Qemu Replication Design Sketch

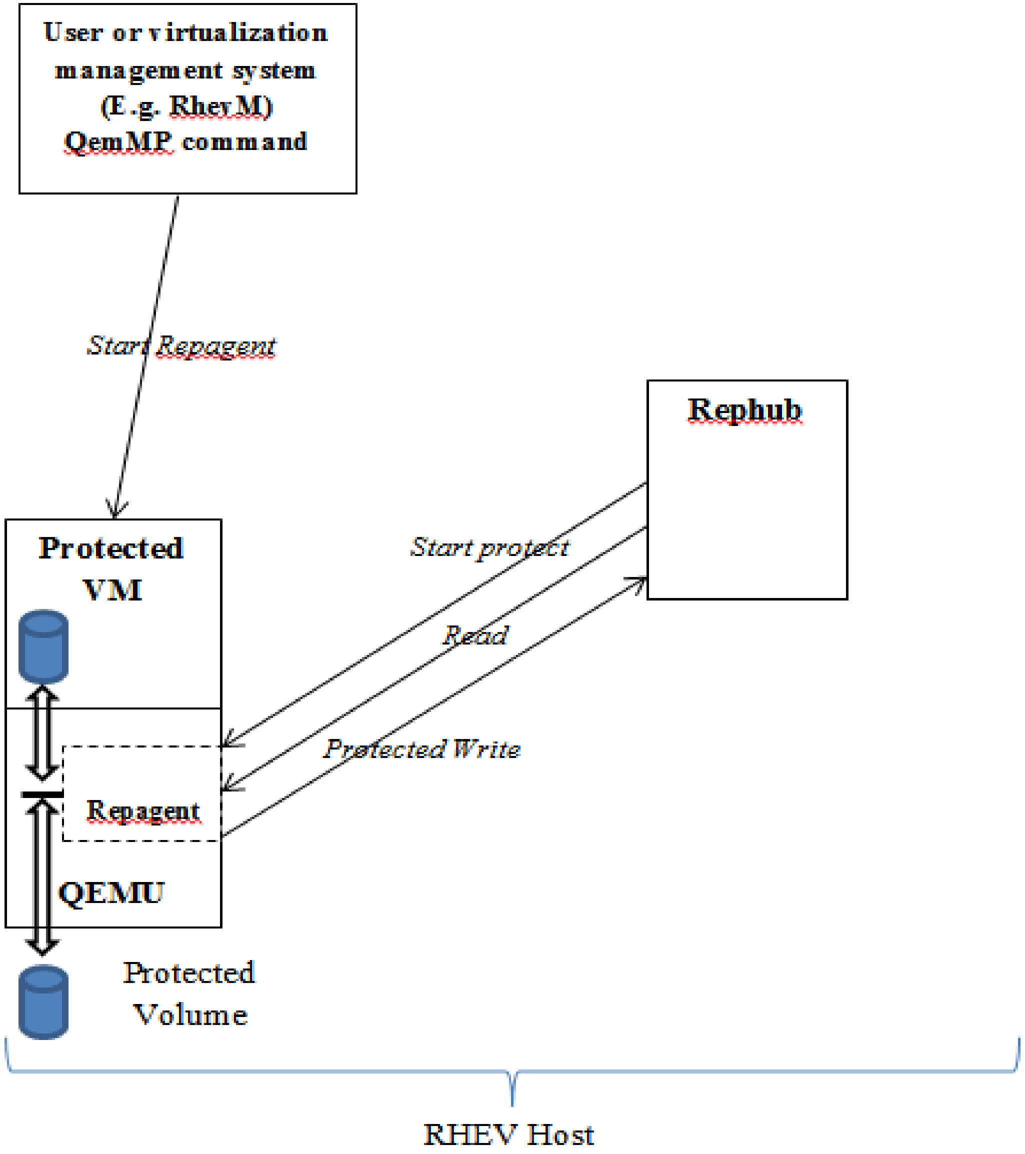
Terms

FO – Fail Over
FOT – Fail Over Test
Promote – Apply history of Ios from the journal to the recovery volume.

