Cinder: configurazioni avanzate

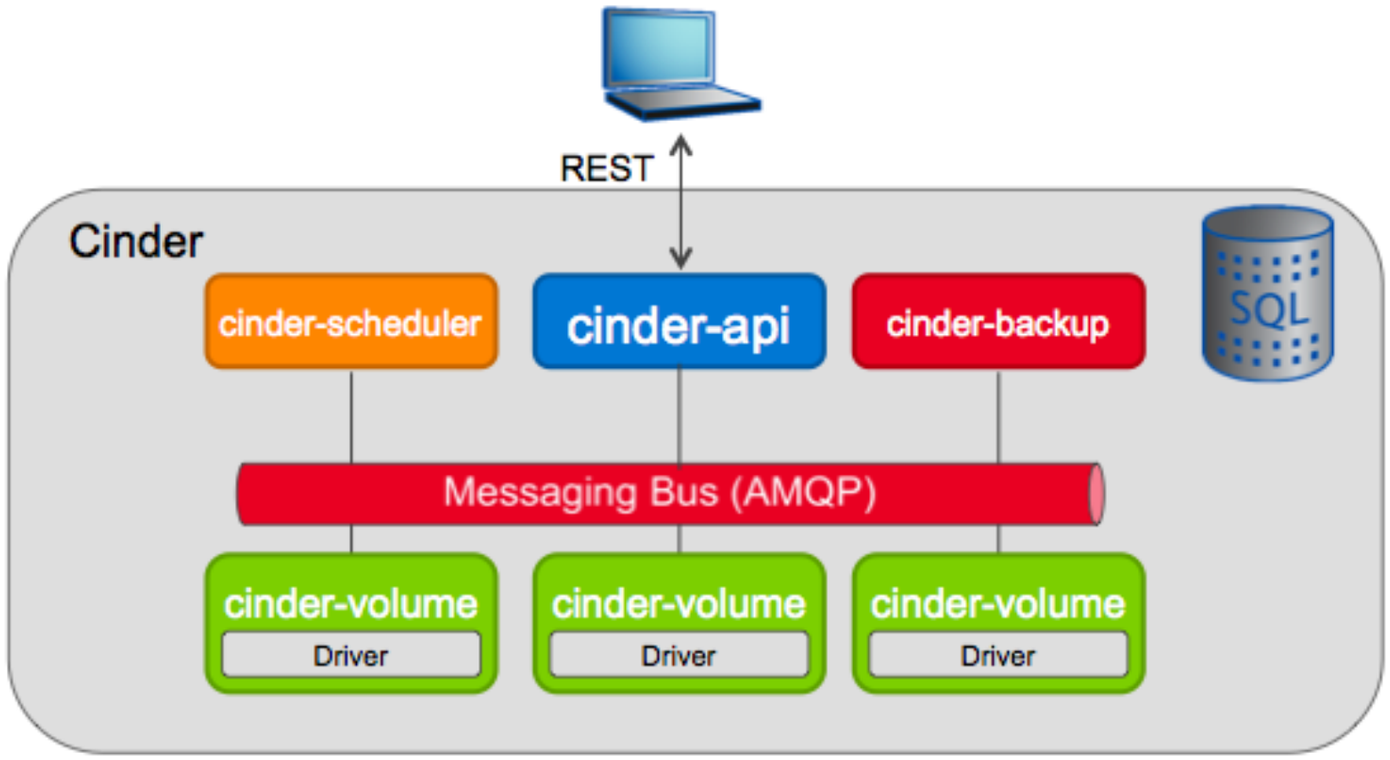
Marica Antonacci - INFN Bari

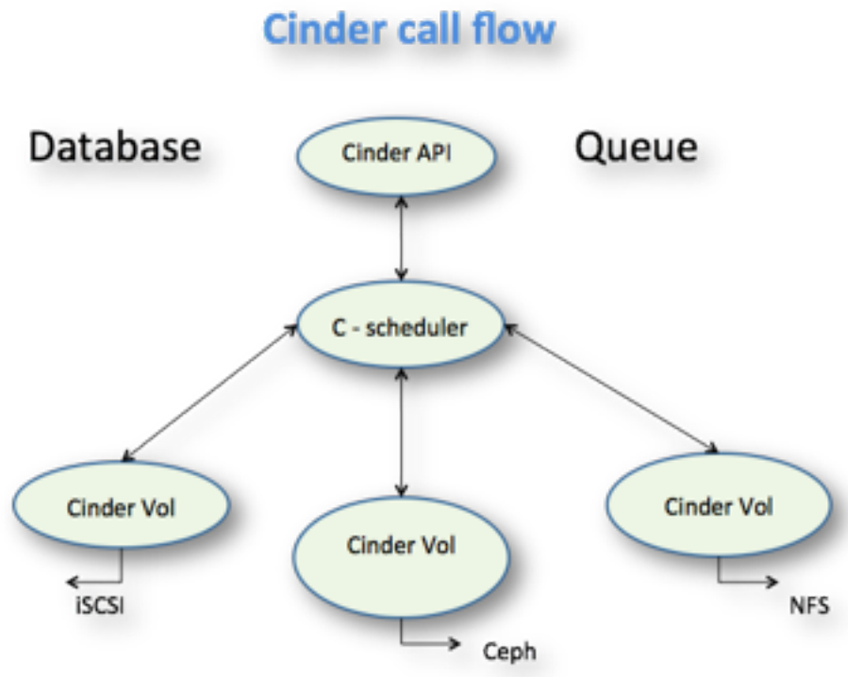
Tutorial Days di CCR
Napoli, 17-19 Dicembre 2014

Outline

- Multi-backend
- QoS & Rate-limiting
- Encryption
- Backup & Disaster-Recovery

Cinder: backend multipli





Un nodo cinder-volume può gestire uno o più backend

Openstack Block Storage Drivers Support Matrix: https://wiki.openstack.org/wiki/CinderSupportMatrix

La configurazione

- backend diversi (driver diversi)

cinder.conf

```
enabled_backends=lvm1,nfs1
[lvm1]
volume_driver=cinder.volume.drivers.lvm.LVMISCSIDriver
volume_backend_name=LVM_iSCSI
volume_group=cinder-volumes
[nfs1]
nfs_shares_config=${PATH_TO_YOUR_SHARES_FILE}
volume_driver=cinder.volume.drivers.nfs.NfsDriver
volume_backend_name=NFS
```

