 --dest-cert-
dir=./certs/ --dest-creds myregistryusername:myregistrypassword1
docker://registry.redhat.io/rhceph/grafana-rhel9:latest
docker://admin.lab.redhat.com:5000/rhceph/grafana-rhel9:latest
```

```
[root@admin ~]# podman run -v /opt/registry/certs:/certs:Z -v
/opt/registry/certs/domain.cert:/certs/domain.cert:Z --rm registry.redhat.io/rhel9/skopeo
skopeo copy --remove-signatures --src-creds myusername:mypassword1 --dest-cert-
dir=./certs/ --dest-creds myregistryusername:myregistrypassword1
docker://registry.redhat.io/openshift4/ose-prometheus:v4.12
docker://admin.lab.redhat.com:5000/openshift4/ose-prometheus:v4.12
```

```
[root@admin ~]# podman run -v /opt/registry/certs:/certs:Z -v
/opt/registry/certs/domain.cert:/certs/domain.cert:Z --rm registry.redhat.io/rhel9/skopeo
skopeo copy --remove-signatures --src-creds myusername:mypassword1 --dest-cert-
dir=./certs/ --dest-creds myregistryusername:myregistrypassword1
docker://registry.redhat.io/openshift4/ose-prometheus-alertmanager:v4.12
docker://admin.lab.redhat.com:5000/openshift4/ose-prometheus-alertmanager:v4.12
```

18. 使用 `curl` 命令，验证镜像是否驻留在本地 registry 中：

语法

```
curl -u PRIVATE_REGISTRY_USERNAME:PRIVATE_REGISTRY_PASSWORD
https://LOCAL_NODE_FQDN:5000/v2/_catalog
```

示例

```
[root@admin ~]# curl -u myregistryusername:myregistrypassword1
https://admin.lab.redhat.com:5000/v2/_catalog

{"repositories":["openshift4/ose-prometheus","openshift4/ose-prometheus-
alertmanager","openshift4/ose-prometheus-node-exporter","rhceph/rhceph-7-dashboard-
rhel9","rhceph/rhceph-7-rhel9"]}
```

其它资源

- 有关不同镜像 Ceph 软件包版本的更多信息，请参阅知识库文章解决方案来了解[什么是 Red Hat Ceph Storage 版本和对应的 Ceph 软件包版本？](#)

3.10.7. 为断开连接的安装运行 preflight playbook

您可以使用 **cephadm-preflight.yml** Ansible playbook 来配置 Ceph 存储库并准备存储集群以进行引导。它还会安装一些前提条件，如 **podman**、**lvm2**、**chronyd** 和 **cephadm**。

preflight playbook 使用 **cephadm-ansible** 清单 **hosts** 文件来识别存储集群中所有节点。**cephadm-ansible**、**cephadm-preflight.yml** 和清单 **hosts** 文件的默认位置是 **/usr/share/cephadm-ansible/**。

以下示例显示了典型的清单文件的结构：

示例

```
host02
host03
host04

[admin]
host01
```

清单文件中的 **[admin]** 组包含存储了 admin 密钥环的节点的名称。



注意

在引导初始主机前运行 preflight playbook。

先决条件

- **cephadm-ansible** 软件包安装在 Ansible 管理节点上。
- 对存储集群中所有节点的根级别访问权限。
- 在存储集群的所有主机上设置免密码 **ssh**。
- 配置为访问启用了以下软件仓库的本地 YUM 存储库服务器：
 - **rhel-9-for-x86_64-baseos-rpms**
 - **rhel-9-for-x86_64-appstream-rpms**
 - **rhceph-7-tools-for-rhel-9-x86_64-rpms**



注意

有关设置本地 YUM 存储库的更多信息，请参阅[创建本地仓库并与 Disconnected/Offline/Air-gapped 系统共享](#)

流程

1. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录。
2. 打开并编辑 `hosts` 文件并添加节点。
3. 在运行 `preflight` playbook 时使用设置为 `custom` 的 `ceph_origin` 参数，以使用本地 YUM 存储库：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars
"ceph_origin=custom" -e "custom_repo_url=CUSTOM_REPO_URL"
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --
extra-vars "ceph_origin=custom" -e
"custom_repo_url=http://mycustomrepo.lab.redhat.com/x86_64/os/"
```

安装完成后，`cephadm` 驻留在 `/usr/sbin/` 目录中。



注意

使用 Ansible playbook 填充 `registry.conf` 文件的内容：

语法

```
ansible-playbook -vvv -i INVENTORY_HOST_FILE cephadm-set-container-
insecure-registries.yml -e insecure_registry=REGISTRY_URL
```

示例

```
[root@admin ~]# ansible-playbook -vvv -i hosts cephadm-set-container-
insecure-registries.yml -e insecure_registry=host01:5050
```

4. 另外，您可以使用 `--limit` 选项在存储集群中选定的一组主机上运行 `preflight` playbook：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars
"ceph_origin=custom" -e "custom_repo_url=CUSTOM_REPO_URL" --limit
GROUP_NAME|NODE_NAME
```

将 `GROUP_NAME` 替换为清单文件中的组名称。将 `NODE_NAME` 替换为清单文件中的特定节点名称。

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --
extra-vars "ceph_origin=custom" -e
"custom_repo_url=http://mycustomrepo.lab.redhat.com/x86_64/os/" --limit clients
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --
extra-vars "ceph_origin=custom" -e
"custom_repo_url=http://mycustomrepo.lab.redhat.com/x86_64/os/" --limit host02
```



注意

运行 `preflight` playbook 时，**cephadm-ansible** 会自动在客户端节点上安装 **chronyd** 和 **ceph-common**。

3.10.8. 执行断开连接的安装

在执行安装前，您必须先可以从访问红帽 registry 的代理主机获取 Red Hat Ceph Storage 容器镜像，或将该镜像复制到本地 registry。



注意

如果您的本地 registry 使用带有本地 registry 的自签名证书，请确保将可信 root 证书添加到 bootstrap 主机。如需更多信息，请参阅[为断开连接的安装配置私有 registry](#)。



重要

在开始 bootstrap 过程前，请确保要使用的容器镜像与 **cephadm** 具有相同的 Red Hat Ceph Storage 版本。如果两个版本不匹配，bootstrapping 会在创建初始 **admin** 用户阶段失败。

先决条件

- 至少一个正在运行的虚拟机 (VM) 或服务器。
- 所有节点的根级别访问权限。
- 在存储集群的所有主机上设置免密码 **ssh**。
- `preflight` playbook 已在存储集群中的 bootstrap 主机上运行。如需更多信息，请参阅[为断开连接的安装运行 preflight playbook](#)。
- 配置了私有 registry，且 bootstrap 节点可以访问它。如需更多信息，请参阅[为断开连接的安装配置私有 registry](#)。
- Red Hat Ceph Storage 容器镜像位于自定义 registry 中。

流程

1. 登录到 bootstrap 主机。
2. 引导存储集群：

语法

```
cephadm --image
PRIVATE_REGISTRY_NODE_FQDN:5000/CUSTOM_IMAGE_NAME:IMAGE_TAG
bootstrap --mon-ip IP_ADDRESS --registry-url PRIVATE_REGISTRY_NODE_FQDN:5000 --
registry-username PRIVATE_REGISTRY_USERNAME --registry-password
PRIVATE_REGISTRY_PASSWORD
```

- 将 PRIVATE_REGISTRY_NODE_FQDN 替换为私有 registry 的完全限定域名。
- 将 CUSTOM_IMAGE_NAME 和 IMAGE_TAG 替换为位于私有 registry 中的 Red Hat Ceph Storage 容器镜像的名称和标签。
- 将 IP_ADDRESS 替换为您要用来运行 **cephadm bootstrap** 的节点的 IP 地址。
- 将 PRIVATE_REGISTRY_USERNAME 替换为要用于私有 registry 的用户名。
- 将 PRIVATE_REGISTRY_PASSWORD 替换为要用于私有 registry 用户名的密码。

示例

```
[root@host01 ~]# cephadm --image admin.lab.redhat.com:5000/rhceph-7-rhel9:latest
bootstrap --mon-ip 10.10.128.68 --registry-url admin.lab.redhat.com:5000 --registry-
username myregistryusername --registry-password myregistrypassword1
```

完成该脚本需要几分钟时间。脚本完成后，会提供 Red Hat Ceph Storage Dashboard URL 提供凭据、用于访问 Ceph 命令行界面 (CLI) 的命令，以及启用遥测的请求。

Ceph Dashboard is now available at:

```
URL: https://host01:8443/
User: admin
Password: i8nhu7zham
```

Enabling client.admin keyring and conf on hosts with "admin" label
You can access the Ceph CLI with:

```
sudo /usr/sbin/cephadm shell --fsid 266ee7a8-2a05-11eb-b846-5254002d4916 -c
/etc/ceph/ceph.conf -k /etc/ceph/ceph.client.admin.keyring
```

Please consider enabling telemetry to help improve Ceph:

```
ceph telemetry on
```

For more information see:

```
https://docs.ceph.com/docs/master/mgr/telemetry/
```

Bootstrap complete.

bootstrap 过程完成后，请参阅[为断开连接的安装更改自定义容器镜像的配置](#)来配置容器镜像。

其它资源

- 在您的存储集群启动并运行后，请参阅[Red Hat Ceph Storage Operations Guide](#)以了解有关配置其他守护进程和服务的更多信息。

3.10.9. 为断开连接的安装更改自定义容器镜像配置

为断开连接的节点执行初始 bootstrap 后，您必须为监控堆栈守护进程指定自定义容器镜像。您可以覆盖用于监控堆栈守护进程的默认容器镜像，因为节点无法访问默认的容器 Registry。



注意

在进行任何配置更改前，请确保初始主机上的 bootstrap 过程已完成。

默认情况下，监控堆栈组件根据主 Ceph 镜像进行部署。对于存储集群的断开连接的环境，您可以使用最新的监控堆栈组件镜像。



注意

使用自定义 registry 时，请务必在添加任何 Ceph 守护进程前登录到新添加的节点上的自定义 registry。

语法

```
# ceph cephadm registry-login --registry-url CUSTOM_REGISTRY_NAME --
registry_username REGISTRY_USERNAME --registry_password
REGISTRY_PASSWORD
```

示例

```
# ceph cephadm registry-login --registry-url myregistry --registry_username
myregistryusername --registry_password myregistrypassword1
```

先决条件

- 至少一个正在运行的虚拟机 (VM) 或服务器。
- Red Hat Enterprise Linux 8.10 或 9.4，将 **ansible-core** 捆绑到 AppStream 中。
- 所有节点的根级别访问权限。
- 在存储集群的所有主机上设置免密码 **ssh**。

流程

1. 使用 **ceph config** 命令设置自定义容器镜像：

语法

```
ceph config set mgr mgr/cephadm/OPTION_NAME
CUSTOM_REGISTRY_NAME/CONTAINER_NAME
```

在 OPTION_NAME 中使用以下选项：

```
container_image_prometheus
container_image_grafana
container_image_alertmanager
container_image_node_exporter
```

示例

```
[root@host01 ~]# ceph config set mgr mgr/cephadm/container_image_prometheus
myregistry/mycontainer
[root@host01 ~]# ceph config set mgr mgr/cephadm/container_image_grafana
myregistry/mycontainer
[root@host01 ~]# ceph config set mgr mgr/cephadm/container_image_alertmanager
myregistry/mycontainer
[root@host01 ~]# ceph config set mgr mgr/cephadm/container_image_node_exporter
myregistry/mycontainer
```

2. 重新部署 **node-exporter** :