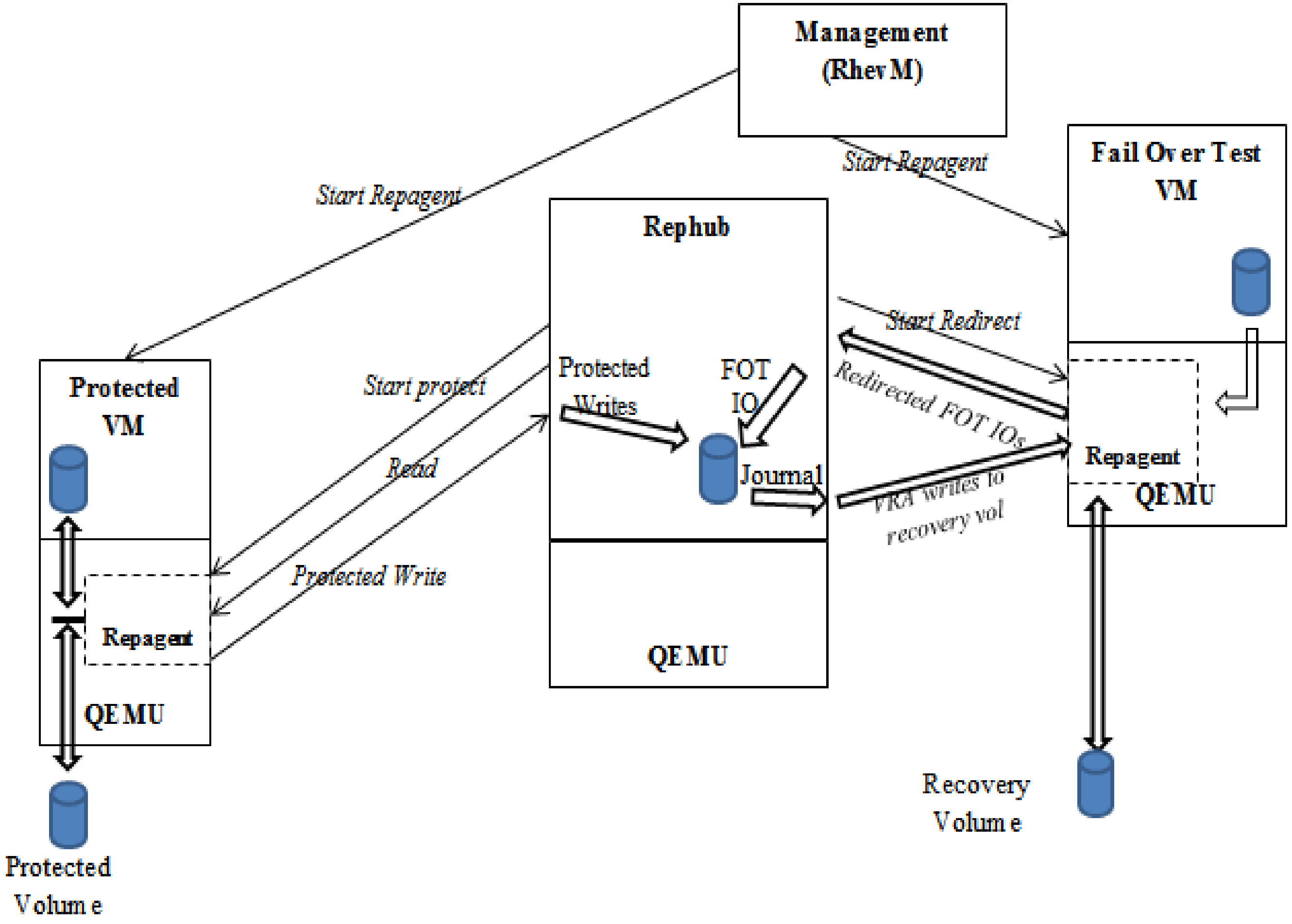
Main entities

- RepAgent
  - Located in Qemu code
    - It is a block driver – a filter inside the Qemu storage stack.
  - Protects a volume in Qemu. Sends protected volume writes to RepHub
  - Enables IO redirection in FOT machines
    - Capture and processing of any IO made by the FOT machine.
  - Enables Rephub to read/write to a volume
- RepHub
  - The replication system – in charge of replicating all the protected volume writes.
  - All RepAgents on a host connect to it
  - Manages the RepAgents
    - Manages bitmaps
    - Reads protected drive for sync
  -

Block diagram - Protecting



Block diagram – Protecting + Fail Over Test



* Note that Rephub here is only a sample implementation for clarity.

Flows

Protect flow in Repagent

1 VM1 has one volume (Vol1). It is unprotected

2 User descides to protect VM1

2.1 User runs Rephub

2.2 User sends a QMP command to VM1 to start repagent

2.2.1 The properties include VRA IP and port

3 RepAgent in VM1 handles the command

3.1 Adds the repagent block driver to the Vol1 stack in the block layer

3.2 Connects to Rephub with a TCP socket

3.3 Tells the Rephub the location of all its volumes – in this case Vol1.

4 Rephub sends RepAgent a “StartProtect” command for Vol1.

5 The Rephub starts a sync process

5.1 Repagent allows Rephub direct read access to the protected drive (via Rephub commands sent on the socket)

6 Repagent starts to replicate IO writes on the protected VM -

6.1 Each guest (VM1) write is delivered to the Rephub. The write is not delayed.

The flow above lacks a bitmap component. We chose to postpone the use of a bitmap to a later stage.