- backend dello stesso tipo (driver)

cinder.conf

```
enabled_backends=lvmdriver-1,lvmdriver-2
[lvmdriver-1]
volume_group=cinder-volumes-1
volume_driver=cinder.volume.drivers.lvm.LVMISCSIDriver
volume_backend_name=LVM_iSCSI
[lvmdriver-2]
volume_group=cinder-volumes-2
volume_driver=cinder.volume.drivers.lvm.LVMISCSIDriver
volume_backend_name=LVM_iSCSI
[lvmdriver-3]
volume_group=cinder-volumes-3
volume_driver=cinder.volume.drivers.lvm.LVMISCSIDriver
volume_backend_name=LVM_iSCSI_b
```

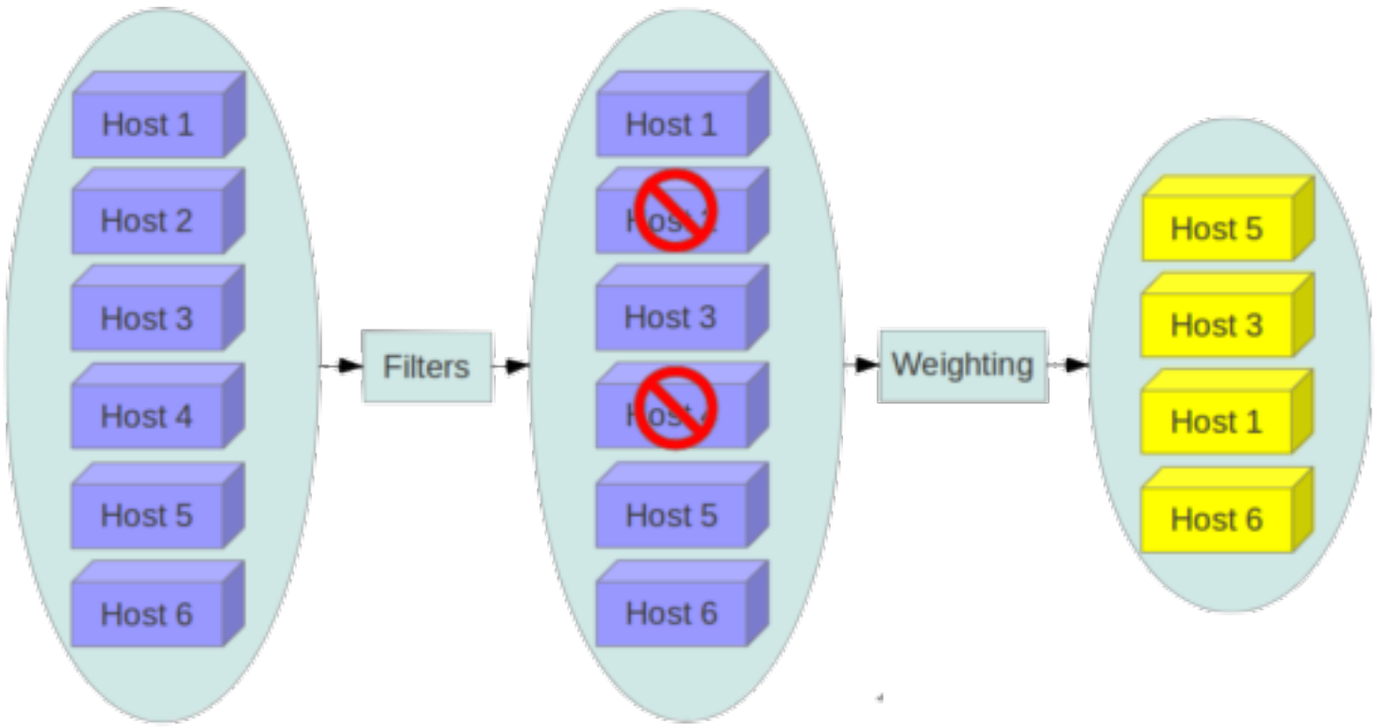
Cinder scheduler

- Per utilizzare i backend multipli, va abilitato il Filter scheduler:

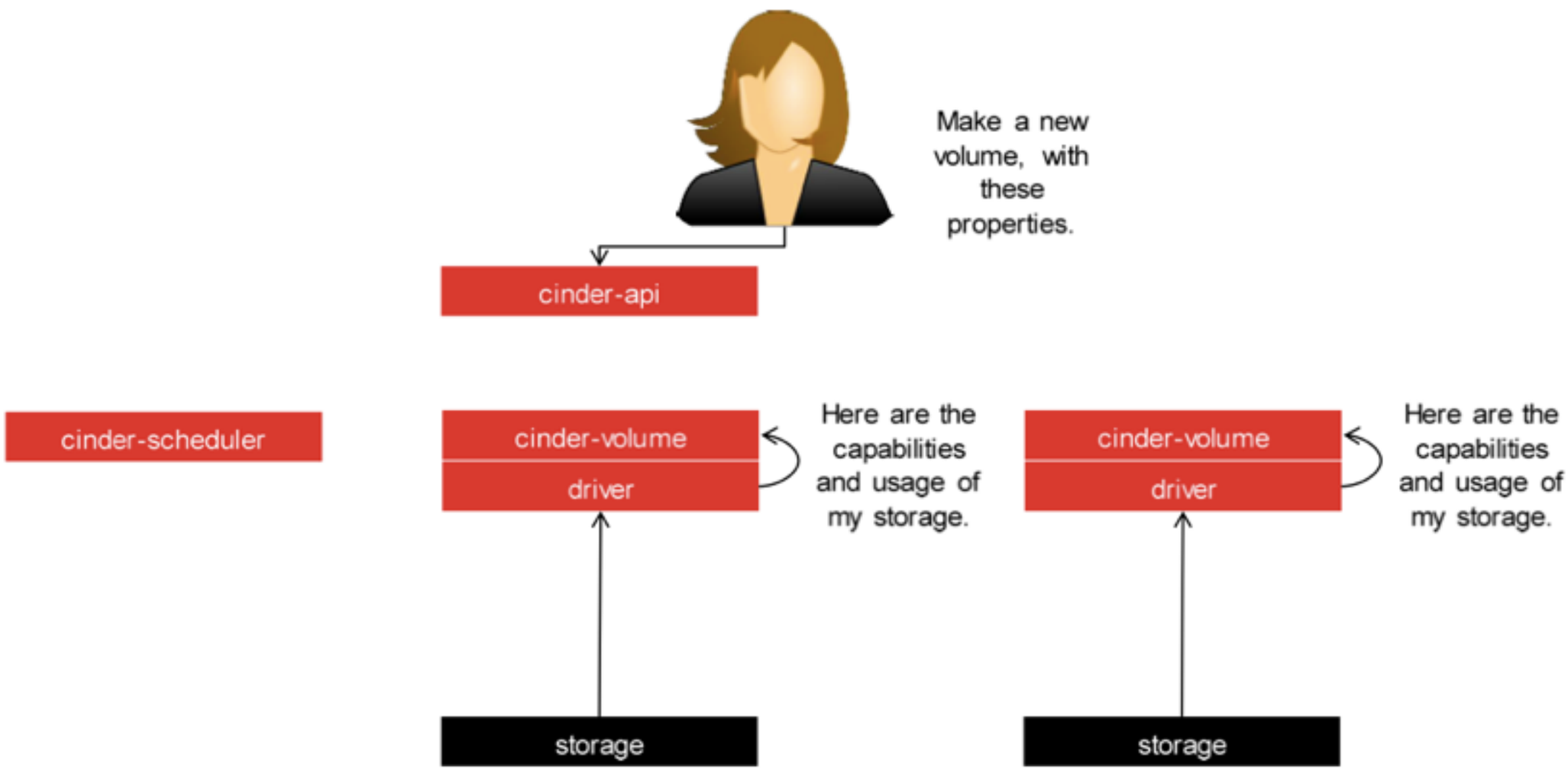
  ```
  scheduler_driver=cinder.scheduler.filter_scheduler.FilterScheduler
  ```