语法

```
ceph orch redeploy node-exporter
```



注意

如果有任何服务没有部署，您可以使用 **ceph orch redeploy** 命令重新部署这些服务。



注意

通过设置自定义镜像，配置镜像名称和标签的默认值将被覆盖，但不能覆盖。当更新可用时，默认值会改变。通过设置自定义镜像，您将无法配置已为其设置自定义镜像以进行自动更新的组件。您需要手动更新配置镜像名称和标签，以便能安装更新。

- 如果选择恢复为使用默认配置，可以重置自定义容器镜像。使用 **ceph config rm** 重置配置选项：

语法

```
ceph config rm mgr mgr/cephadm/OPTION_NAME
```

示例

```
ceph config rm mgr mgr/cephadm/container_image_prometheus
```

其它资源

- 有关执行断开连接的安装的详情，请参考 [执行断开连接的安装](#)。

3.11. 分发 SSH 密钥

您可以使用 **cephadm-distribute-ssh-key.yml** playbook 来分发 SSH 密钥，而不是手动创建和分发密钥。playbook 通过清单中的所有主机分发 SSH 公钥。

您还可以在 Ansible 管理节点上生成 SSH 密钥对，并将公钥分发到存储集群中的每个节点，以便 Ansible 可以在不提示输入密码的情况下访问节点。

先决条件

- Ansible 安装在管理节点上。
- 访问 Ansible 管理节点。
- 具有 `sudo` 对存储集群中所有节点的访问权限的 Ansible 用户。
- bootstrap 已完成。请参阅 [Red Hat Ceph Storage 安装指南中的引导新存储集群](#) 部分。

流程

1. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ansible@admin ~]$ cd /usr/share/cephadm-ansible
```

2. 在 Ansible 管理节点中，分发 SSH 密钥。可选的 `cephadm_pubkey_path` 参数是 ansible 控制器主机上的 SSH 公钥文件的完整路径名称。



注意

如果没有指定 `cephadm_pubkey_path`，则 playbook 从 `cephadm get-pub-key` 命令获取密钥。这意味着您至少已引导一个最小集群。

语法

```
ansible-playbook -i INVENTORY_HOST_FILE cephadm-distribute-ssh-key.yml -e
cephadm_ssh_user=USER_NAME -e cephadm_pubkey_path= home/cephadm/ceph.key -e
admin_node=ADMIN_NODE_NAME_1
```

示例

```
[ansible@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-distribute-ssh-
key.yml -e cephadm_ssh_user=ceph-admin -e
cephadm_pubkey_path=/home/cephadm/ceph.key -e admin_node=host01
```

```
[ansible@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-distribute-ssh-
key.yml -e cephadm_ssh_user=ceph-admin -e admin_node=host01
```

3.12. 启动 CEPHADM SHELL

`cephadm shell` 命令在容器中打开一个 `bash` shell，并安装了所有 Ceph 软件包。使用 shell 运行 "Day One" 集群设置任务，如安装和引导，以及运行 `ceph` 命令。



注意

如果节点包含 `/etc/ceph/` 中的配置和密钥环文件，则容器环境将使用这些文件中的值作为 `cephadm shell` 的默认值。如果在 MON 节点上执行 `cephadm shell`，`cephadm shell` 会继承 MON 容器的默认配置，而不使用默认配置。

先决条件

- 已安装并引导的存储集群。
- 对存储集群中所有节点的根级别访问权限。

流程

使用以下方法之一打开 **cephadm shell** :

- 在系统提示符处输入 **cephadm shell**。本例从 shell 中运行 **ceph -s** 命令。

示例

```
[root@host01 ~]# cephadm shell
[ceph: root@host01 /]# ceph -s
```

- 在系统提示符处，输入 **cephadm shell** 和您要运行的命令：

示例

```
[root@host01 ~]# cephadm shell ceph -s
```



注意

要退出 **cephadm shell**，请使用 **exit** 命令。

```
[ceph: root@host01 /]# exit
[root@host01 ~]#
```

3.13. 验证集群安装

集群安装完成后，您可以验证 Red Hat Ceph Storage 7 安装是否正确运行。

以 root 用户身份验证存储集群安装的方法：

- 运行 **podman ps** 命令。
- 运行 **cephadm shell ceph -s**。

先决条件

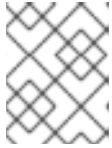
- 对存储集群中所有节点的根级别访问权限。

流程

- 运行 **podman ps** 命令：

示例

```
[root@host01 ~]# podman ps
```

**注意**

在 Red Hat Ceph Storage 7 中，**systemd** 单元的格式已更改。在 **NAMES** 列中，单元文件现在包含 **FSID**。

- 运行 **cephadm shell ceph -s** 命令：

示例

```
[root@host01 ~]# cephadm shell ceph -s

cluster:
  id: f64f341c-655d-11eb-8778-fa163e914bcc
  health: HEALTH_OK

services:
  mon: 3 daemons, quorum host01,host02,host03 (age 94m)
  mgr: host01.lbnhug(active, since 59m), standbys: host02.rofgay, host03.ohipra
  mds: 1/1 daemons up, 1 standby
  osd: 18 osds: 18 up (since 10m), 18 in (since 10m)
  rgw: 4 daemons active (2 hosts, 1 zones)

data:
  volumes: 1/1 healthy
  pools: 8 pools, 225 pgs
  objects: 230 objects, 9.9 KiB
  usage: 271 MiB used, 269 GiB / 270 GiB avail
  pgs: 225 active+clean

io:
  client: 85 B/s rd, 0 op/s rd, 0 op/s wr
```

**注意**

存储集群的运行状况处于 **HEALTH_WARN** 状态，因为主机不会添加守护进程。

3.14. 添加主机

启动 Red Hat Ceph Storage 安装会创建一个可正常工作的存储集群，该集群由同一容器中的一个 monitor 守护进程和一个管理器守护进程组成。作为存储管理员，您可以向存储集群添加额外的主机并进行配置。



注意

- 运行 preflight playbook 将 **podman**、**lvm2**、**chronyd** 和 **cephadm** 安装到 Ansible 清单文件中列出的所有主机上。
- 使用自定义 registry 时，请务必在添加任何 Ceph 守护进程前登录到新添加的节点上的自定义 registry。

```
.Syntax
[source,subs="verbatim,quotes"]
----
# ceph cephadm registry-login --registry-url _CUSTOM_REGISTRY_NAME_
--registry_username _REGISTRY_USERNAME_ --registry_password
_REGISTRY_PASSWORD_
----
```

```
.Example
----
# ceph cephadm registry-login --registry-url myregistry --registry_username
myregistryusername --registry_password myregistrypassword1
----
```

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- 对存储集群中所有节点具有 sudo 访问权限的 root 级别或用户。
- 将节点注册到 CDN 并附加订阅。
- 具有 sudo 的 Ansible 用户，对存储集群中所有节点的 ssh 访问和免密码访问。

流程

+



注意

在以下步骤中，按指示使用 **root** 用户，或使用用户启动的用户名。

1. 从包含 admin 密钥环的节点，在新主机的 root 用户的 **authorized_keys** 文件中安装存储集群的公共 SSH 密钥：

语法

```
ssh-copy-id -f -i /etc/ceph/ceph.pub user@NEWHOST
```

示例

```
[root@host01 ~]# ssh-copy-id -f -i /etc/ceph/ceph.pub root@host02
[root@host01 ~]# ssh-copy-id -f -i /etc/ceph/ceph.pub root@host03
```

2. 进入 Ansible 管理节点上的 **/usr/share/cephadm-ansible** 目录。

— 10

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

- 从 Ansible 管理节点，将新主机添加到 Ansible 清单文件。该文件的默认位置为 `/usr/share/cephadm-ansible/hosts`。以下示例显示了典型的清单文件的结构：

示例

```
[ceph-admin@admin ~]$ cat hosts
```

```
host02
host03
host04
```

```
[admin]
host01
```

**注意**

如果您之前已将新主机添加到 Ansible 清单文件，并在主机上运行 preflight playbook，请跳至第 4 步。

- 使用 `--limit` 选项运行 preflight playbook：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars "ceph_origin=rhcs"
--limit NEWHOST
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --
extra-vars "ceph_origin=rhcs" --limit host02
```

preflight playbook 在新主机上安装 **podman**、**lvm2**、**chronyd** 和 **cephadm**。安装完成后，**cephadm** 驻留在 `/usr/sbin/` 目录中。

- 从 bootstrap 节点，使用 **cephadm** 编排器将新主机添加到存储集群中：

语法

```
ceph orch host add NEWHOST
```

示例

```
[ceph: root@host01 /]# ceph orch host add host02
Added host 'host02' with addr '10.10.128.69'
[ceph: root@host01 /]# ceph orch host add host03
Added host 'host03' with addr '10.10.128.70'
```

6. 可选：您还可以在运行 preflight playbook 前和之后按 IP 地址添加节点。如果您在存储集群环境中没有配置 DNS，您可以根据 IP 地址添加主机，以及主机名。

语法

```
ceph orch host add HOSTNAME IP_ADDRESS
```

示例

```
[ceph: root@host01 /]# ceph orch host add host02 10.10.128.69
Added host 'host02' with addr '10.10.128.69'
```

验证

- 查看存储集群的状态，并验证是否已添加新主机。主机 STATUS 为空，在 `ceph orch host ls` 命令的输出中。

示例

```
[ceph: root@host01 /]# ceph orch host ls
```

其它资源

- 请参阅 [Red Hat Ceph Storage 安装指南中的将 Red Hat Ceph Storage 节点注册到 CDN 并附加订阅部分](#)。
- 请参阅 [Red Hat Ceph Storage 安装指南中的创建带有 sudo 访问权限的 Ansible 用户部分](#)。

3.14.1. 使用 `addr` 选项来识别主机

`addr` 选项提供了联系主机的其他方法。将主机的 IP 地址添加到 `addr` 选项。如果 `ssh` 无法通过其主机名连接到主机，则它将使用 `addr` 中存储的值通过 IP 地址访问主机。

先决条件

- 已安装并引导的存储集群。
- 对存储集群中所有节点的根级别访问权限。

流程

从 `cephadm` shell 内运行此步骤。

- 添加 IP 地址：

语法

```
ceph orch host add HOSTNAME IP_ADDR
```

示例

```
[ceph: root@host01 /]# ceph orch host add host01 10.10.128.68
```



注意

如果通过主机名添加主机会导致该主机使用 IPv6 地址而不是 IPv4 地址添加，请使用 `ceph orch host` 来指定该主机的 IP 地址：

```
ceph orch host set-addr HOSTNAME IP_ADDR
```

要将添加主机的 IPv6 格式的 IP 地址转换为 IPv4 格式，请使用以下命令：

```
ceph orch host set-addr HOSTNAME IPV4_ADDRESS
```

3.14.2. 添加多个主机

使用 YAML 文件同时将多个主机添加到存储集群。



注意

务必在本地主机上创建 `hosts.yaml` 文件，或者在本地主机上创建文件，然后使用 `cephadm shell` 在容器内挂载文件。`cephadm shell` 会自动将挂载的文件放置在 `/mnt` 中。如果您直接在本地主机上创建该文件，然后应用 `hosts.yaml` 文件而不是挂载它，您可能会看到 `File does not exist` 错误。

先决条件

- 已安装并引导的存储集群。
- 对存储集群中所有节点的根级别访问权限。

流程

1. 将公共 `ssh` 密钥复制到您要添加的每个主机。
2. 使用文本编辑器创建 `hosts.yaml` 文件。
3. 将主机描述添加到 `hosts.yaml` 文件中，如下例所示。包含标签，以标识您要在每个主机上部署的守护进程的放置。使用三个短划线(`---`)分隔每个主机描述。

