1. Introduction

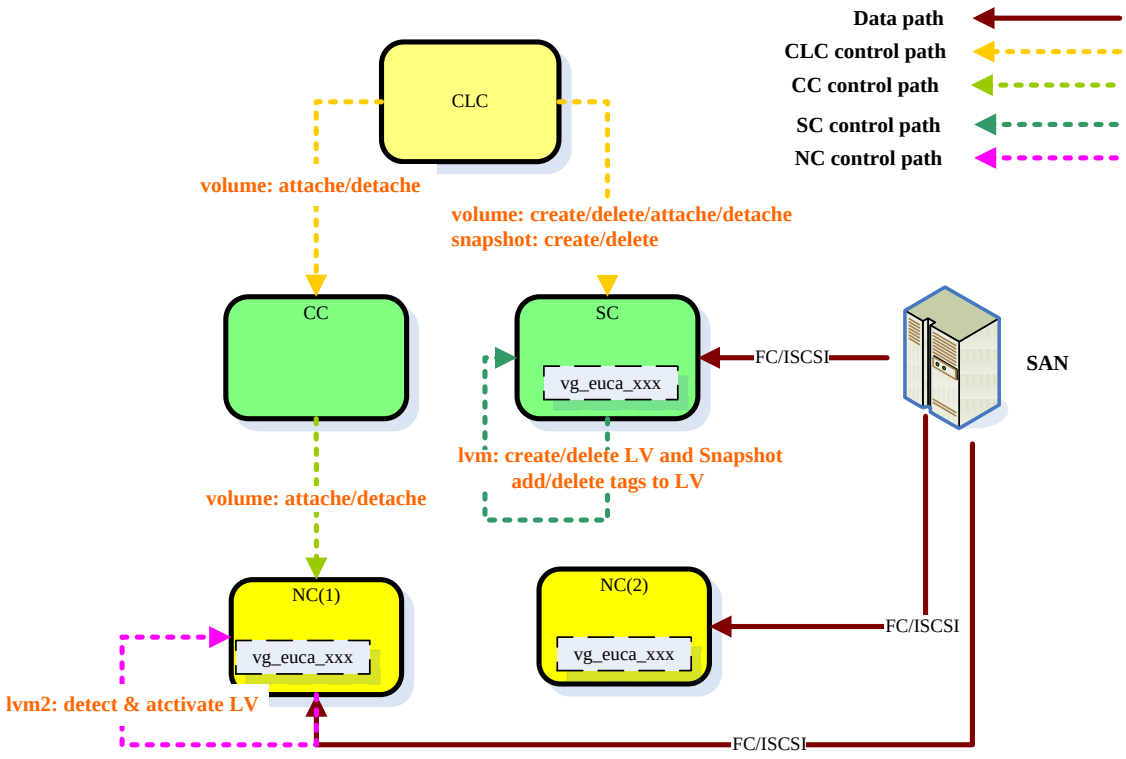
The generic SAN adapter is a effort for providing a common solution to leverage all existing SAN devices in the data center and relieve the concerns that push eucalyptus storage controller in data path between VM and SAN devices in which storage controller is easy to be the bottleneck of I/O performance and stabilities

In the solution of generic SAN adapter, the eucalyptus storage controller will never interact with SAN controller for creating LUN and exporting the LUN to node controller.

Then there is no limitation by eucalyptus platform for how the SAN connecting to storage controller and node controller. both FC and ISCSI are supported. Any kind of connection redundancy and performance tuning solution provide by the SAN vendor also can be supported in the generic SAN adapter.

2. Solution Overview

Below is a diagram to depict the data & controlling path in the generic SAN adapter



1) How the SAN connect to SC/NC

Eucalyptus **don't manage** the data connection between SAN and NCs/SC in the generic san adapter.

The SAN device can be connect to SC and all NC nodes through FC or ISCSI protocol .