  - Filtra i backend disponibili. Default: `AvailabilityZoneFilter`, `CapacityFilter` e `CapabilitiesFilter`.

  - Associa un peso ad ogni backend filtrato. Default: `CapacityWeigher` option.



Cinder Scheduler



Cinder Scheduler

cinder-api

New volume request

cinder-scheduler

cinder-volume

driver

cinder-volume

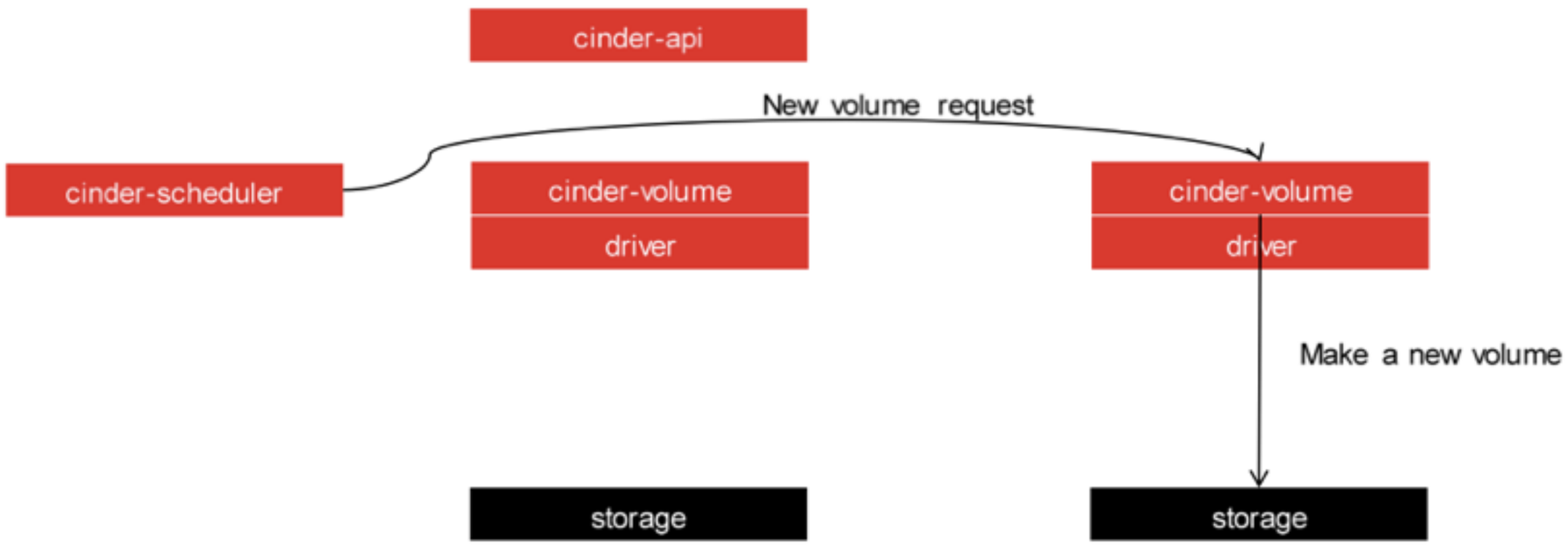
driver

1. What backends are there?
2. How are they doing?
3. What's the best candidate?

storage

storage

Cinder Scheduler



Uso dei volume_type

- I volume-type possono essere utilizzati per controllare dove i volumi verranno allocati:

```
# cinder create --volume_type lvm --display_name my-test 1
```

- Ogni volume-type contiene un set di coppie chiave-valore chiamati **extra-specs**. Queste informazioni sono usate dal Cinder-Scheduler per prendere decisioni sul placement dei volumi in base alle capabilities dei backend disponibili

```
# cinder type-create lvm
# cinder type-key lvm set volume_backend_name=LVM_iSCSI
# cinder extra-specs-list
```

```
+--------------------------------------+----------------+-----------------------------------------+
|                  ID                  |      Name      |                extra_specs              |
+--------------------------------------+----------------+-----------------------------------------+
| 0c91c16f-f3ab-493c-960b-75eb3f10d90e |    services    |  {u'volume_backend_name': u'LVM_SERVICES'}  |
| 143b23ca-e47c-401f-aa0b-8b9a408c72b7 |      data      |    {u'volume_backend_name': u'CEPH_DATA'}   |
| 5c1d93c7-64a9-496a-b0dc-2f675bddb057 | encrypted-data | {u'volume_backend_name': u'CEPH_DATA_ENCR'} |
```