示例

```
service_type: host
addr:
hostname: host02
labels:
- mon
- osd
- mgr
---
service_type: host
addr:
hostname: host03
labels:
- mon
- osd
- mgr
```

```

---
service_type: host
addr:
hostname: host04
labels:
- mon
- osd

```

4. 如果在主机容器中创建了 **hosts.yaml** 文件，请调用 **ceph orch apply** 命令：

示例

```

[root@host01 ~]# ceph orch apply -i hosts.yaml
Added host 'host02' with addr '10.10.128.69'
Added host 'host03' with addr '10.10.128.70'
Added host 'host04' with addr '10.10.128.71'

```

5. 如果您直接在本地主机上创建了 **hosts.yaml** 文件，请使用 **cephadm shell** 来挂载该文件：

示例

```

[root@host01 ~]# cephadm shell --mount hosts.yaml -- ceph orch apply -i /mnt/hosts.yaml

```

6. 查看主机及其标签列表：

示例

```

[root@host01 ~]# ceph orch host ls
HOST  ADDR  LABELS  STATUS
host02 host02  mon osd mgr
host03 host03  mon osd mgr
host04 host04  mon osd

```



注意

如果主机在线且正常运行，则其状态为空。脱机主机显示 OFFLINE 状态，处于维护模式的主机则显示 MAINTENANCE 状态。

3.14.3. 在断开连接的部署中添加主机

如果您在专用网络上运行存储集群，且无法通过专用 IP 访问您的主机名 (DNS)，您必须同时包含您要添加到存储集群的每个主机的主机名和 IP 地址。

先决条件

- 正在运行的存储集群。
- 对存储集群中所有主机的根级别访问权限。

流程

- 调用 **cephadm shell**。

语法

```
[root@host01 ~]# cephadm shell
```

2. 添加主机：

语法

```
ceph orch host add HOST_NAME HOST_ADDRESS
```

示例

```
[ceph: root@host01 /]# ceph orch host add host03 10.10.128.70
```

3.14.4. 删除主机

您可以使用 Ceph 编排器删除 Ceph 集群的主机。所有守护进程都会使用 **drain** 选项删除，该选项添加了 **_no_schedule** 标签，以确保您无法部署任何守护进程或集群完成这个操作。



重要

如果您要删除 bootstrap 主机，请确保在删除主机前将 admin 密钥环和配置文件复制到存储集群中的另一主机上。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- 所有节点的根级别访问权限。
- 主机添加到存储集群中。
- 部署所有服务。
- Cephadm 部署在必须移除服务的节点上。

流程

1. 登录到 Cephadm shell：

示例

```
[root@host01 ~]# cephadm shell
```

2. 获取主机详情：

示例

```
[ceph: root@host01 /]# ceph orch host ls
```

3. 排空主机中的所有守护进程：

语法

```
ceph orch host drain HOSTNAME
```

示例

```
[ceph: root@host01 /]# ceph orch host drain host02
```

`_no_schedule` 标签自动应用到阻止部署的主机。

4. 检查移除 OSD 的状态：

示例

```
[ceph: root@host01 /]# ceph orch osd rm status
```

当 OSD 上没有剩余的放置组(PG)时，该 OSD 会停用并从存储集群中移除。

5. 检查所有守护进程是否已从存储集群中移除：

语法

```
ceph orch ps HOSTNAME
```

示例

```
[ceph: root@host01 /]# ceph orch ps host02
```

6. 删除主机：

语法

```
ceph orch host rm HOSTNAME
```

示例

```
[ceph: root@host01 /]# ceph orch host rm host02
```

其它资源

- 如需更多信息，请参阅 Red Hat Ceph Storage Operations Guide 中的[使用 Ceph Orchestrator 添加主机](#)的内容。
- 如需更多信息，请参阅 Red Hat Ceph Storage Operations Guide 中的[使用 Ceph Orchestrator 列出主机](#)的内容。

3.15. 标记主机

Ceph 编配器支持将标签分配到主机。标签是自由格式的，没有具体含义。这意味着，您可以使用 `mon`、`monitor`、`mycluster_monitor` 或任何其他文本字符串。每一主机可以有多个标签。

例如，将 **mon** 标签应用到您要在其上部署 Ceph 监控守护进程的所有主机，**mgr** 适用于您要在其上部署 Ceph Manager 守护进程、**rgw** 用于 Ceph 对象网关守护进程等。

标记存储集群中的所有主机有助于简化系统管理任务，允许您快速识别每个主机上运行的守护进程。此外，您可以使用 Ceph 编排器或 YAML 文件在具有特定主机标签的主机上部署或删除守护进程。

3.15.1. 为主机添加标签

使用 Ceph 编排器向主机添加标签。标签可用于指定守护进程的放置。

几个标签示例是 **mgr**、**mon** 和 **osd**，基于主机上部署的服务。每一主机可以有多个标签。

您还可以添加以下主机标签，它们对 **cephadm** 具有特殊含义，它们以 **_** 开头：

- **_no_schedule**: 此标签会阻止 **cephadm** 调度或部署主机上的守护进程。如果它被添加到已包含 Ceph 守护进程的现有主机中，它会导致 **cephadm** 在其他位置移动这些守护进程，除了自动移除的 OSD 除外。当添加带有 **_no_schedule** 标签的主机时，不会在其上部署守护进程。当守护进程在删除主机前排空时，在该主机上设置了 **_no_schedule** 标签。
- **_no_autotune_memory** : 此标签不会在主机上自动微调内存。即使为 host 上的一个或多个守护进程启用了 **osd_memory_target_autotune** 选项，也会防止守护进程内存被调优。
- **_admin**: 默认情况下，**_admin** 标签应用于存储集群中的 bootstrapped 主机，**client.admin** 密钥被设置为使用 **ceph orch client-keyring {ls|set|rm}** 功能分发到该主机。将此标签添加到其他主机通常会导致 **cephadm** 在 **/etc/ceph** 目录中部署配置和密钥环文件。

先决条件

- 已安装并引导的存储集群。
- 对存储集群中所有节点的根级别访问权限。
- 主机添加到存储集群中。

流程

1. 登录到 Cephadm shell :

示例

```
[root@host01 ~]# cephadm shell
```

2. 为主机添加标签：

语法

```
ceph orch host label add HOSTNAME LABEL
```

示例

```
[ceph: root@host01 /]# ceph orch host label add host02 mon
```

验证

- 列出主机：

示例

```
[ceph: root@host01 /]# ceph orch host ls
```

3.15.2. 从主机中删除标签

您可以使用 Ceph 编配器从主机移除标签。

先决条件

- 已安装并引导的存储集群。
- 对存储集群中所有节点的根级别访问权限。

流程

1. 启动 **cephadm** shell：

```
[root@host01 ~]# cephadm shell
[ceph: root@host01 /]#
```

2. 删除标签。

语法

```
ceph orch host label rm HOSTNAME LABEL
```

示例

```
[ceph: root@host01 /]# ceph orch host label rm host02 mon
```

验证

- 列出主机：

示例

```
[ceph: root@host01 /]# ceph orch host ls
```

3.15.3. 使用主机标签在特定主机上部署守护进程

您可以使用主机标签将守护进程部署到特定的主机上。使用主机标签在特定主机上部署守护进程的方法有两种：

- 从命令行使用 **--placement** 选项。
- 通过使用 YAML 文件。

先决条件

- 已安装并引导的存储集群。
- 对存储集群中所有节点的根级别访问权限。

流程

1. 登录到 Cephadm shell :

示例

```
[root@host01 ~]# cephadm shell
```

2. 列出当前的主机和标签 :

示例

```
[ceph: root@host01 /]# ceph orch host ls
HOST  ADDR  LABELS  STATUS
host01      _admin mon osd mgr
host02      mon osd mgr mylabel
```

- **方法 1** : 使用 `--placement` 选项从命令行部署守护进程 :

语法

```
ceph orch apply DAEMON --placement="label:LABEL"
```

示例

```
[ceph: root@host01 /]# ceph orch apply prometheus --placement="label:mylabel"
```

- **方法 2** 将守护进程分配给 YAML 文件中的特定主机标签, 请在 YAML 文件中指定服务类型和标签 :
 - a. 创建 `placement.yml` 文件 :

示例

```
[ceph: root@host01 /]# vi placement.yml
```

- b. 在 `placement.yml` 文件中指定服务类型和标签 :

示例

```
service_type: prometheus
placement:
  label: "mylabel"
```

- c. 应用守护进程放置文件 :

语法

```
ceph orch apply -i FILENAME
```

示例

```
[ceph: root@host01 /]# ceph orch apply -i placement.yml
Scheduled prometheus update...
```

验证

- 列出守护进程的状态：

语法

```
ceph orch ps --daemon_type=DAEMON_NAME
```

示例

```
[ceph: root@host01 /]# ceph orch ps --daemon_type=prometheus
NAME          HOST PORTS STATUS   REFRESHED AGE MEM USE MEM LIM
VERSION IMAGE ID CONTAINER ID
prometheus.host02 host02 *:9095 running (2h) 8m ago 2h 85.3M - 2.22.2
ac25aac5d567 ad8c7593d7c0
```

3.16. 添加 MONITOR 服务

典型的 Red Hat Ceph Storage 集群在不同主机上部署了三个或五个 monitor 守护进程。如果您的存储集群有五个或更多主机，红帽建议您部署五个 monitor 节点。



注意

如果使用防火墙，请参阅 Red Hat Ceph Storage 配置指南中的 [Ceph Monitor 节点的防火墙设置部分](#)。



注意

bootstrap 节点是存储集群的初始监控器。确保将 bootstrap 节点包含在要部署的主机列表中。



注意

如果要将 monitor 服务应用到多个特定的主机，请务必在同一 **ceph orch apply** 命令中指定所有主机名。如果您指定了 **ceph orch apply mon --placement host1**，然后指定了 **ceph orch apply mon --placement host2**，第二个命令将删除 host1 上的 monitor 服务，并将 monitor 服务应用到 host2。

如果您的 monitor 节点或整个集群都位于单个子网中，则 **cephadm** 会在向集群添加新主机时自动添加最多五个 monitor 守护进程。**cephadm** 自动配置新主机上的 monitor 守护进程。新主机与存储集群中的第一个（引导）主机位于同一个子网中。**cephadm** 还可以部署和缩放 monitor，以响应存储集群大小的变化。

先决条件

- 对存储集群中所有主机的根级别访问权限。
- 正在运行的存储群集。

流程

1. 将五个 monitor 守护进程应用到存储集群中的五个随机主机：

```
ceph orch apply mon 5
```

2. 禁用自动监控器部署：

```
ceph orch apply mon --unmanaged
```

3.16.1. 将 monitor 节点添加到特定主机

使用主机标签标识包含 monitor 节点的主机。

先决条件

- 对存储集群中所有节点的根级别访问权限。
- 正在运行的存储群集。

流程

1. 为主机分配 **mon** 标签：

语法

```
ceph orch host label add HOSTNAME mon
```

示例

```
[ceph: root@host01 /]# ceph orch host label add host01 mon
```

2. 查看当前的主机和标签：

语法

```
ceph orch host ls
```

示例

```
[ceph: root@host01 /]# ceph orch host label add host02 mon
[ceph: root@host01 /]# ceph orch host label add host03 mon
[ceph: root@host01 /]# ceph orch host ls
HOST ADDR LABELS STATUS
host01    mon
host02    mon
host03    mon
```

```
host04
host05
host06
```