Typically one big LUN (or several LUNs) should be created and attach to SC/NCs

2) Use LVM2 for creating EBS volume on the LUNs of SAN

All LUNs attached to SC/NCs will be added into one volume group, the volume group is created at first time the generic adapter get started and can be extended by adding more LUN of the SAN device.

Each EBS volume will be a LV in the VG.

snapshotting is implemented by using snapshot LV.

3) LVM2 synchronization & Lock

As the one LUN is shared and attached to SC and NCs , the LVM metadata (LV information) should be synchronized between SC and NC. tool **"vgscan"** in LVM2 is used to synchronize the LVM information between SC and NCs.

Any write operation of LVM (for example create vg, create/remove lv, lv change) should be only allowed to be done in SC. and the NCs are only allowed to read the changes of the VG and activate the LV assign to this NC. A customized LVM lock are implemented to do this.

4) LV access controlling

As the LUNs of SAN are shared by all NCs, so all NCs can see the LVs created by SC. there should be a access controlling mechanism that make sure the specific NC can only activate and use the LV assigned to it.

LVM2 already offer controlling the access to specific VG or LV. for detail please refer to LVM conf http://linux.die.net/man/5/lvm.conf.

In the generic SAN adapter, **host tag** and activation filter are used to control the LV activation

5) Fencing

There are some extreme cases in which one NC can't be reached through network but LUN device in this NC (for example through FC network) can still be accessed by it.

In this case, it cloud happens that CC/CLC identify the VMs running in this NC are failed then all EBS volume attached these VMs will be detached and be available to attach to other VM. if this happen, the EBS volume actually are used by two VM concurrently which cloud cause the data corruption.
To prevent this happen, a fencing mechanism is need which can guarantee the isolated NC will be forced to detach EBS volumes.

6). Processes

In SC, the generic SAN adapter is responsible for:

- ✧ create/extend volume group when attach new SAN lun to SC/NCs.
- ✧ create/delete LV when SC gets command from CLC to create/delete EBS volume
- ✧ create/delete snapshot LV when SC gets command from CLC to create/delete EBS snapshot
- ✧ create/remove host tag from LV when gets command from NC to export & unexport EBS volume.

In NC, there are some changes for the generic SAN adapter:

- ✧ the LVM in NCs nodes are configured with external locking type, with this configuration, NC node is not allowed to create or change the LV, also the NC are only allowed to activate the LV which has the host tag for this NC.
- ✧ when NC gets attaching command from CC, it issue a export volume command to SC then scan the LVM to synchronize the LV information with SC. then activate the LV which ais allow to activate in this node and attach it to the VM.

3. Details of Design

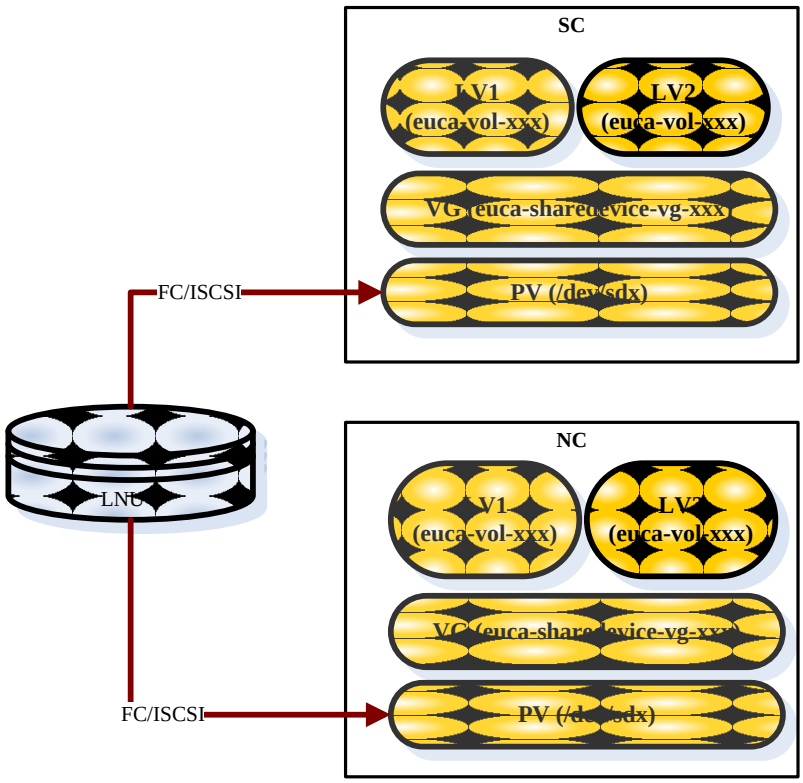
Currently, the generic SAN adapter comprises of 3 parts (lack of fencing process, which will be added soon)