Esempio di configurazione multi-tier

- Per esempio, assumiamo di avere 2 pool su Ceph che utilizzano storage device differenti:
- il pool "*cinder-sata*" usa un rack SATA
- il pool "*cinder-ssd*" usa un rack SSD

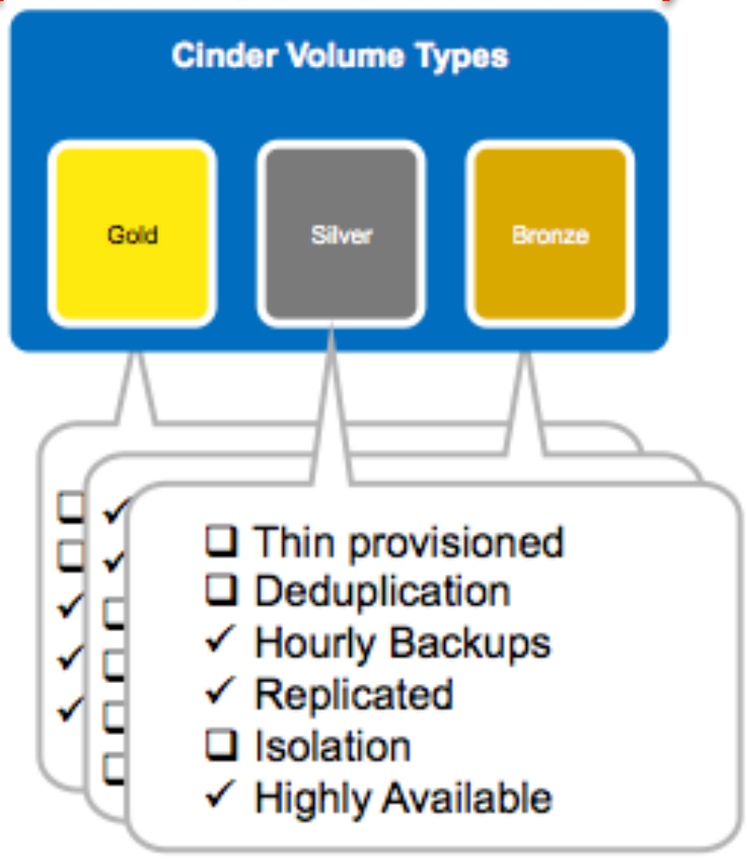
```
# Multi backend options

# Define the names of the groups for multiple volume backends
enabled_backends=rbd-sata,rbd-ssd

# Define the groups as above
[rbd-sata]
volume_driver=cinder.volume.driver.RBDDriver
rbd_pool=cinder-sata
volume_backend_name=RBD_SATA
# if cephX is enable
#rbd_user=cinder
#rbd_secret_uuid=<None>
[rbd-ssd]
volume_driver=cinder.volume.driver.RBDDriver
rbd_pool=cinder-ssd
volume_backend_name=RBD_SSD
# if cephX is enable
#rbd_user=cinder
#rbd_secret_uuid=<None>
```

Quality of Service

- I Volume type possono essere usati per fornire agli utenti differenti livelli (tier) di storage:

  - ✓ performance diverse (p.e. HDD tier, mixed HDD-SDD tier, o SSD tier),
  - ✓ resilienza (selezionando differenti livelli di RAID, o replica)
  - ✓ specifiche features (p.e. compressione, data-deduplication, etc.).

Volume-types
(i nomi sono arbitrari)



QoS e Rate limiting

- Feature introdotta in **Havana**
- Implementa il supporto **QoS** in Nova e Cinder (sfruttando il rate limiting già supportato in KVM e QEMU attraverso libvirt) - utile nel caso in cui lo storage non espone questa funzionalità
- Il limiting può quindi essere realizzato dal “frontend” (hypervisor) o dal “backend” (storage subsystem) o entrambi
- **Backend**: campi specifici definiti dal vendor:
  - ❖ HP 3PAR (IOPS, tput: min, max; latency, priority)
  - ❖ Solidfire (IOPS: min, max, burst)
  - ❖ NetApp* (QoS Policy Group)
  - ❖ Huawei* (priority)

**defined through extra specs*

Rate limiting options

- **Frontend** QoS options:
  - throughput
    - `total_bytes_sec`: the total allowed bandwidth for the guest per second
    - `read_bytes_sec`: sequential read limitation
    - `write_bytes_sec`: sequential write limitation
  - IOPS
    - `total_iops_sec`: the total allowed IOPS for the guest per second
    - `read_iops_sec`: random read limitation
    - `write_iops_sec`: random write limitation
- Il file di definizione della VM a cui viene agganciato il volume con *qos-specs* conterrà un campo xml extra "<**iotune**>" nella sezione <disk>. Es.

```
<iotune>
  <read_iops_sec>2000</read_iops_sec>
  <write_iops_sec>1000</write_iops_sec>
</iotune>
```

Rate limiting: cinder CLI (solo admin)

 # create qos specs

```
$ cinder qos-create <name> <key=value>
[<key=value> ...]
```

```
$ cinder qos-create high-iops consumer="front-end" read_iops_sec=2000
write_iops_sec=1000
+----------+-----------------------------------------------------------+
| Property |                           Value                           |
+----------+-----------------------------------------------------------+
| consumer |                         front-end                         |
|    id    |          c38d72f8-f4a4-4999-8acd-a17f34b040cb             |
|   name   |                       high-iops                           |
|  specs   | {u'write_iops_sec': u'1000', u'read_iops_sec': u'2000'}   |
+----------+-----------------------------------------------------------+
```

 # Associate qos specs with specific volume type

```
$ cinder qos-associate <qos_specs> <volume_type_id>
```

Esempi:
extra-specs + qos-specs

Mettiamo insieme un po' tutto:

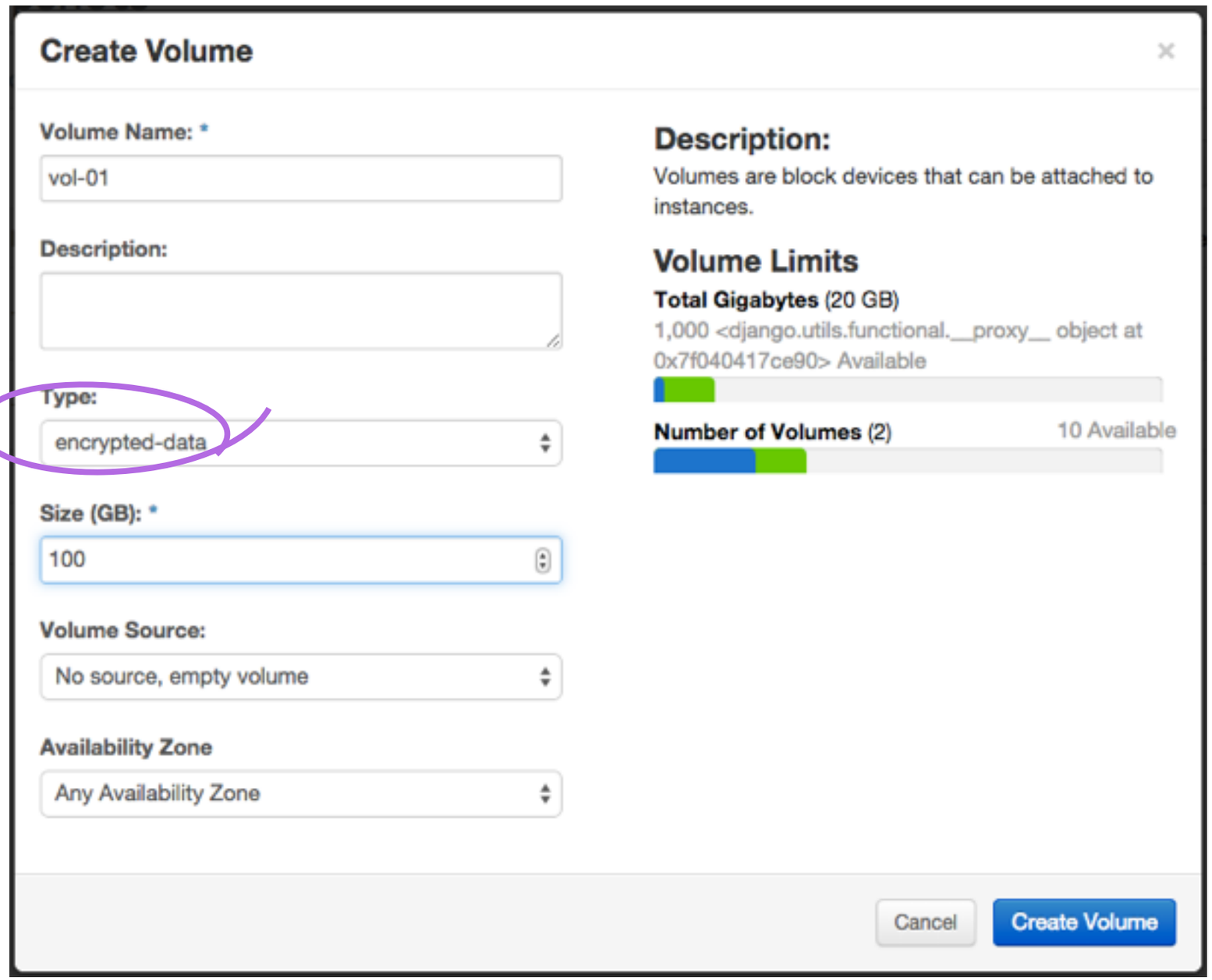
- volume-types,
- extra-specs,
- qos-specs

Volume Type	Extra Specs	QoS Specs
Gold	{netapp:disk_type=**SSD**, netapp_thick_provisioned=**True**}	{}
Silver	{}	{total_iops_sec=**500**}
Bronze	{volume_backend_name=**lvm**}	{total_iops_sec=**100**}

QoS “dinamico”

- **Volume-Retype**: consente di cambiare il tipo di volume dopo la sua creazione.
  - Questa funzionalità è utile per esempio per modificare il livello di QoS dinamicamente (nel caso in cui un volume sia sottoposto ad utilizzo pesante nel tempo e si renda necessario il passaggio ad un tier che offra un servizio migliore).
- In ***Icehouse***:
  - Purtroppo l’implementazione del blueprint non è completa
    - test **OK** tra backend di tipo LVM;
    - test FAILED tra backend RBD o tra LVM <—> RBD
  - Il comando volume-retype manca nella CLI per le API V1
    - bug: https://bugs.launchpad.net/python-cinderclient/+bug/1316939
    - patch: https://review.openstack.org/#/c/92768/

Creazione di volumi da dashboard



La dashboard consente la creazione dei volume-type, ma non permette al momento l'associazione con i backend, né la definizione di qos-specs.

Non sono neanche implementate le funzioni relative alla gestione degli encryption-type.