3. 根据主机标签部署 monitor :

语法

```
ceph orch apply mon label:mon
```

4. 在特定的一组主机上部署 monitor:

语法

```
ceph orch apply mon HOSTNAME1,HOSTNAME2,HOSTNAME3
```

示例

```
[root@host01 ~]# ceph orch apply mon host01,host02,host03
```



注意

确保将 bootstrap 节点包含在要部署的主机列表中。

3.17. 在现有集群中设置自定义 SSH 密钥

作为存储管理员，使用 Cephadm，您可以使用 SSH 密钥与远程主机安全地进行身份验证。SSH 密钥存储在 monitor 中，以连接到远程主机。

当引导集群时，会自动生成此 SSH 密钥，且不需要额外的配置。但是，您可以使用 **ceph cephadm generate-key** 命令生成新的 SSH 密钥。

先决条件

- Ansible 管理节点。
- 对 Ansible 管理节点的根级别访问权限。
- **cephadm-ansible** 软件包安装在节点上。

流程

1. 前往 **cephadm-ansible** 目录。
2. 生成一个新的 SSH 密钥：

示例

```
[ceph-admin@admin cephadm-ansible]$ ceph cephadm generate-key
```

3. 检索 SSH 密钥的公共部分：

示例

```
[ceph-admin@admin cephadm-ansible]$ ceph cephadm get-pub-key
```

4. 删除当前存储的 SSH 密钥：

示例

```
[ceph-admin@admin cephadm-ansible]$ ceph cephadm clear-key
```

5. 重启 mgr 守护进程以重新载入配置：

示例

```
[ceph-admin@admin cephadm-ansible]$ ceph mgr fail
```

3.17.1. 配置不同的 SSH 用户

作为存储管理员，您可以配置一个非 root SSH 用户，该用户可登录具有足够特权的所有 Ceph 集群节点，以下载容器镜像、启动容器，并在不提示输入密码的情况下执行命令。



重要

在配置非 root SSH 用户之前，需要将集群 SSH 密钥添加到用户的 **authorized_keys** 文件中，非 root 用户必须具有免密码 sudo 访问权限。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- Ansible 管理节点。
- 对 Ansible 管理节点的根级别访问权限。
- **cephadm-ansible** 软件包安装在节点上。
- 将集群 SSH 密钥添加到用户的 **authorized_keys**。
- 为非 root 用户启用免密码 sudo 访问权限。

流程

1. 前往 **cephadm-ansible** 目录。
2. 为 Cephadm 提供要执行所有 Cephadm 操作的用户名称：

语法

```
[ceph-admin@admin cephadm-ansible]$ ceph cephadm set-user <user>
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ceph cephadm set-user user
```


- 3. 检索 SSH 公钥。

语法

```
ceph cephadm get-pub-key > ~/ceph.pub
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ceph cephadm get-pub-key > ~/ceph.pub
```

- 4. 将 SSH 密钥复制到所有主机。

语法

```
ssh-copy-id -f -i ~/ceph.pub USER@HOST
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ssh-copy-id ceph-admin@host01
```

3.18. 设置管理节点

使用管理节点来管理存储集群。

管理节点同时包含集群配置文件和 admin 密钥环。这两个文件都存储在 `/etc/ceph` 目录中，并使用存储集群的名称作为前缀。

例如，默认的 ceph 集群名称是 `ceph`。在使用默认名称的集群中，管理员密钥环名为 `/etc/ceph/ceph.client.admin.keyring`。对应的集群配置文件命名为 `/etc/ceph/ceph.conf`。

要以 admin 节点形式设置存储集群中的其他主机，请将 `_admin` 标签应用到您要指定为管理员节点的主机。



注意

默认情况下，在将 `_admin` 标签应用到节点后，`cephadm` 会将 `ceph.conf` 和 `client.admin` 密钥环文件复制到该节点。`_admin` 标签自动应用到 bootstrap 节点，除非使用 `cephadm bootstrap` 命令指定 `--skip-admin-label` 选项。

先决条件

- 正在运行的存储集群安装了 `cephadm`。
- 存储集群正在运行 monitor 和 Manager 节点。
- 对集群中的所有节点的根级别访问权限。

流程

1. 使用 `ceph orch host ls` 查看您的存储集群中的主机：

示例

```
[root@host01 ~]# ceph orch host ls
HOST ADDR LABELS STATUS
host01    mon,mgr,_admin
host02    mon
host03    mon,mgr
host04
host05
host06
```

2. 使用 **_admin** 标签指定存储集群中的 admin 主机。为获得最佳结果，此主机应同时运行 monitor 和 Manager 守护进程。

语法

```
ceph orch host label add HOSTNAME _admin
```

示例

```
[root@host01 ~]# ceph orch host label add host03 _admin
```

3. 验证 admin 主机是否具有 **_admin** 标签。

示例

```
[root@host01 ~]# ceph orch host ls
HOST ADDR LABELS STATUS
host01    mon,mgr,_admin
host02    mon
host03    mon,mgr,_admin
host04
host05
host06
```

4. 登录 admin 节点，以管理存储集群。

3.18.1. 使用主机标签部署 Ceph 监控节点

典型的 Red Hat Ceph Storage 集群在不同主机上部署了三个或五个 Ceph monitor 守护进程。如果您的存储集群有五个或更多主机，红帽建议您部署五个 Ceph 监控节点。

如果您的 Ceph 监控节点或整个集群都位于单个子网中，则 **cephadm** 会在向集群添加新节点时自动添加最多五个 Ceph 监控守护进程。**cephadm** 自动配置新节点上的 Ceph monitor 守护进程。新节点与存储集群中的第一个（引导）节点位于同一个子网中。**cephadm** 还可以部署和缩放 monitor，以响应存储集群大小的变化。



注意

使用主机标签标识包含 Ceph 监控节点的主机。

先决条件

- 对存储集群中所有节点的根本级别访问权限。

- 正在运行的存储群集。

流程

1. 为主机分配 mon 标签：

语法

```
ceph orch host label add HOSTNAME mon
```

示例

```
[ceph: root@host01 /]# ceph orch host label add host02 mon
[ceph: root@host01 /]# ceph orch host label add host03 mon
```

2. 查看当前的主机和标签：

语法

```
ceph orch host ls
```

示例

```
[ceph: root@host01 /]# ceph orch host ls
HOST ADDR LABELS STATUS
host01      mon,mgr,_admin
host02      mon
host03      mon
host04
host05
host06
```

- 根据主机标签部署 Ceph monitor 守护进程：

语法

```
ceph orch apply mon label:mon
```

- 在特定的一组主机上部署 Ceph monitor 守护进程：

语法

```
ceph orch apply mon HOSTNAME1,HOSTNAME2,HOSTNAME3
```

示例

```
[ceph: root@host01 /]# ceph orch apply mon host01,host02,host03
```



注意

确保将 bootstrap 节点包含在要部署的主机列表中。

3.18.2. 通过 IP 地址或网络名称添加 Ceph 监控节点

典型的 Red Hat Ceph Storage 集群在不同主机上部署了三个或五个 monitor 守护进程。如果您的存储集群有五个或更多主机，红帽建议您部署五个 monitor 节点。

如果您的 monitor 节点或整个集群都位于单个子网中，则 **cephadm** 会在向集群添加新节点时自动添加最多五个 monitor 守护进程。您不需要在新节点上配置 monitor 守护进程。新节点与存储集群中的第一个节点位于同一个子网中。存储集群中的第一个节点是 bootstrap 节点。**cephadm** 还可以部署和缩放 monitor，以响应存储集群大小的变化。

先决条件

- 对存储集群中所有节点的根级别访问权限。
- 正在运行的存储群集。

流程

1. 部署每个额外的 Ceph 监控节点：

语法

```
ceph orch apply mon NODE:IP_ADDRESS_OR_NETWORK_NAME
[NODE:IP_ADDRESS_OR_NETWORK_NAME...]
```

示例

```
[ceph: root@host01 /]# ceph orch apply mon host02:10.10.128.69 host03:mynetwork
```

3.19. 添加管理器服务

在引导过程中，**cephadm** 会在 bootstrap 节点上自动安装管理器守护进程。使用 Ceph 编配器部署额外的管理器守护进程。

Ceph 编配器默认部署两个管理器守护进程。要部署不同数量的管理器守护进程，请指定不同的数字。如果您不指定应当部署管理器守护进程的主机，Ceph 编配器会随机选择主机，并将管理器守护进程部署到主机上。



注意

如果要部署管理器守护进程应用到多个特定的主机，请务必在同一 **ceph orch apply** 命令中指定所有主机名。如果您指定了 **ceph orch apply mgr --placement host1**，然后指定了 **ceph orch apply mgr --placement host2**，第二个命令将删除 host1 上的 Manager 守护进程，并将管理器守护进程应用到 host2。

红帽建议您使用 **--placement** 选项来部署到特定主机。

先决条件

- 正在运行的存储群集。

流程

- 指定您要将一定数量的 Manager 守护进程应用到随机选择的主机：

语法

```
ceph orch apply mgr NUMBER_OF_DAEMONS
```

示例

```
[ceph: root@host01 /]# ceph orch apply mgr 3
```

- 将 Manager 守护进程添加到存储集群中的特定主机上：

语法

```
ceph orch apply mgr --placement "HOSTNAME1 HOSTNAME2 HOSTNAME3"
```

示例

```
[ceph: root@host01 /]# ceph orch apply mgr --placement "host02 host03 host04"
```

3.20. 添加 OSD

Cephadm 不会在不可用的设备上调配 OSD。如果满足以下条件，则存储设备被视为可用：

- 该设备不能有分区。
- 不得挂载该设备。
- 该设备不得包含文件系统。
- 该设备不得包含 Ceph BlueStore OSD。
- 该设备必须大于 5 GB。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。

流程

1. 列出可用的设备来部署 OSD：

语法

```
ceph orch device ls [--hostname=HOSTNAME1 HOSTNAME2] [--wide] [--refresh]
```

示例

```
[ceph: root@host01 /]# ceph orch device ls --wide --refresh
```

2. 您可以在特定主机上或所有可用设备上部署 OSD：

- 从特定主机上的特定设备创建 OSD :

语法

```
ceph orch daemon add osd HOSTNAME:DEVICE_PATH
```

示例

```
[ceph: root@host01 /]# ceph orch daemon add osd host02:/dev/sdb
```

- 若要在任何可用的和未使用的设备上部署 OSD，可使用 **--all-available-devices** 选项。

示例

```
[ceph: root@host01 /]# ceph orch apply osd --all-available-devices
```



注意

这个命令会创建 colocated WAL 和 DB 守护进程。如果要创建非并置守护进程，请不要使用此命令。

其它资源

- 有关 OSD 驱动器规格的更多信息，请参阅 Red Hat Ceph Storage Operations 指南中的 [部署 OSD 的高级服务规格和过滤器](#)。
- 如需有关 zapping 设备清除设备的更多信息，请参阅 Red Hat Ceph Storage Operations 指南中的 [用于 Ceph OSD 部署的 Zapping 设备](#) 部分。

3.21. 运行 CEPHADM-CLIENTS PLAYBOOK

`cephadm-clients.yml` playbook 处理配置和 admin keyring 文件的分发到一组 Ceph 客户端。



注意

如果在运行 playbook 时未指定配置文件，则 playbook 将生成并分发最小配置文件。默认情况下，生成的文件位于 `/etc/ceph/ceph.conf` 中。