- ✧ New block storage manager i
- ✧ A customized LVM2 lock
- ✧ Patches to node controller

1). Block Storage Manager

Called "CLVM" which can be plugged into storage controller. the CLVM block storage manager is very similar to DAS manager which manipulate the volume group through lVM2 CLI.

Manage Volume Group



The block manager will create the PV and Volume Group (name pattern **euca-sharedevice-vg-xxxxx**) when it's first get started

During the running period, block storage manager will create LV or delete LV, add host tag to LV, remove host tag from LV according to requests from CLC or NC.

Host Tag - In order to export volume that allow a NC to activate the LV, block manager will add a host tag (the IP of NC) into the LV. the IP of the NC will be retrieved from the IQN string in the parameters of export volume request. so in the NC node, The iscsi initiatorName should be configured as "iqn.1994-05.com.redhat:xxx.xxx.xxx.xxx (ip of the node controller)

Snapshot - generic SAN adapter doesn't use the snapshot features in the SAN device, it leverage the snapshot ability of LVM2. and the snapshot file also be cached in storage controller 's directory $EUCALYPTUS/var/lib/eucalyptus/volumes.

API s (not all listed here) -

API	description	Details of the implementation
creatVolume	Create a EBS volume	Create a LV through the LVM2 CLI and keep it unactivated
DeleteVolume	Delete a EBS volume	Delete the LV through the LVM2 CLI
exportVolume	Export the EBS volume	Create a host tag on the LV, **the host tag is the IP of the node controller which are retrieved from the IQN string (parameter of exportvolume request from NC),** SC return a connection string to NC. the return string has pattern like: "user,auth_mode,lun,password,iface,host,iqn" In this block storage manager, the **password=CLVM a**nd **iface=CLVM**
unexportvolume	Unexport the EBS volume	Remove the host tag from LV.
createSnapshot	Create snapshot for a EBS volume	1) Activate the LV and 2) create a temporary snapshot LV for this LV 3) copy it to file in directory $EUCALYPTUS/var/lib/eucalyptus/volumes. 4) Delete the snapshot LV 5) Deactivate the LV
deleteSnapshot	Delete the snapshot of a EBS volume	Delete the snapshot file in $EUCALYPTUS/var/lib/eucalyptus/volumes

HA Consideration - this block storage manager can support HA deployment in which two SCs are registered in one cluster because it's out of data path between SAN and VM.

As the snapshots are implemented by LVM2 LV snapshotting and the cached snapshot file are stored in directory $EUCALYPTUS/var/lib/eucalyptus/volume. to support HA, this directory must be a file system mount point and the file system is created on the SAN which can be seen by two SCs.

when CS is enabled, the block storage manager will synchronize the LVM information (use vgscan) and remount the filesystem to $EUCALYPTUS/var/lib/eucalyptus/volumes

2) Customized LVM2 Lock

Purpose of this Customized LVM2 Lock is to control the operation on the volume group, for example if the SC and NC create a lV concurrently on the same volume group, this cloud cause the metadata of volume group is not consistent , then need a lock to guarantee this operation is controlled and not corrupt the meta data in volume group.
In generic san adapter, to simplify the use case, operation like LV creating/deleting/changing are not allowed in the NCs, this can be implemented by adding a read only lock in NCs.