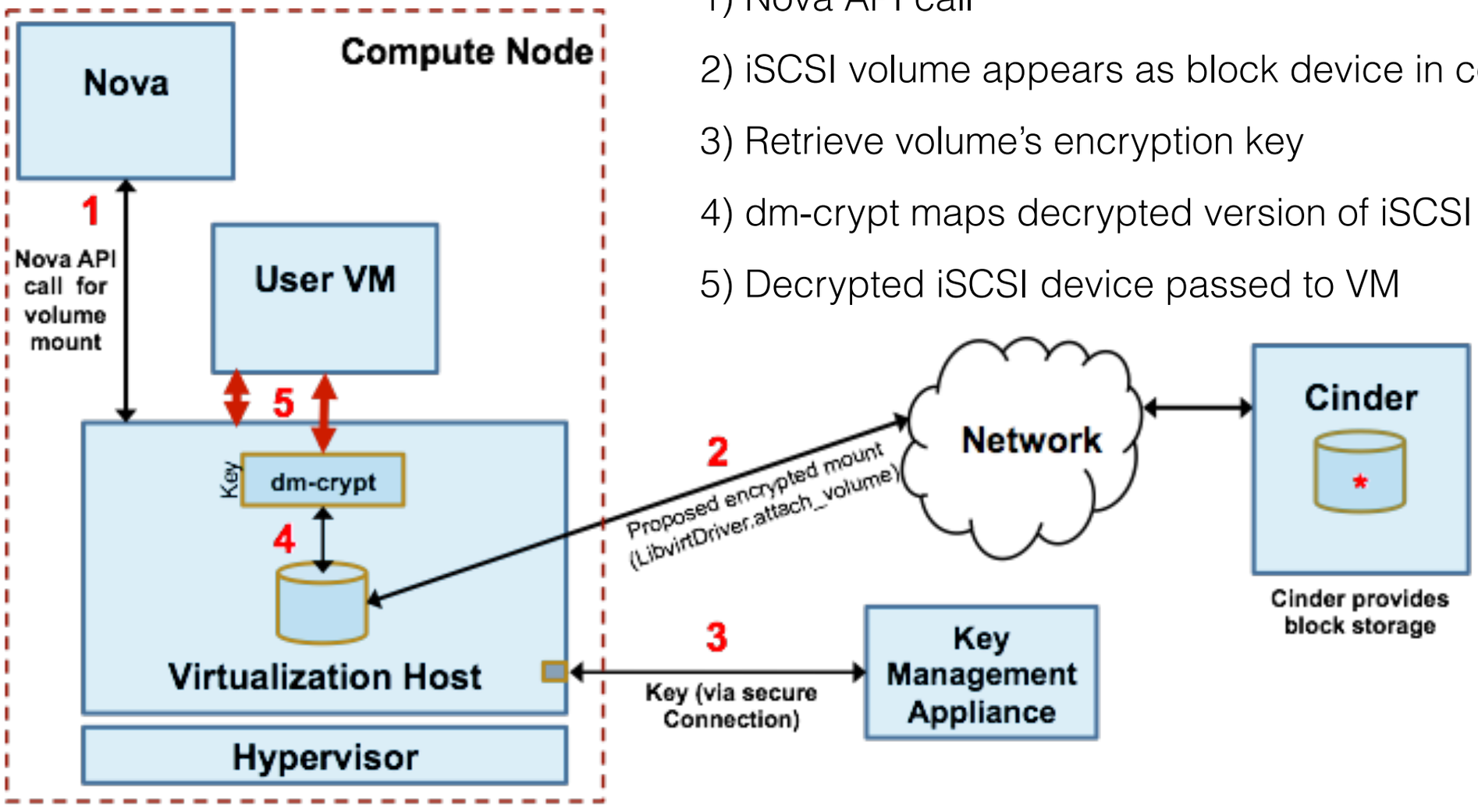
Volume encryption

- Questa funzionalità è stata introdotta nella release **Havana**
- Utilizzabile nel caso di backend ***LVM-iSCSI***
- Semplice da configurare, trasparente per l'utente finale (creare un volume cifrato richiede le stesse operazioni di un volume non cifrato)
- Il transito dei dati è sicuro
  - p.e. non è necessario usare IPsec per proteggere il traffico iSCSI
- Supporta:
  - le funzionalità esistenti in cinder (p.e. snapshot)
  - boot da volumi criptati
  - possibilità di scegliere il key-manager da usare per gestire le chiavi

Key Manager

- Il key manager di default è "**configuration-based**"
  - supporta singola chiave statica usata per tutti i volumi
    - da **NON** usare in produzione. La sicurezza dei dati dipende dalla segretezza della chiave
  - consigliato utilizzare un key-manager esterno (e.g. **Barbican**)
- Il key-manager espone un'interfaccia astratta che consente di integrare qualunque key-manager (incluso sistemi commerciali, p.e. Safenet, IBM, HP, etc.)

Block encryption



Steps for block encryption:

Preparatory Step: Cinder creates encrypted data volume

1) Nova API call

2) iSCSI volume appears as block device in compute host

3) Retrieve volume's encryption key

4) dm-crypt maps decrypted version of iSCSI volume

5) Decrypted iSCSI device passed to VM

https://wiki.openstack.org/wiki/VolumeEncryption

cinder encryption-types

- Estensione dell'astrazione del volume-type
- Utilizzo di metadata predefiniti
  - cipher: modalità di cifratura
    - e.g. aes-cbc-essiv:sha256 o aes-xts-plain64
  - key-length: dimensione della chiave in bits
    - e.g. 128 o 256
  - Provider: classe responsabile dell'attachment/detachment del volume criptato
    - `nova.volume.encryptors.cryptsetup.CryptsetupEncryptor`: uses "raw" cryptsetup
    - `nova.volume.encryptors.luks.LuksEncryptor`: uses LUKS extensions to cryptsetup
  - Control location: servizio che esegue l'encryption
    - '**front-end**' → Nova; '**back-end**' → Cinder
      - 'back-end' (i.e., encryption by Cinder) not yet implemented

CLI (solo admin) - example:

```
cinder encryption-type-create --cipher aes-xts-plain64 --key_size 512 --control_location front-end LUKS
nova.volume.encryptors.luks.LuksEncryptor
+--------------------------------------+---------------------------------------------+-----------------+----------+------------------+
|            Volume Type ID            |                   Provider                  |      Cipher     | Key Size | Control Location |
+--------------------------------------+---------------------------------------------+-----------------+----------+------------------+
| 90867515-543e-472d-b614-7816ca405fba | nova.volume.encryptors.luks.LuksEncryptor | aes-xts-plain64 |   512    |    front-end     |
+--------------------------------------+---------------------------------------------+-----------------+----------+------------------+
```

Ceph disk-encryption

- Ceph supporta dm-crypt
  - `# ceph-deploy osd --dmcrypt [--dmcrypt-key-dir KEYDIR] create|prepare HOST:DISK`
- creare pool su OSD criptati
- configurare in cinder.conf un nuovo backend associandolo al pool *encrypted*
- creare un volume-type specifico

Cinder backup

- Un backup è una copia del volume archiviata nell'Object Store

- Gestito da un servizio a parte: **cinder-backup** (non attivo di default)

- Driver configurabili:

  ➡ Ceph

  ➡ Swift

  ➡ IBM Tivoli Storage Manager

Backup driver Swift

Modificare il file cinder.conf - sezione DEFAULT

```
backup_driver=cinder.backup.drivers.swift

# The URL of the Swift endpoint (string value)
backup_swift_url=http://localhost:8080/v1/AUTH_

# Swift authentication mechanism (string value)
backup_swift_auth=per_user

# Swift user name (string value)
#backup_swift_user=<None>

# Swift key for authentication (string value)
#backup_swift_key=<None>

# The default Swift container to use (string value)
backup_swift_container=volumebackups

# The size in bytes of Swift backup objects (integer value)
backup_swift_object_size=52428800

# The number of retries to make for Swift operations (integer
# value)
#backup_swift_retry_attempts=3

# The backoff time in seconds between Swift retries (integer
# value)
#backup_swift_retry_backoff=2

# Compression algorithm (None to disable) (string value)
#backup_compression_algorithm=zlib
```