注意

在升级 Ceph 集群后如果不使用 `cephadm-ansible` playbook，则必须升级客户端节点上的 `ceph-common` 软件包和客户端库。有关更多信息，请参阅 [Red Hat Ceph Storage 升级指南](#) 中的升级 Red Hat Ceph Storage 集群部分。

先决条件

- 对 Ansible 管理节点的根级别访问权限。
- 具有 sudo 的 Ansible 用户，对存储集群中所有节点的 ssh 访问和免密码访问。
- 已安装 `cephadm-ansible` 软件包。

- `preflight` playbook 在存储集群中的初始主机上运行。如需更多信息，请参阅 [运行 preflight playbook](#)。
- `client_group` 变量必须在 Ansible 清单文件中指定。
- `[admin]` 组在清单文件中定义，其中有一个节点位于 `/etc/ceph/ceph.client.admin.keyring` 中。

流程

1. 导航到 `/usr/share/cephadm-ansible` 目录。
2. 在客户端组中初始主机上运行 `cephadm-clients.yml` playbook。使用 `PATH_TO_KEYRING` 的 `admin` 主机上的 `admin` 密钥环的完整路径名称。可选：如果要指定要使用的现有配置文件，请为 `CONFIG-FILE` 指定配置文件的完整路径。将 Ansible 组名称用于 `ANSIBLE_GROUP_NAME` 的客户端组。使用集群的 `FSID`，其中存储了 `admin keyring` 和配置文件用于 `FSID`。`FSID` 的默认路径是 `/var/lib/ceph/`。

语法

```
ansible-playbook -i hosts cephadm-clients.yml -extra-vars '{"fsid":"FSID",
"client_group":"ANSIBLE_GROUP_NAME", "keyring":"PATH_TO_KEYRING",
"conf":"CONFIG_FILE"}
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-clients.yml --
extra-vars '{"fsid":"be3ca2b2-27db-11ec-892b-
005056833d58","client_group":"fs_clients","keyring":"/etc/ceph/fs.keyring", "conf":
"/etc/ceph/ceph.conf"}
```

安装完成后，组中的指定客户端具有 `admin` 密钥环。如果您没有指定配置文件，`cephadm-ansible` 会在每个客户端上创建最小的默认配置文件。

其它资源

- 有关 `admin` 密钥的更多信息，请参阅 [Red Hat Ceph Storage Administration Guide](#) 中的 [Ceph User Management](#) 部分。

3.22. 清除 CEPH 存储集群

清除 Ceph 存储集群会清除服务器上以前部署中剩余的任何数据或连接。使用 `cephadm rm-cluster` 命令，因为不支持 Ansible。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。

流程

1. 禁用 `cephadm` 来停止所有编配操作，以避免部署新守护进程：

示例

```
[ceph: root#host01 /]# ceph mgr module disable cephadm
```

2. 获取集群的 FSID :

示例

```
[ceph: root#host01 /]# ceph fsid
```

3. 退出 cephadm shell。

示例

```
[ceph: root@host01 /]# exit
```

4. 从集群中的所有主机清除 Ceph 守护进程 :

语法

```
cephadm rm-cluster --force --zap-osds --fsid FSID
```

示例

```
[root@host01 ~]# cephadm rm-cluster --force --zap-osds --fsid a6ca415a-cde7-11eb-a41a-002590fc2544
```

3.23. 部署客户端节点

作为存储管理员，您可以通过运行 **cephadm-preflight.yml** 和 **cephadm-clients.yml** playbook 来部署客户端节点。**cephadm-preflight.yml** playbook 配置 Ceph 存储库，并为引导准备存储集群。它还会安装一些前提条件，如 **podman**、**lvm2**、**chronyd** 和 **cephadm**。

cephadm-clients.yml playbook 处理配置和密钥环文件分发到一组 Ceph 客户端。



注意

如果没有使用 **cephadm-ansible** playbook，在升级 Ceph 集群后，您必须升级客户端节点上的 **ceph-common** 软件包和客户端库。如需更多信息，请参阅[升级 Red Hat Ceph Storage 集群](#)。

先决条件

- 对 Ansible 管理节点的根级别访问权限。
- 具有 sudo 的 Ansible 用户，对存储集群中所有节点的 ssh 访问和免密码访问。
- 安装 **cephadm-ansible** 软件包。
- **[clients]** 组变量必须在 Ansible 清单文件中指定。
- **[admin]** 组在清单文件中定义，其中有一个节点位于 **/etc/ceph/ceph.client.admin.keyring** 中。

流程

1. 以 Ansible 用户身份，进入到 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

2. 打开并编辑 `hosts` 清单文件，并将 `[clients]` 组和客户端添加到您的清单中：

示例

```
host02
host03
host04

[admin]
host01

[clients]
client01
client02
client03
```

3. 运行 `cephadm-preflight.yml` playbook，来在客户端上安装先决条件：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --limit
CLIENT_GROUP_NAME|CLIENT_NODE_NAME
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --
limit clients
```

4. 运行 `cephadm-clients.yml` playbook，将密钥环和 Ceph 配置文件分发到一组客户端。

- a. 使用自定义目标密钥环名称复制密钥环：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-clients.yml --extra-vars
{'fsid':"FSID","keyring":"KEYRING_PATH","client_group":"CLIENT_GROUP_NAME","co
nf":"CEPH_CONFIGURATION_PATH","keyring_dest":"KEYRING_DESTINATION_PATH
"}
```

- 将 `INVENTORY_FILE` 替换为 Ansible 清单文件名称。
- 将 `FSID` 替换为集群的 `FSID`。
- 将 `KEYRING_PATH` 替换为您要复制到客户端的 admin 主机上的密钥环的完整路径名称。
- 可选：将 `CLIENT_GROUP_NAME` 替换为要设置的客户端的 Ansible 组名称。

可选：将 `CEPH_CONFIGURATION_PATH` 替换为管理节点上的 Ceph 配置文件的路径。

- 可选：将 `CEPH_CONFIGURATION_PATH` 替换为管理节点上 Ceph 配置文件的完整路径。
- 可选：将 `KEYRING_DESTINATION_PATH` 替换为要复制密钥环的目标的完整路径名称。



注意

如果在运行 `playbook` 时不使用 `conf` 选项指定配置文件，则 `playbook` 将生成并分发最小配置文件。默认情况下，生成的文件位于 `/etc/ceph/ceph.conf` 中。

示例

```
[ceph-admin@host01 cephadm-ansible]$ ansible-playbook -i hosts
cephadm-clients.yml --extra-vars '{"fsid":"266ee7a8-2a05-11eb-b846-
5254002d4916","keyring":"/etc/ceph/ceph.client.admin.keyring","client_g
roup":"clients","conf":"/etc/ceph/ceph.conf","keyring_dest":"/etc/ceph/cus
tom.name.ceph.keyring"}
```

- b. 使用默认目标密钥环名称 `ceph.keyring` 复制密钥环，并使用默认客户端组：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-clients.yml --extra-vars
{'fsid':'FSID','keyring':'KEYRING_PATH','conf':'CONF_PATH'}
```

验证

登录客户端节点，并验证是否存在密钥环和配置文件。

示例

```
[user@client01 ~]# ls -l /etc/ceph/
-rw-----. 1 ceph ceph 151 Jul 11 12:23 custom.name.ceph.keyring
-rw-----. 1 ceph ceph 151 Jul 11 12:23 ceph.keyring
-rw-----. 1 ceph ceph 269 Jul 11 12:23 ceph.conf
```

第 4 章 使用 CEPHADM-ANSIBLE 模块管理红帽 CEPH 存储集群

作为存储管理员，您可以在 Ansible playbook 中使用 **cephadm-ansible** 模块来管理 Red Hat Ceph Storage 集群。**cephadm-ansible** 软件包提供了多个模块，可以嵌套 **cephadm** 调用，以让您编写自己的唯一 Ansible playbook 来管理集群。



注意

目前，**cephadm-ansible** 模块仅支持最重要的任务。并非 **cephadm-ansible** 模块涵盖的任何操作都必须在 playbook 中使用 **command** 或 **shell** Ansible 模块来完成。

4.1. CEPHADM-ANSIBLE 模块

cephadm-ansible 模块是一组模块，通过打包 **cephadm** 和 **ceph orch** 命令提供一个打包程序来简化 Ansible playbook 的编写过程。您可以使用模块自行编写 Ansible playbook，以通过一个或多个模块来管理集群。

cephadm-ansible 软件包包含以下模块：

- **cephadm_bootstrap**
- **ceph_orch_host**
- **ceph_config**
- **ceph_orch_apply**
- **ceph_orch_daemon**
- **cephadm_registry_login**

4.2. CEPHADM-ANSIBLE 模块选项

下表列出了 **cephadm-ansible** 模块的可用选项。使用 Ansible playbook 中的模块时，需要设置列为必需选项。以默认值 **true** 列出的选项表示在使用模块时会自动设置该选项，且不需要在 playbook 中指定它。例如，对于 **cephadm_bootstrap** 模块，将安装 Ceph 仪表盘，除非设置了 **dashboard: false**。

表 4.1. **cephadm_bootstrap** 模块的可用选项。

cephadm_bootstrap	描述	必填	默认
mon_ip	Ceph 监控 IP 地址。	true	
image	Ceph 容器镜像。	false	
docker	使用 docker 而不是 podman 。	false	
fsid	定义 Ceph FSID。	false	
pull	拉取 Ceph 容器镜像。	false	true

cephadm_bootstrap	描述	必填	默认
dashboard	部署 Ceph 仪表板。	false	true
dashboard_user	指定特定的 Ceph Dashboard 用户。	false	
dashboard_password	Ceph 仪表板密码。	false	
monitoring	部署监控堆栈。	false	true
firewalld	使用 firewalld 管理防火墙规则。	false	true
allow_overwrite	允许覆盖现有 --output-config、--output-keyring 或 --output-pub-ssh-key 文件。	false	false
registry_url	自定义 registry 的 URL。	false	
registry_username	自定义 registry 的用户名。	false	
registry_password	自定义 registry 密码。	false	
registry_json	带有自定义 registry 登录信息的 JSON 文件。	false	
ssh_user	用于 cephadm ssh 到主机的 SSH 用户。	false	
ssh_config	用于 cephadm SSH 客户端的 SSH 配置文件路径。	false	
allow_fqdn_hostname	允许主机名，即完全限定域名(FQDN)。	false	false
cluster_network	用于集群复制、恢复和心跳的子网。	false	

表 4.2. *ceph_orch_host* 模块的可用选项。

ceph_orch_host	描述	必填	默认
fsid	要与之交互的 Ceph 集群的 FSID。	false	
image	要使用的 Ceph 容器镜像。	false	
name	要添加的、删除或更新的主机的名称。	true	
address	主机的 IP 地址。	当 state 为 present 时为 true。	
set_admin_label	在指定主机上设置 _admin 标签。	false	false
labels	应用到主机的标签列表。	false	[]
state	如果设置为 present ，它将确保名称中指定的名称存在。如果设置为 absent ，它将删除名称中指定的主机。如果设置为 drain ，它将调度从名称中指定的主机中删除所有守护进程。	false	存在

表 4.3. ceph_config 模块的可用选项

ceph_config	描述	必填	默认
fsid	要与之交互的 Ceph 集群的 FSID。	false	
image	要使用的 Ceph 容器镜像。	false	
action	在 option 中指定的参数为 set 或 get 。	false	set
who	哪个守护进程将配置设置为。	true	
选项	要进行 set 或 get 的参数名称。	true	
value	要设置的参数值。	如果操作是 set 为 true	

表 4.4. *ceph_orch_apply* 模块的可用选项。

ceph_orch_apply	描述	必填
fsid	要与之交互的 Ceph 集群的 FSID。	false
image	要使用的 Ceph 容器镜像。	false
spec	要应用的服务规格。	true

表 4.5. *ceph_orch_daemon* 模块的可用选项。

ceph_orch_daemon	描述	必填
fsid	要与之交互的 Ceph 集群的 FSID。	false
image	要使用的 Ceph 容器镜像。	false
state	在 name 中指定的理想的服务状态。	true 如果为 started ，它将确保服务已启动。 如果为 stopped ，它将确保该服务已经停止。 如果为 restarted ，它将重启该服务。
daemon_id	服务的 ID。	true
daemon_type	服务的类型。	true

表 4.6. *cephadm_registry_login* 模块的可用选项

cephadm_registry_login	描述	必填	默认
state	登录或注销 registry。	false	login
docker	使用 docker 而不是 podman 。	false	
registry_url	自定义 registry 的 URL。	false	
registry_username	自定义 registry 的用户名。	当 state 为 login 时为 true	

cephadm_registry_login	描述	必填	默认
registry_password	自定义 registry 密码。	当 state 为 login 时为 true	
registry_json	到一个 JSON 文件的路径。在运行此任务前，该文件必须存在于远程主机上。目前不支持这个选项。		

4.3. 使用 CEPHADM_BOOTSTRAP 和 CEPHADM_REGISTRY_LOGIN 模块引导存储集群

作为存储管理员，您可以使用 Ansible 中的 `cephadm_bootstrap` 和 `cephadm_registry_login` 模块来引导存储集群。

先决条件

- 第一个 Ceph 监控容器的 IP 地址，也是存储集群中第一个节点的 IP 地址。
- 登录到 registry.redhat.io。
- 至少 10 GB 的可用空间用于 `/var/lib/containers/`。
- Red Hat Enterprise Linux 8.10 或 9.4，将 `ansible-core` 捆绑到 AppStream 中。
- 在 Ansible 管理节点上安装 `cephadm-ansible` 软件包。
- 在存储集群中的所有主机上设置免密码 SSH。
- 主机通过 CDN 注册。

流程

1. 登录 Ansible 管理节点。
2. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

3. 创建 `hosts` 文件并添加主机、标签和监控存储集群中第一个主机的 IP 地址：

语法