Customized Lock Implementation - LVM2 has several built-in locks to control the operation in the single server or cluster environment. also, it offers the ability to plug-in external lock implementation. for detail please refer to LVM2 manual page:
http://linux.die.net/man/5/lvm.conf
Below are some points of the customized external locking

1) lf target of the operation is a volume group whose name start with "**euca-sharedevice-vg**", then only read operation is allowed, any writing operation will be blocked

2) otherwise, it apply a file lock (the default lock) to this operation.

Why hard-coding the volume group prefix - this is just for simplifying the implementation, in the future, a new call between NC and SC can be added to allow the NC get the volume group name from SC**.**

3) Patches in NC

In NC, some patches added for

- ✧ Synchronize the LV information
- ✧ Control the LV activation and access
- ✧ Attach the LV to VM

Control LV activation and Access -

This is implemented by using the LVM2 built-in host tag and volume activation filter.

below are example of configurations in lvm.conf.

Add host tag:

tags {
*   hosttags=1*
*   @192.168.1.101 { }*
}
with above configuration, the NC has a host tag "192.168.1.101"

Add Activation Filter:

Activiation {
*   ....*
*   volume_list=[@*]*
}

This configuration allow the NC node to activate a LV which has host tag "192.168.1.101"

Synchronize the LV information & Attach to VM -

For generic san adapter, NC will use **vgscan** to synchronize the LV information.

In the time of NC getting command from CC to attach a EBS volume:

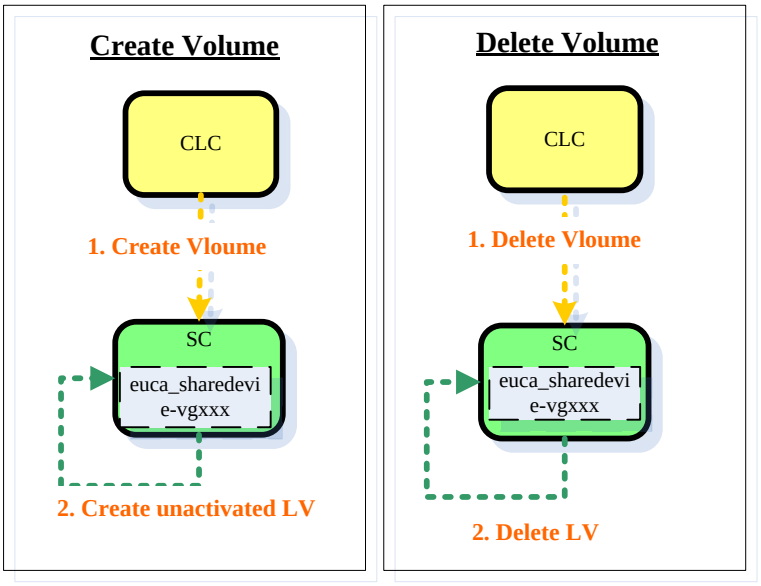
1) NC will issue request to SC for exporting a ebs volume. SC will return a connecting string to NC with pattern " user,auth_mode,lun,password,iface,host,iqn",

in generic san adapter, the password in connection string is "CLVM" and iface will be "CLVM" ,iqn is the name of the LV.