Backup driver Swift

Creiamo il backup con cinder

```
root@wn-recas-uniba-30:~# cinder backup-create --display-name test-bck 4b849af0-f989-4e95-9d79-60aede80a4ca
+-----------+--------------------------------------+
|  Property |                Value                 |
+-----------+--------------------------------------+
|     id    | 0542b982-45c5-4b39-8caf-930c05c12654 |
|    name   |               test-bck               |
| volume_id | 4b849af0-f989-4e95-9d79-60aede80a4ca |
+-----------+--------------------------------------+
```

e lo ritroviamo in Swift:

```
root@wn-recas-uniba-30:~# swift list volumebackups
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00001
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00002
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00003
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00004
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00005
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00006
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00007
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00008
volume_4b849af0-f989-4e95-9d79-60aede80a4ca/20140429134728/az_nova_backup_a1821891-c7a1-4a31-a962-9f9fb254ebd6-00009
```

Nota: cinder-backup consente di creare **backup** dei volumi con **replica** (sfruttando le capabilities dell'Object Storage). Se il backend Swift è distribuito geograficamente, allora è garantito anche il **disaster-recovery**.

Backup driver Ceph

Modificare il file cinder.conf - sezione DEFAULT

```
backup_driver=cinder.backup.drivers.ceph

# Ceph configuration file to use. (string value)
backup_ceph_conf=/etc/ceph/ceph.conf

# The Ceph user to connect with. Default here is to use the
# same user as for Cinder volumes. If not using cephx this
# should be set to None. (string value)
backup_ceph_user=cinder-backup

# The chunk size, in bytes, that a backup is broken into
# before transfer to the Ceph object store. (integer value)
#backup_ceph_chunk_size=134217728

# The Ceph pool where volume backups are stored. (string
# value)
backup_ceph_pool=backups

# RBD stripe unit to use when creating a backup image.
# (integer value)
#backup_ceph_stripe_unit=0

# RBD stripe count to use when creating a backup image.
# (integer value)
#backup_ceph_stripe_count=0

# If True, always discard excess bytes when restoring volumes
# i.e. pad with zeroes. (boolean value)
#restore_discard_excess_bytes=true
```

Backup di un volume RBD su Ceph

- Il driver è in grado di rilevare se il volume è un volume Ceph RBD

- in questo caso tenta di fare un backup incrementale e in caso di failure un backup full

- supporta il backup

  - ✓ all'interno dello stesso pool (not recommended)

  - ✓ tra pool diversi

  - ✓ tra cluster diversi

Ceph backup: under the hood

Workflow di creazione del primo backup di un volume

1. Create a base backup image used for storing differential exports
2. Snapshot source volume to create a new point-in-time
3. Perform differential transfer:

```
rbd export-diff --id cinder --conf /etc/ceph/ceph.conf --pool volumes volumes/volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba@backup.4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1.snap.1418722200.64 -

rbd import-diff --id cinder-backup --conf /etc/ceph/ceph.conf --pool backups - backups/volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base
```

Results in rbd:

```
# rbd -p volumes ls -l
NAME                                                                                SIZE PARENT FMT
PROT LOCK
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba                                      10240M          2
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba@backup.4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1.snap.1418722200.64 10240M          2

# rbd -p backups ls -l
NAME                                                                                                SIZE
PARENT FMT PROT LOCK
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base                                          10240M
2
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base@backup.4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1.snap.1418722200.64 10240M
2
```

Ceph backup: under the hood (2)

Workflow di creazione del backup successivo

1. Snapshot source volume to create a new point-in-time
2. Perform differential transfer using --from-snap:

```
rbd export-diff --id cinder --conf /etc/ceph/ceph.conf --pool volumes --from-snap backup.
4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1.snap.1418722200.64 volumes/volume-afa33905-0d87-42ff-
ad36-9c75fdcf09ba@backup.c255e3ca-f01b-4fe6-ad9f-af0524a7b531.snap.1418725945.25 -

rbd import-diff --id cinder-backup --conf /etc/ceph/ceph.conf --pool backups - backups/volume-
afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base
```

Results in rbd:

```
# rbd -p volumes ls -l
NAME                                                                                 SIZE PARENT FMT
PROT LOCK
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba                                       10240M          2
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba@backup.c255e3ca-f01b-4fe6-ad9f-af0524a7b531.snap.1418725945.25 10240M          2

# rbd -p backups ls -l
NAME                                                                                                  SIZE
PARENT FMT PROT LOCK
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base                                            10240M
2
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base@backup.4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1.snap.1418722200.64 10240M
2
volume-afa33905-0d87-42ff-ad36-9c75fdcf09ba.backup.base@backup.c255e3ca-f01b-4fe6-ad9f-af0524a7b531.snap.1418725945.25 10240M
2
```

Verso il disaster-recovery

- La funzionalità di cinder **backup-restore** consente di ripristinare lo stato di un volume all'interno della stessa istanza di Openstack.

- A partire da Icehouse esiste un'estensione delle API di cinder backup:

  - **import/export** dei metadata

  - occorre patchare il client *Icehouse* (https://review.openstack.org/#/c/72743)

- In **Juno** le API di cinder sono state ulteriormente arricchite per supportare la replica dei volumi