```
sudo vi INVENTORY_FILE

HOST1 labels=["LABEL1", 'LABEL2']
HOST2 labels=["LABEL1", 'LABEL2']
```

```
HOST3 labels=["LABEL1"]
```

```
[admin]
```

```
ADMIN_HOST monitor_address=MONITOR_IP_ADDRESS labels=["ADMIN_LABEL',  
'LABEL1', 'LABEL2']"
```

示例

```
[ceph-admin@admin cephadm-ansible]$ sudo vi hosts
```

```
host02 labels=["mon', 'mgr']"
```

```
host03 labels=["mon', 'mgr']"
```

```
host04 labels=["osd"]"
```

```
host05 labels=["osd"]"
```

```
host06 labels=["osd"]"
```

```
[admin]
```

```
host01 monitor_address=10.10.128.68 labels=["_admin', 'mon', 'mgr']"
```

4. 运行 preflight playbook :

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars "ceph_origin=rhcs"
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml --  
extra-vars "ceph_origin=rhcs"
```

5. 创建 playbook 以启动集群 :

语法

```
sudo vi PLAYBOOK_FILENAME.yml
```

```
---
```

```
- name: NAME_OF_PLAY
```

```
  hosts: BOOTSTRAP_HOST
```

```
  become: USE_ELEVATED_PRIVILEGES
```

```
  gather_facts: GATHER_FACTS_ABOUT_REMOTE_HOSTS
```

```
  tasks:
```

```
    -name: NAME_OF_TASK
```

```
      cephadm_registry_login:
```

```
        state: STATE
```

```
        registry_url: REGISTRY_URL
```

```
        registry_username: REGISTRY_USER_NAME
```

```
        registry_password: REGISTRY_PASSWORD
```

```
    - name: NAME_OF_TASK
```

```
      cephadm_bootstrap:
```

```
        mon_ip: "{{ monitor_address }}"
```

```
        dashboard_user: DASHBOARD_USER
```



```

dashboard_password: DASHBOARD_PASSWORD
allow_fqdn_hostname: ALLOW_FQDN_HOSTNAME
cluster_network: NETWORK_CIDR

```

示例

```
[ceph-admin@admin cephadm-ansible]$ sudo vi bootstrap.yml
```

```

---
- name: bootstrap the cluster
  hosts: host01
  become: true
  gather_facts: false
  tasks:
    - name: login to registry
      cephadm_registry_login:
        state: login
        registry_url: registry.redhat.io
        registry_username: user1
        registry_password: mypassword1

    - name: bootstrap initial cluster
      cephadm_bootstrap:
        mon_ip: "{{ monitor_address }}"
        dashboard_user: mydashboarduser
        dashboard_password: mydashboardpassword
        allow_fqdn_hostname: true
        cluster_network: 10.10.128.0/28

```

6. 运行playbook :

语法

```
ansible-playbook -i INVENTORY_FILE PLAYBOOK_FILENAME.yml -vvv
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts bootstrap.yml -vvv
```

验证

- 在运行playbook 后检查Ansible 输出。

4.4. 使用 CEPH_ORCH_HOST 模块添加或删除主机

作为存储管理员，您可以使用 Ansible playbook 中的 `ceph_orch_host` 模块添加和删除存储集群中的主机。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- 将节点注册到 CDN 并附加订阅。

- 具有 `sudo` 的 Ansible 用户，对存储集群中的所有节点进行免密码 SSH 访问。
- 在 Ansible 管理节点上安装 **cephadm-ansible** 软件包。
- 新主机具有存储集群的公共 SSH 密钥。有关将存储集群的公共 SSH 密钥复制到新主机的更多信息，请参阅 [添加主机](#)。

流程

1. 使用以下步骤在集群中添加新主机：
 - a. 登录 Ansible 管理节点。
 - b. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

- c. 将新主机和标签添加到 Ansible 清单文件。

语法

```
sudo vi INVENTORY_FILE
```

```
NEW_HOST1 labels=["LABEL1', 'LABEL2']"
NEW_HOST2 labels=["LABEL1', 'LABEL2']"
NEW_HOST3 labels=["LABEL1']"
```

```
[admin]
```

```
ADMIN_HOST monitor_address=MONITOR_IP_ADDRESS labels=["ADMIN_LABEL',
'LABEL1', 'LABEL2']"
```

示例

```
[ceph-admin@admin cephadm-ansible]$ sudo vi hosts
```

```
host02 labels=["mon', 'mgr']"
host03 labels=["mon', 'mgr']"
host04 labels=["osd']"
host05 labels=["osd']"
host06 labels=["osd']"
```

```
[admin]
```

```
host01 monitor_address= 10.10.128.68 labels=["_admin', 'mon', 'mgr']"
```

- d. 使用 `--limit` 选项运行 `preflight` playbook：

语法

```
ansible-playbook -i INVENTORY_FILE cephadm-preflight.yml --extra-vars
"ceph_origin=rhcs" --limit NEWHOST
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts cephadm-preflight.yml
--extra-vars "ceph_origin=rhcs" --limit host02
```

preflight playbook 在新主机上安装 **podman**、**lvm2**、**chronyd** 和 **cephadm**。安装完成后，**cephadm** 驻留在 `/usr/sbin/` 目录中。

- e. 创建 playbook 以将新主机添加到集群中：

语法

```
sudo vi PLAYBOOK_FILENAME.yml

---
- name: PLAY_NAME
  hosts: HOSTS_OR_HOST_GROUPS
  become: USE_ELEVATED_PRIVILEGES
  gather_facts: GATHER_FACTS_ABOUT_REMOTE_HOSTS
  tasks:
    - name: NAME_OF_TASK
      ceph_orch_host:
        name: "{{ ansible_facts['hostname'] }}"
        address: "{{ ansible_facts['default_ipv4']['address'] }}"
        labels: "{{ labels }}"
      delegate_to: HOST_TO_DELEGATE_TASK_TO

    - name: NAME_OF_TASK
      when: inventory_hostname in groups['admin']
      ansible.builtin.shell:
        cmd: CEPH_COMMAND_TO_RUN
      register: REGISTER_NAME

    - name: NAME_OF_TASK
      when: inventory_hostname in groups['admin']
      debug:
        msg: "{{ REGISTER_NAME.stdout }}"
```



注意

默认情况下，Ansible 在与 playbook 的 **hosts** 行匹配的主机上执行所有任务。**ceph orch** 命令必须在包含管理员密钥环和 Ceph 配置文件的主机上运行。使用 **delegate_to** 关键字指定集群中的 admin 主机。

示例

```
[ceph-admin@admin cephadm-ansible]$ sudo vi add-hosts.yml

---
- name: add additional hosts to the cluster
  hosts: all
  become: true
  gather_facts: true
  tasks:
    - name: add hosts to the cluster
      ceph_orch_host:
```

```

    name: "{{ ansible_facts['hostname'] }}"
    address: "{{ ansible_facts['default_ipv4']['address'] }}"
    labels: "{{ labels }}"
    delegate_to: host01

- name: list hosts in the cluster
  when: inventory_hostname in groups['admin']
  ansible.builtin.shell:
    cmd: ceph orch host ls
  register: host_list

- name: print current list of hosts
  when: inventory_hostname in groups['admin']
  debug:
    msg: "{{ host_list.stdout }}"

```

在本例中，playbook 将新主机添加到集群中，并显示当前的主机列表。

- f. 运行playbook 以将其他主机添加到集群中：

语法

```
ansible-playbook -i INVENTORY_FILE PLAYBOOK_FILENAME.yml
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts add-hosts.yml
```

2. 使用以下步骤从集群中删除主机：

- a. 登录 Ansible 管理节点。
- b. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

- c. 创建playbook 以从集群中删除主机或主机：

语法

```

sudo vi PLAYBOOK_FILENAME.yml

---
- name: NAME_OF_PLAY
  hosts: ADMIN_HOST
  become: USE_ELEVATED_PRIVILEGES
  gather_facts: GATHER_FACTS_ABOUT_REMOTE_HOSTS
  tasks:
    - name: NAME_OF_TASK
      ceph_orch_host:
        name: HOST_TO_REMOVE
        state: STATE

```

```

- name: NAME_OF_TASK
  ceph_orch_host:
    name: HOST_TO_REMOVE
    state: STATE
  retries: NUMBER_OF_RETRIES
  delay: DELAY
  until: CONTINUE_UNTIL
  register: REGISTER_NAME

- name: NAME_OF_TASK
  ansible.builtin.shell:
    cmd: ceph orch host ls
  register: REGISTER_NAME

- name: NAME_OF_TASK
  debug:
    msg: "{{ REGISTER_NAME.stdout }}"

```

示例

```

[ceph-admin@admin cephadm-ansible]$ sudo vi remove-hosts.yml

---
- name: remove host
  hosts: host01
  become: true
  gather_facts: true
  tasks:
    - name: drain host07
      ceph_orch_host:
        name: host07
        state: drain

    - name: remove host from the cluster
      ceph_orch_host:
        name: host07
        state: absent
      retries: 20
      delay: 1
      until: result is succeeded
      register: result

    - name: list hosts in the cluster
      ansible.builtin.shell:
        cmd: ceph orch host ls
      register: host_list

    - name: print current list of hosts
      debug:
        msg: "{{ host_list.stdout }}"