In the current stage the bitmap will be maintained inside the Rephub.

In general the bitmap tracks all guest IOs until they are acknowledged by the Rephub.

Protect sample flow in Rephub

1 When the user started to protect Vol1 on VM1:

1.1 Rephub creates a new volume and uses it as a recovery volume for Vol1.

1.2 It starts to mirror protected Ios to the recovery volume.

1.3 If the VRA has no room for another Volume (when it has 27 volumes attached to it already), then ZVM creates a new VM (diskbox1) and attaches the volume to it.

2 TheVRA writes the replicated protected Ios to Vol1R – either directly or via network to a diskbox.

Fail Over Test

Fail over test is a flow where the replication system creates a recovery VM and runs it in parallel to the protected production VM, without stopping the protection.

1 The protected site continues as usual – VM1 is protected and sends all Ios to Rephub

2 Current status of the Rephub is that it has a recovery volume Vol1R attached to it and it protects VM1 with it.

3 Management system creates a new VM called VM1R – a recovery VM for VM1

3.1 Sends a QMP command to VM1R Qemu to start repagent

3.2 Repagent in VM1R connects to the Rephub

4 Rephub connects the recovery volume Vol1R to VM1R

4.1 From this point Rephub continues to protect VM1 by writing protected data to Vol1R via Repagent in VM1R instead of writing directly to Vol1R.

5 Rephub tells Repagent in VM1R to redirect Vol1R Ios to VRA.

5.1 Redirect means that any read or write made by VM1R to Vol1R would be captured by Repagent, sent to Rephub, and only then answered.

6 From now on all IOs made in VM1R to Vol1R are intercepted by Repagent and sent to the Rephub for handling.

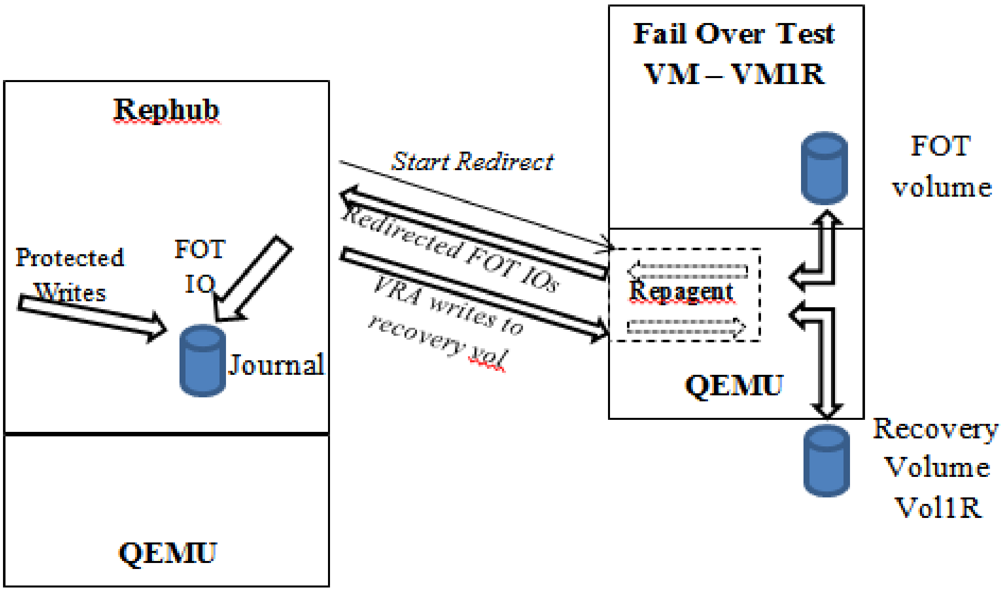
6.1 IO writes are written to a cache or a journal

6.2 IO reads are read from the journal and from the recovery volume Vol1R

6.3 This is needed for 2 main reasons:

6.3.1 The Rephub may keep a journal holding history of the replication, and in that case any read by VM1R needs to be checked first in the journal.

6.3.2 The recovery volume Vol1R is still used for protecting the production VM VM1 – so the FOT VM VM1R must not write directly to it.



Fail Over

In general, FO is similar to FOT, but without the need to redirect the FO VM Ios to Rephub:

1. Management system creates VM1R as a FO machine
2. Rephub writes all saved IO history to Vol1R.
3. Management system connected Vol1R to VM1R and starts the VM.