

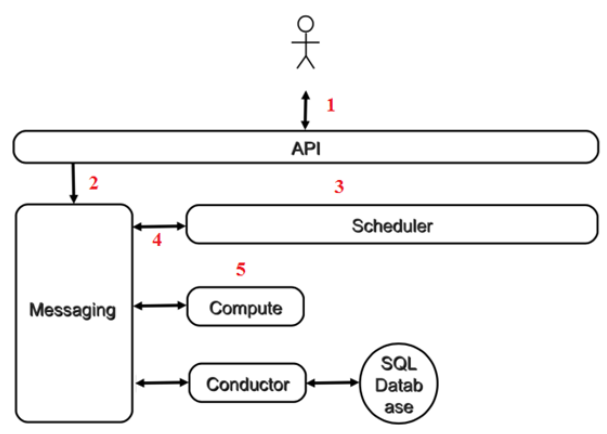
Rebuild 可以恢复损坏的 instance。

那如果是宿主机坏了怎么办呢？ 比如硬件故障或者断电造成整台计算节点无法工作，该节点上运行的 instance 如何恢复呢？

用 Shelve 或者 Migrate 可不可以？ 很不幸，这两个操作都要求 instance 所在计算节点的 nova-compute 服务正常运行。 幸运的是，还有 Evacuate 操作。

Evacuate 可在 nova-compute 无法工作的情况下将节点上的 instance 迁移到其他计算节点上。但有个前提： Instance 的镜像文件必须放在共享存储上。

下面是 Evacuate instance 的流程图

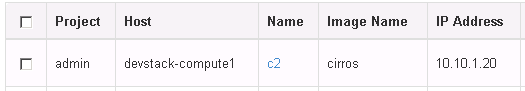


1. 向 nova-api 发送请求
2. nova-api 发送消息
3. nova-scheduler 执行调度
4. nova-scheduler 发送消息
5. nova-compute 执行操作

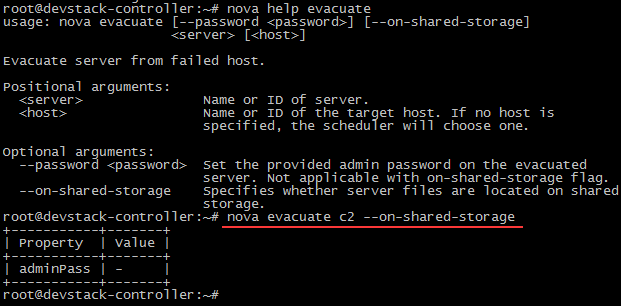
下面我们详细讨论每一个步骤。

**向 nova-api 发送请求**

我们的实验场景如下： Instance c2 运行在 devstack-compute1 上。

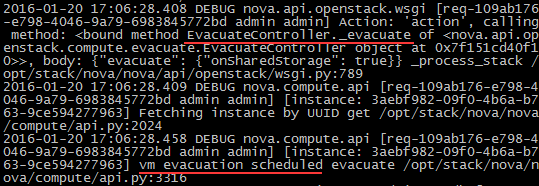


通过断电模拟计算节点故障，然后执行 Evacuate 操作恢复 instance c2。 目前 Evacuate 只能通过 CLI 执行。



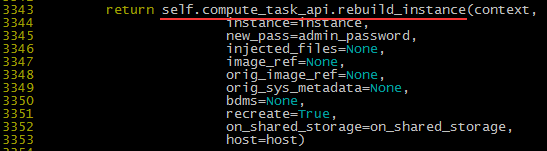
这里需要指定 --on-shared-storage 这个参数

查看日志 /opt/stack/logs/n-api.log



**nova-api 发送消息**

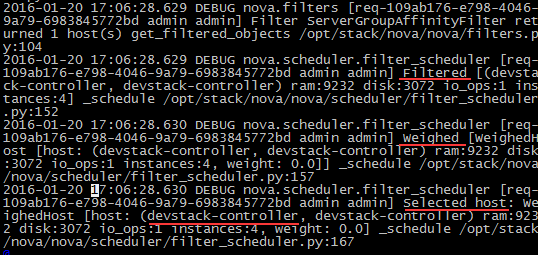
nova-api 向 Messaging（RabbitMQ）发送了一条消息：“Evacuate 这个 Instance” 查看源代码 /opt/stack/nova/nova/compute/api.py，方法是 evacuate。



大家注意到没有，**evacuate 实际上是通过 rebuild 操作实现的**。 这是可以理解的，因为 evacuate 是用共享存储上 instance 的镜像文件重新创建虚机

**nova-scheduler 执行调度**

nova-scheduler 收到消息后，会为 instance 选择合适的计算节点。 查看日志 /opt/stack/logs/n-sch.log。



nova-scheduler 最后选择在 devstack-controller 计算节点上重建 instance。

**nova-scheduler 发送消息**

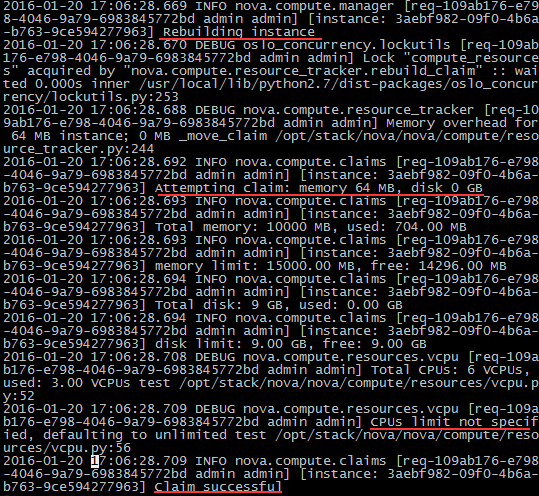
nova-scheduler 发送消息，通知计算节点可以创建 instance 了。 源代码在 /opt/stack/nova/nova/scheduler/filter\_scheduler.py 第 95 行，方法为 select\_destinations。

image254.png

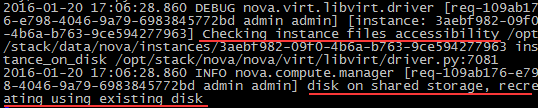
**nova-compute 执行操作**

计算节点上的工作是用共享存储上的镜像文件重建 instance。 日志在 devstack-controller:/opt/stack/logs/n-cpu.log。

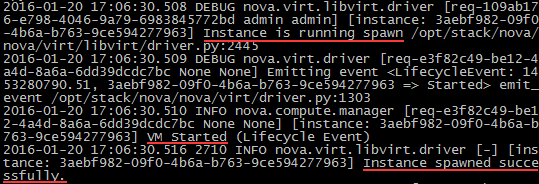
**为instance分配资源**



**使用共享存储上的镜像文件**



**启动 instance**



Evacuate 操作完成后，instance 在 devstack-controller 上运行。

以上是 Evacuate 操作的详细分析。  
至此，我们已经学习完 Nova 所有的操作，下一节将用一张图总结这些操作的用途和使用场景。