```

在本例中，playbook 任务排空 **host07** 上的所有守护进程，从集群中删除主机，并显示当前主机列表。

- d. 运行playbook 以从集群中删除主机：

语法

```
ansible-playbook -i INVENTORY_FILE PLAYBOOK_FILENAME.yml
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts remove-hosts.yml
```

验证

- 查看 Ansible 任务输出显示集群中主机的当前列表：

示例

```
TASK [print current hosts]
*****
Friday 24 June 2022 14:52:40 -0400 (0:00:03.365)    0:02:31.702 *****
ok: [host01] =>
msg: |-
  HOST  ADDR      LABELS  STATUS
  host01 10.10.128.68 _admin mon mgr
  host02 10.10.128.69 mon mgr
  host03 10.10.128.70 mon mgr
  host04 10.10.128.71 osd
  host05 10.10.128.72 osd
  host06 10.10.128.73 osd
```

4.5. 使用 CEPH_CONFIG 模块设置配置选项

作为存储管理员，您可以使用 `ceph_config` 模块设置或获取 Red Hat Ceph Storage 配置选项。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- 具有 `sudo` 的 Ansible 用户，对存储集群中的所有节点进行免密码 SSH 访问。
- 在 Ansible 管理节点上安装 `cephadm-ansible` 软件包。
- Ansible 清单文件包含集群和 admin 主机。有关将主机添加到存储集群中的更多信息，请参阅使用 `ceph_orch_host` 模块添加或删除主机。

流程

1. 登录 Ansible 管理节点。
2. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

3. 使用配置更改创建 playbook :

语法

```

sudo vi PLAYBOOK_FILENAME.yml

---
- name: PLAY_NAME
  hosts: ADMIN_HOST
  become: USE_ELEVATED_PRIVILEGES
  gather_facts: GATHER_FACTS_ABOUT_REMOTE_HOSTS
  tasks:
    - name: NAME_OF_TASK
      ceph_config:
        action: GET_OR_SET
        who: DAEMON_TO_SET_CONFIGURATION_TO
        option: CEPH_CONFIGURATION_OPTION
        value: VALUE_OF_PARAMETER_TO_SET

    - name: NAME_OF_TASK
      ceph_config:
        action: GET_OR_SET
        who: DAEMON_TO_SET_CONFIGURATION_TO
        option: CEPH_CONFIGURATION_OPTION
        register: REGISTER_NAME

    - name: NAME_OF_TASK
      debug:
        msg: "MESSAGE_TO_DISPLAY {{ REGISTER_NAME.stdout }}"

```

示例

```

[ceph-admin@admin cephadm-ansible]$ sudo vi change_configuration.yml

---
- name: set pool delete
  hosts: host01
  become: true
  gather_facts: false
  tasks:
    - name: set the allow pool delete option
      ceph_config:
        action: set
        who: mon
        option: mon_allow_pool_delete
        value: true

    - name: get the allow pool delete setting
      ceph_config:
        action: get
        who: mon
        option: mon_allow_pool_delete
        register: verify_mon_allow_pool_delete

```

```
- name: print current mon_allow_pool_delete setting
  debug:
    msg: "the value of 'mon_allow_pool_delete' is {{ verify_mon_allow_pool_delete.stdout }}"
```

在本例中，playbook 首先将 **mon_allow_pool_delete** 选项设置为 **false**。然后，playbook 获取当前的 **mon_allow_pool_delete** 设置，并在 Ansible 输出中显示值。

4. 运行 playbook :

语法

```
ansible-playbook -i INVENTORY_FILE _PLAYBOOK_FILENAME.yml
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts change_configuration.yml
```

验证

- 检查 playbook 任务的输出。

示例

```
TASK [print current mon_allow_pool_delete setting]
*****
Wednesday 29 June 2022 13:51:41 -0400 (0:00:05.523)    0:00:17.953 *****
ok: [host01] =>
  msg: the value of 'mon_allow_pool_delete' is true
```

其它资源

- 有关配置选项的更多详细信息，请参阅 [Red Hat Ceph Storage 配置指南](#)。

4.6. 使用 CEPH_ORCH_APPLY 模块应用服务规格

作为存储管理员，您可以使用 Ansible playbook 中的 **ceph_orch_apply** 模块将服务规格应用到存储集群。服务规格是一个数据结构，它指定用于部署 Ceph 服务的属性和服务配置。您可以使用服务规格来部署 Ceph 服务类型，如 **mon**、**crash**、**mds**、**mgr**、**osd**、**rdb** 或 **rbd-mirror**。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- 具有 sudo 的 Ansible 用户，对存储集群中的所有节点进行免密码 SSH 访问。
- 在 Ansible 管理节点上安装 **cephadm-ansible** 软件包。
- Ansible 清单文件包含集群和 admin 主机。有关将主机添加到存储集群中的更多信息，请参阅使用 [ceph_orch_host](#) 模块添加或删除主机。

流程

1. 登录 Ansible 管理节点。

2. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录 :

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

3. 使用服务规格创建 playbook :

语法

```
sudo vi PLAYBOOK_FILENAME.yml

---
- name: PLAY_NAME
  hosts: HOSTS_OR_HOST_GROUPS
  become: USE_ELEVATED_PRIVILEGES
  gather_facts: GATHER_FACTS_ABOUT_REMOTE_HOSTS
  tasks:
    - name: NAME_OF_TASK
      ceph_orch_apply:
        spec: |
          service_type: SERVICE_TYPE
          service_id: UNIQUE_NAME_OF_SERVICE
          placement:
            host_pattern: 'HOST_PATTERN_TO_SELECT_HOSTS'
            label: LABEL
          spec:
            SPECIFICATION_OPTIONS:
```

示例

```
[ceph-admin@admin cephadm-ansible]$ sudo vi deploy_osd_service.yml

---
- name: deploy osd service
  hosts: host01
  become: true
  gather_facts: true
  tasks:
    - name: apply osd spec
      ceph_orch_apply:
        spec: |
          service_type: osd
          service_id: osd
          placement:
            host_pattern: '*'
            label: osd
          spec:
            data_devices:
              all: true
```

在本例中, `playbook` 在所有主机上部署 Ceph OSD 服务, 其标签为 **osd**。

4. 运行 `playbook` :

语法

```
ansible-playbook -i INVENTORY_FILE _PLAYBOOK_FILENAME.yml
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts deploy_osd_service.yml
```

验证

- 检查 playbook 任务的输出。

其它资源

- 如需了解有关服务规格选项的更多详细信息，请参阅 [Red Hat Ceph Storage Operations Guide](#)。

4.7. 使用 CEPH_ORCH_DAEMON 模块管理 CEPH 守护进程状态

作为存储管理员，您可以使用 Ansible playbook 中的 `ceph_orch_daemon` 模块在主机上启动、停止和重启 Ceph 守护进程。

先决条件

- 一个正在运行的 Red Hat Ceph Storage 集群。
- 具有 `sudo` 的 Ansible 用户，对存储集群中的所有节点进行免密码 SSH 访问。
- 在 Ansible 管理节点上安装 `cephadm-ansible` 软件包。
- Ansible 清单文件包含集群和 admin 主机。有关将主机添加到存储集群中的更多信息，请参阅使用 [ceph_orch_host](#) 模块添加或删除主机。

流程

1. 登录 Ansible 管理节点。
2. 进入 Ansible 管理节点上的 `/usr/share/cephadm-ansible` 目录：

示例

```
[ceph-admin@admin ~]$ cd /usr/share/cephadm-ansible
```

3. 创建带有守护进程状态更改的 playbook：

语法

```
sudo vi PLAYBOOK_FILENAME.yml

---
- name: PLAY_NAME
  hosts: ADMIN_HOST
  become: USE_ELEVATED_PRIVILEGES
  gather_facts: GATHER_FACTS_ABOUT_REMOTE_HOSTS
```

```
tasks:
  - name: NAME_OF_TASK
    ceph_orch_daemon:
      state: STATE_OF_SERVICE
      daemon_id: DAEMON_ID
      daemon_type: TYPE_OF_SERVICE
```

示例

```
[ceph-admin@admin cephadm-ansible]$ sudo vi restart_services.yml
```

```
---
- name: start and stop services
  hosts: host01
  become: true
  gather_facts: false
  tasks:
    - name: start osd.0
      ceph_orch_daemon:
        state: started
        daemon_id: 0
        daemon_type: osd

    - name: stop mon.host02
      ceph_orch_daemon:
        state: stopped
        daemon_id: host02
        daemon_type: mon
```

在本例中，playbook 启动 ID 为 **0** 的 OSD，并停止 ID 为 **host02** 的 Ceph Monitor。

4. 运行 playbook :

语法

```
ansible-playbook -i INVENTORY_FILE _PLAYBOOK_FILENAME.yml
```

示例

```
[ceph-admin@admin cephadm-ansible]$ ansible-playbook -i hosts restart_services.yml
```

验证

- 检查 playbook 任务的输出。

第5章 接下来该怎么办？第2天

作为存储管理员，一旦安装和配置了 Red Hat Ceph Storage 7，您便准备好为存储集群执行"第2天"操作。这些操作包括添加元数据服务器(MDS)和对象网关(RGW)，以及配置 NFS 等服务。

有关如何使用 **cephadm** 编配器执行"第2天"操作的更多信息，请参阅[Red Hat Ceph Storage 7 操作指南](#)。

要在"第2天"操作中部署、配置和管理 Ceph 对象网关，请参阅[Red Hat Ceph Storage 7 对象网关指南](#)。

附录 A. CEPH ANSIBLE 和 CEPHADM 之间的比较

`cephadm` 用于容器化存储集群部署。

下表将 `Cephadm` 与 `Ceph-Ansible` `playbook` 进行比较，以管理 `Ceph` 集群的容器化部署，以进行第一天和第二天操作。

表 A.1. 第一天操作

描述	Ceph-Ansible	Cephadm
安装 Red Hat Ceph Storage 集群	运行 <code>site-container.yml</code> <code>playbook</code> 。	运行 <code>cephadm bootstrap</code> 命令，在管理节点上引导集群。
添加主机	使用 <code>Ceph Ansible</code> 清单。	运行 <code>ceph orch add host HOST_NAME</code> 以将主机添加到集群中。
添加 monitor	运行 <code>add-mon.yml</code> <code>playbook</code> 。	运行 <code>ceph orch apply mon</code> 命令。
增加经理	运行 <code>site-container.yml</code> <code>playbook</code> 。	运行 <code>ceph orch apply mgr</code> 命令。
添加 OSD	运行 <code>add-osd.yml</code> <code>playbook</code> 。	运行 <code>ceph orch apply osd</code> 命令，以在所有可用设备或特定主机上添加 OSD。
在特定设备上添加 OSD	选择 <code>osd.yml</code> 文件中的 <code>devices</code> ，然后运行 <code>add-osd.yml</code> <code>playbook</code> 。	选择 <code>osd.yml</code> 文件中的 <code>data_devices</code> 下的 <code>paths</code> 过滤器，然后运行 <code>ceph orch apply -i FILE_NAME.yml</code> 命令。
添加 MDS	运行 <code>site-container.yml</code> <code>playbook</code> 。	运行 <code>ceph orch apply FILESYSTEM_NAME</code> 命令以添加 MDS。
添加 Ceph 对象网关	运行 <code>site-container.yml</code> <code>playbook</code> 。	运行 <code>ceph orch apply rgw</code> 命令，以添加 <code>Ceph</code> 对象网关。

表 A.2. 第二天操作

描述	Ceph-Ansible	Cephadm
删除主机	使用 <code>Ansible</code> 清单。	运行 <code>ceph orch host rm HOST_NAME</code> 以删除主机。
删除 monitor	运行 <code>shrink-mon.yml</code> <code>playbook</code> 。	运行 <code>ceph orch apply mon</code> 以重新部署其他 <code>monitor</code> 。