Cinder backup export

```
# cinder --os-volume-api-version 2 backup-export 4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1
+----------------+-----------------------------------------------------------------------+
|    Property    |                                 Value                                 |
+----------------+-----------------------------------------------------------------------+
| backup_service |                        cinder.backup.drivers.ceph                     |
|   backup_url   | eyJzdGF0dXMiOiAiYXZhaWxhYmxlIiwgIm9iamVjdF9jb3VudCI6IG51bGwsICJkZWxldGVkX2F0 |
|                | IjogbnVsbCwgInByb2plY3RfaWQiOiAiYWZiNDk3ODc5NmY1NDIyZDlhY2RlYzY0ZGE2YWFmNWYi |
|                | LCAidXNlcl9pZCI6ICIxM2IzMDBmOTkwZWM0ODI2YTIzY2QyNTIwODllNmNhZCIsICJzZXJ2aWNl |
|                | IjogImNpbmRlci5iYWNrdXAuZHJpdmVycy5jZXBoIiwgImF2YWlsYWJpbGl0eV96b25lIjogIm5v |
|                | dmEiLCAiZGVsZXRlZCI6IGZhbHNlLCAiY3JlYXRlZF9hdCI6ICIyMDE0LTEyLTE2VDA5OjMwOjAw |
|                | LjAwMDAwMCIsICJ1cGRhdGVkX2F0IjogIjIwMTQtMTItMTZUMTE6Mjc6MTkuMDAwMDAwIiwgImRp |
|                | c3BsYXlfZGVzY3JpcHRpb24iOiBudWxsLCAiaG9zdCI6ICJ3bi1yZWNhcy11bmliYS0zMCIsICJj |
|                | b250YWluZXIiOiAiYmFja3VwcyIsICJ2b2x1bWVfaWQiOiAiYWZhMzM5MDUtMGQ4Ny00MmZmLWFk |
|                | MzYtOWM3NWZkY2YwOWJhIiwgImRpc3BsYXlfbmFtZSI6ICJ2b2wtdGVzdC1iY2siLCAiZmFpbF9y |
|                | ZWFzb24iOiBudWxsLCAic2VydmljZV9tZXRhZGF0YSI6IG51bGwsICJpZCI6ICI0ZTUwZTk0OS0z |
|                |          ZGNkLTRmZjEtODllMC1hNmE5YzFiZWI1YzEiLCAic2l6ZSI6IDEwfQ==   |
|                |                                                                       |
+----------------+-----------------------------------------------------------------------+
```

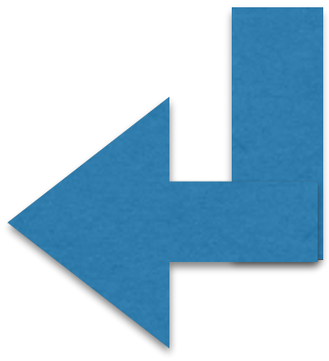
base64 --decode

```
{"status": "available", "object_count": null, "deleted_at": null, "project_id":
"afb4978796f5422d9acdec64da6aaf5f", "user_id": "13b300f990ec4826a23cd252089e6cad", "service":
"cinder.backup.drivers.ceph", "availability_zone": "nova", "deleted": false, "created_at":
"2014-12-16T09:30:00.000000", "updated_at": "2014-12-16T11:27:19.000000", "display_description": null,
"host": "wn-recas-uniba-30", "container": "backups", "volume_id": "afa33905-0d87-42ff-ad36-9c75fdcf09ba",
"display_name": "vol-test-bck", "fail_reason": null, "service_metadata": null, "id":
"4e50e949-3dcd-4ff1-89e0-a6a9c1beb5c1", "size": 10}
```

Cinder backup import

```
# cinder --os-volume-api-version 2 backup-import cinder.backup.drivers.ceph
eyJzdGF0dXMiOiAiYXZhaWxhYmxlIiwgIm9iamVjdF9jb3VudCI6IG51bGwsICJkZWxldGVkX2F0IjogbnVsbCwgInByb2plY3RfaWQiO
iAiYWZiNDk3ODc5NmY1NDIyZDlhY2RlYzY0ZGE2YWFmNWYiLCAidXNlcl9pZCI6ICIxM2IzMDBmOTkwZWM0ODI2YTIzY2QyNTIwODllNm
NhZCIsICJzZXJ2aWNlIjogImNpbmRlci5iYWNrdXAuZHJpdmVycy5jZXBoIiwgImF2YWlsYWJpbGl0eV96b25lIjogIm5vdmEiLCAiZGV
sZXRlZCI6IGZhbHNlLCAiY3JlYXRlZF9hdCI6ICIyMDE0LTEyLTE2VDA5OjMwOjAwLjAwMDAwMCIsICJ1cGRhdGVkX2F0IjogIjIwMTQt
MTItMTZUMTE6Mjc6MTkuMDAwMDAwIiwgImRpc3BsYXlfZGVzY3JpcHRpb24iOiBudWxsLCAiaG9zdCI6ICJ3bi1yZWNhcy11bmliYS0zM
CIsICJjb250YWluZXIiOiAiYmFja3VwcyIsICJ2b2x1bWVfaWQiOiAiYWZhMzM5MDUtMGQ4Ny00MmZmLWFkMzYtOWM3NWZkY2YwOWJhIi
wgImRpc3BsYXlfbmFtZSI6ICJ2b2wtdGVzdC1iY2siLCAiZmFpbF9yZWFzb24iOiBudWxsLCAic2VydmljZV9tZXRhZGF0YSI6IG51bGw
sICJpZCI6ICI0ZTUwZTk0OS0zZGNkLTRmZjEtODllMC1hNmE5YzFiZWI1YzEiLCAic2l6ZSI6IDEwfQ==
+----------+--------------------------------------+
| Property |                Value                 |
+----------+--------------------------------------+
|    id    | 9a3d360a-8cb2-42a3-9289-7279cf6b67cc |
|   name   |                 None                 |
+----------+--------------------------------------+
```

```
# cinder backup-show 9a3d360a-8cb2-42a3-9289-7279cf6b67cc
+-----------------+--------------------------------------+
|     Property    |                Value                 |
+-----------------+--------------------------------------+
| availability_zone |                nova                |
|     container   |               backups                |
|     created_at  |     2014-12-16T13:52:42.000000       |
|    description  |                None                  |
|    fail_reason  |                None                  |
|        id       | 9a3d360a-8cb2-42a3-9289-7279cf6b67cc |
|       name      |            vol-test-bck              |
|    object_count |                None                  |
|       size      |                 10                   |
|      status     |            available                 |
|     volume_id   |        0000-0000-0000-0000           |
+-----------------+--------------------------------------+
```



Altre funzionalità avanzate

- Migrare i volumi tra backend differenti (admin API)

```
cinder migrate <volume_id> <target host>
```

- Estendere la dimensione di un volume

```
cinder extend <vol-id> <newsize>
```

- Trasferire un volume da un tenant ad un altro

```
cinder transfer-create <volume_id> #TenantA
```

```
cinder transfer-accept <transfer_id> <auth_key> #TenantB
```