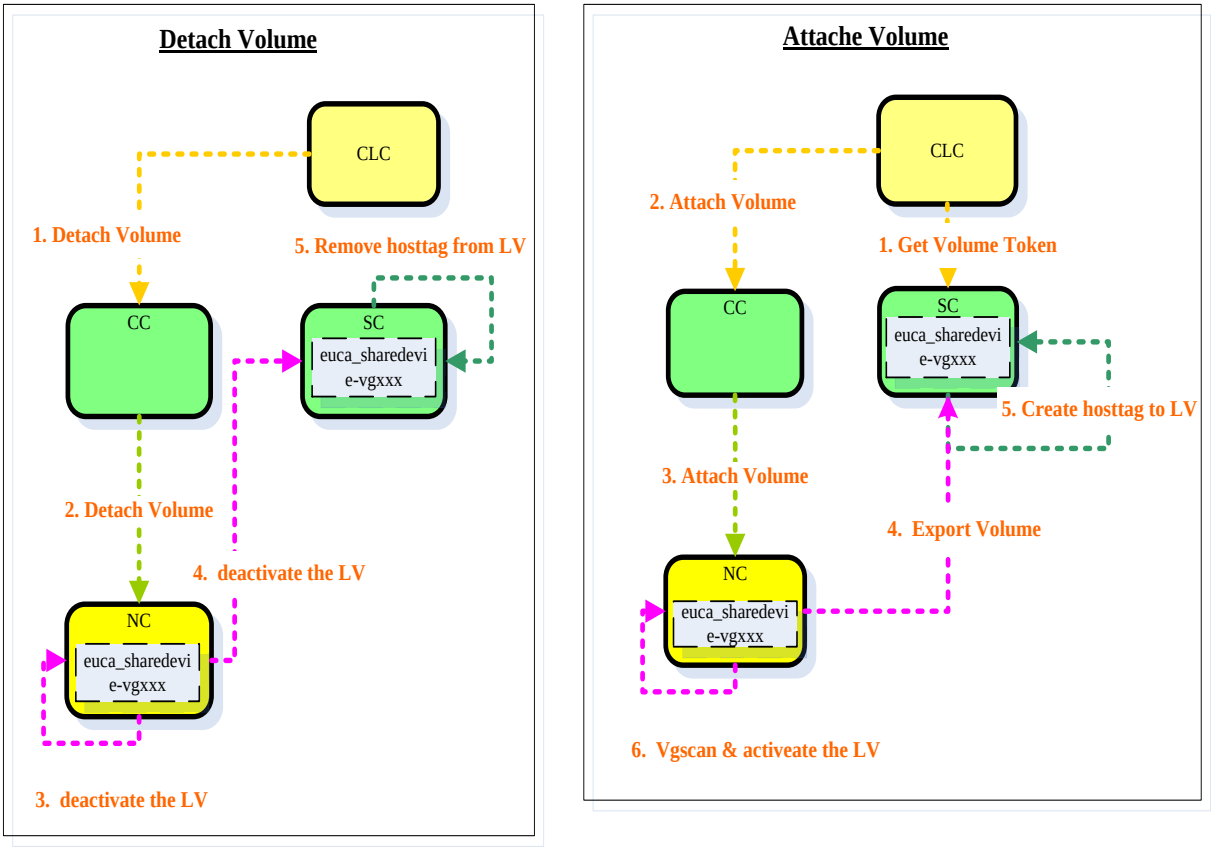
2) Then NC will first use "vgscan" to update the LV information, if it find the LV, it will activate it and get full path of this LV, then pass the path to libvirt for attaching it to VM.