描述	Ceph-Ansible	Cephadm
删除管理器	运行 shrink-mon.yml playbook。	运行 ceph orch apply mgr 以重新部署其他管理器。
删除 OSD	运行 shrink-osd.yml playbook。	运行 ceph orch osd rm <i>OSD_ID</i> 以移除 OSD。
删除 MDS	运行 shrink-mds.yml playbook。	运行 ceph orch rm <i>SERVICE_NAME</i> 以删除特定的服务。
通过 NFS 协议导出 Ceph 文件系统。	在 Red Hat Ceph Storage 4 中不支持。	运行 ceph nfs export create 命令。
部署 Ceph 对象网关	运行 site-container.yml playbook。	运行 ceph orch apply rgw <i>SERVICE_NAME</i> 来部署 Ceph 对象网关服务。
删除 Ceph 对象网关	运行 shrink-rgw.yml playbook。	运行 ceph orch rm <i>SERVICE_NAME</i> 以删除特定的服务。
块设备镜像	运行 site-container.yml playbook。	运行 ceph orch apply rbd-mirror 命令。
Red Hat Ceph Storage 的次版本升级	运行 infrastructure-playbooks/rolling_update.yml playbook。	运行 ceph orch upgrade start 命令。
部署监控堆栈	在安装过程中编辑 all.yml 文件。	指定服务后，运行 ceph orch apply -i <i>FILE.yml</i> 。

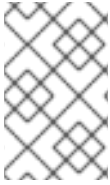
其它资源

- 有关使用 Ceph 编排器的更多详细信息，请参见 [Red Hat Ceph Storage 操作指南](#)。

附录 B. CEPHADM 命令

cephadm 是一种命令行工具，可用于管理 Cephadm 编排器的本地主机。它提供用于调查和修改当前主机状态的命令。

一些命令通常用于调试。



注意

并非所有主机上都不需要 **cephadm**，但在调查特定守护进程时非常有用。**cephadm-ansible-preflight** playbook 在所有主机上安装 **cephadm** -preflight playbook，并且 **cephadm-ansible-purge** playbook 需要在所有主机上安装 **cephadm** 才能正常工作。

adopt

描述

转换升级的存储集群守护进程来运行 **cephadm**。

语法

```
cephadm adopt [-h] --name DAEMON_NAME --style STYLE [--cluster CLUSTER] --legacy-dir [LEGACY_DIR] --config-json CONFIG_JSON [--skip-firewalld] [--skip-pull]
```

示例

```
[root@host01 ~]# cephadm adopt --style=legacy --name prometheus.host02
```

ceph-volume

描述

此命令用于列出特定主机上的所有设备。在容器内运行 **ceph-volume** 命令，利用可插拔工具（如 **lvm** 或物理磁盘）部署 OSD，并遵循准备、激活和启动 OSD 的可靠方法。

语法

```
cephadm ceph-volume inventory/simple/raw/lvm [-h] [--fsid FSID] [--config-json CONFIG_JSON] [--config CONFIG, -c CONFIG] [--keyring KEYRING, -k KEYRING]
```

示例

```
[root@nhost01 ~]# cephadm ceph-volume inventory --fsid f64f341c-655d-11eb-8778-fa163e914bcc
```

check-host

描述

检查适合 Ceph 集群的主机配置。

语法

```
cephadm check-host [--expect-hostname HOSTNAME]
```

示例

```
[root@host01 ~]# cephadm check-host --expect-hostname host02
```

deploy

描述

在本地主机上部署守护进程。

语法

```
cephadm shell deploy DAEMON_TYPE [-h] [--name DAEMON_NAME] [--fsid FSID] [--config CONFIG, -c CONFIG] [--config-json CONFIG_JSON] [--keyring KEYRING] [--key KEY] [--osd-fsid OSD_FSID] [--skip-firewalld] [--tcp-ports TCP_PORTS] [--reconfig] [--allow-pttrace] [--memory-request MEMORY_REQUEST] [--memory-limit MEMORY_LIMIT] [--meta-json META_JSON]
```

示例

```
[root@host01 ~]# cephadm shell deploy mon --fsid f64f341c-655d-11eb-8778-fa163e914bcc
```

enter

描述

在正在运行的守护进程容器内运行交互式 shell。

语法

```
cephadm enter [-h] [--fsid FSID] --name NAME [command [command ...]]
```

示例

```
[root@host01 ~]# cephadm enter --name 52c611f2b1d9
```

帮助

描述

查看 **cephadm** 支持的所有命令。

语法

```
cephadm help
```

示例

```
[root@host01 ~]# cephadm help
```

install

描述

安装软件包。

语法


```
cephadm install PACKAGES
```

示例

```
[root@host01 ~]# cephadm install ceph-common ceph-osd
```

inspect-image

描述

检查本地 Ceph 容器镜像。

语法

```
cephadm --image IMAGE_ID inspect-image
```

示例

```
[root@host01 ~]# cephadm --image  
13ea90216d0be03003d12d7869f72ad9de5cec9e54a27fd308e01e467c0d4a0a inspect-image
```

list-networks

描述

列出 IP 网络。

语法

```
cephadm list-networks
```

示例

```
[root@host01 ~]# cephadm list-networks
```

ls

描述

列出主机上 **cephadm** 已知的守护进程实例。您可以使用 **--no-detail** 命令来更快地运行，这详细介绍了守护进程名称、fsid、样式和 systemd 单元。您可以使用 **--legacy-dir** 选项指定用于搜索后台程序的旧基础目录。

语法

```
cephadm ls [--no-detail] [--legacy-dir LEGACY_DIR]
```

示例

```
[root@host01 ~]# cephadm ls --no-detail
```

logs

描述

输出守护进程容器的 **journald** 日志。这类似于 **journalctl** 命令。

语法

```
cephadm logs [--fsid FSID] --name DAEMON_NAME
cephadm logs [--fsid FSID] --name DAEMON_NAME -- -n NUMBER # Last N lines
cephadm logs [--fsid FSID] --name DAEMON_NAME -- -f # Follow the logs
```

示例

```
[root@host01 ~]# cephadm logs --fsid 57bddb48-ee04-11eb-9962-001a4a000672 --name
osd.8
[root@host01 ~]# cephadm logs --fsid 57bddb48-ee04-11eb-9962-001a4a000672 --name
osd.8 -- -n 20
[root@host01 ~]# cephadm logs --fsid 57bddb48-ee04-11eb-9962-001a4a000672 --name
osd.8 -- -f
```

prepare-host

描述

为 **cephadm** 准备主机。

语法

```
cephadm prepare-host [--expect-hostname HOSTNAME]
```

示例

```
[root@host01 ~]# cephadm prepare-host
[root@host01 ~]# cephadm prepare-host --expect-hostname host01
```

pull

描述

拉取 Ceph 镜像。

语法

```
cephadm [-h] [--image IMAGE_ID] pull
```

示例

```
[root@host01 ~]# cephadm --image
13ea90216d0be03003d12d7869f72ad9de5cec9e54a27fd308e01e467c0d4a0a pull
```

registry-login

描述

为经过身份验证的 registry 提供 cephadm 登录信息。Cephadm 会尝试将调用主机记录到该 registry 中。

语法

```
cephadm registry-login --registry-url [REGISTRY_URL] --registry-username [USERNAME] --
registry-password [PASSWORD] [--fsid FSID] [--registry-json JSON_FILE]
```

示例

```
[root@host01 ~]# cephadm registry-login --registry-url registry.redhat.io --registry-username
myuser1 --registry-password mypassword1
```

您还可以使用包含日志信息格式的 JSON registry 文件，如下所示：

语法

```
cat REGISTRY_FILE

{
  "url":"REGISTRY_URL",
  "username":"REGISTRY_USERNAME",
  "password":"REGISTRY_PASSWORD"
}
```

示例

```
[root@host01 ~]# cat registry_file

{
  "url":"registry.redhat.io",
  "username":"myuser",
  "password":"mypass"
}

[root@host01 ~]# cephadm registry-login -i registry_file
```

rm-daemon

描述

移除特定的守护进程实例。如果您在主机上直接运行 **cephadm rm-daemon** 命令，虽然命令删除了守护进程，但 **cephadm mgr** 模块会注意到守护进程丢失并重新部署。这个命令存在问题，应该只用于实验目的和调试。

语法

```
cephadm rm-daemon [--fsid FSID] [--name DAEMON_NAME] [--force ] [--force-delete-data]
```

示例

```
[root@host01 ~]# cephadm rm-daemon --fsid f64f341c-655d-11eb-8778-fa163e914bcc --name
osd.8
```

rm-cluster

描述

从该特定主机上运行的存储集群中移除所有守护进程。与 `rm-daemon` 类似，如果您以这种方式删除几个守护进程，Ceph 编排器没有暂停，并且其中一些守护进程属于未管理的服务，`cephadm` 编排器刚刚重新部署它们。

语法

```
cephadm rm-cluster [--fsid FSID] [--force]
```

示例

```
[root@host01 ~]# cephadm rm-cluster --fsid f64f341c-655d-11eb-8778-fa163e914bcc
```



重要

为了更好地清理节点作为执行集群删除的一部分，在运行 `cephadm rm-cluster` 命令时会删除 `/var/log/ceph` 目录下的集群日志。只要 `--keep-logs` 没有传递给 `rm-cluster` 命令，集群日志就会被删除。



注意

如果 `cephadm rm-cluster` 命令在作为由 Cephadm 管理的现有集群一部分的主机上运行，并且 Cephadm Manager 模块仍处于启用和运行状态，则 Cephadm 可能会立即开始部署新守护进程，并且可能会出现更多日志。要避免这种情况，请在清除集群前禁用 `cephadm mgr` 模块。

```
# ceph mgr module disable cephadm
```

rm-repo

描述

删除软件包存储库配置。这主要用于 Red Hat Ceph Storage 的断开连接的安装。

语法

```
cephadm rm-repo [-h]
```

示例

```
[root@host01 ~]# cephadm rm-repo
```

run

描述

在容器中运行 Ceph 守护进程（在前台）。

语法

```
cephadm run [--fsid FSID] --name DAEMON_NAME
```

示例

```
[root@host01 ~]# cephadm run --fsid f64f341c-655d-11eb-8778-fa163e914bcc --name osd.8
```

shell

描述

通过推断或指定的 Ceph 集群，运行可以访问 Ceph 命令的交互式 shell。您可以使用 **cephadm shell** 命令进入 shell，并在 shell 中运行所有编配器命令。

语法

```
cephadm shell [--fsid FSID] [--name DAEMON_NAME, -n DAEMON_NAME] [--config CONFIG, -c CONFIG] [--mount MOUNT, -m MOUNT] [--keyring KEYRING, -k KEYRING] [--env ENV, -e ENV]
```

示例

```
[root@host01 ~]# cephadm shell -- ceph orch ls
[root@host01 ~]# cephadm shell
```

unit

描述

通过此操作启动、停止、重新启动、启用和禁用守护进程。这在守护进程的 **systemd** 单元中运行。

语法

```
cephadm unit [--fsid FSID] --name DAEMON_NAME start/stop/restart/enable/disable
```

示例

```
[root@host01 ~]# cephadm unit --fsid f64f341c-655d-11eb-8778-fa163e914bcc --name osd.8
start
```

version

描述

提供存储集群的版本。

语法

```
cephadm version
```

示例

```
[root@host01 ~]# cephadm version
```