4). Processes Scenarios

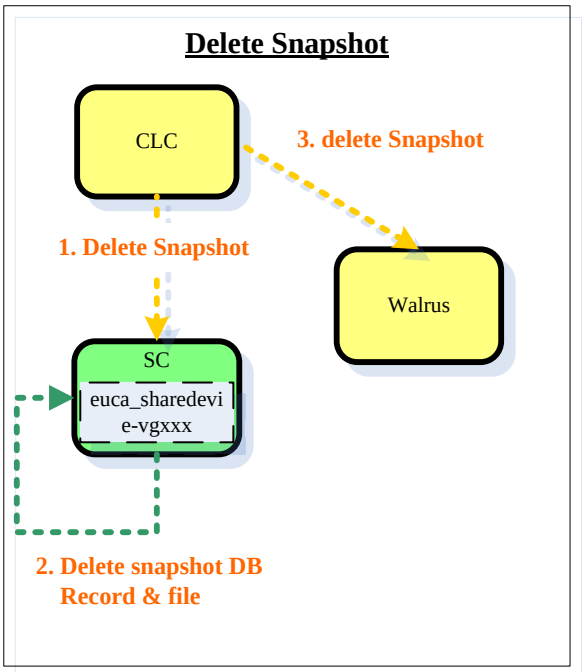
1) Create/delete Volume

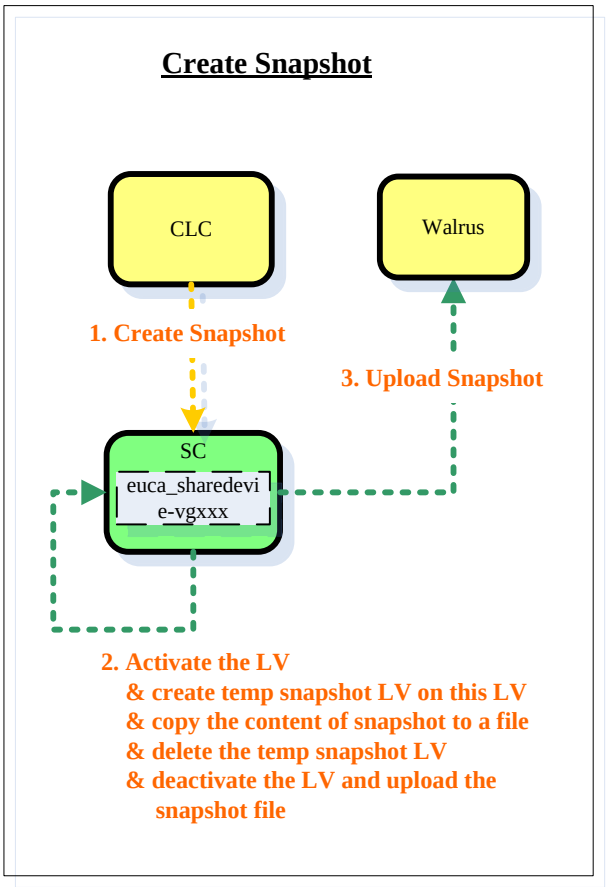


2) Attach/Detach Volume

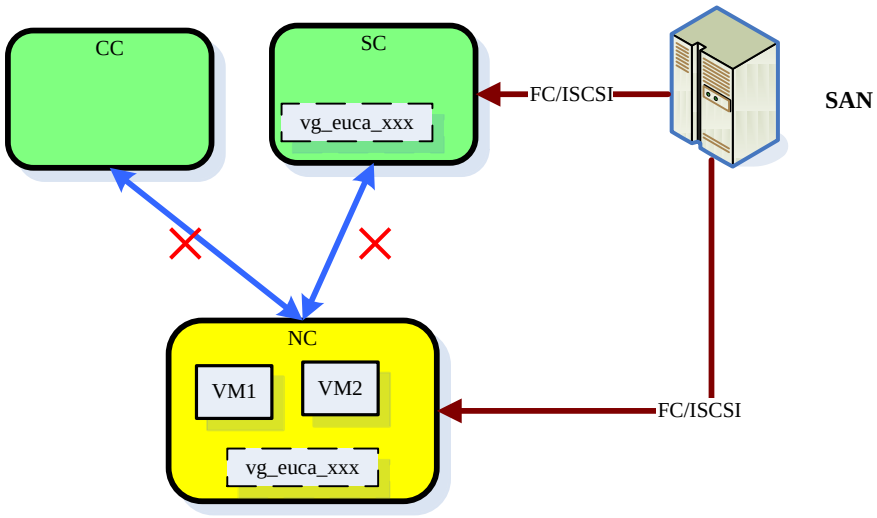


3) Create/Delete Snapshot





4. Fencing



This is a case that the FC/ISCSI connection is good but the network between NC and CC/SC is broken.

In this case, it cloud happens that SC force to detach EBS volumes from VM1 and VM2, but this command can't be sent to NC then VM1 and VM2 will still use this LV because the FC/ISCSI connection is not broken.

Fencing is a mechanism that guarantee EBS volume not be use by the NC any more in this case

In the generic SAN adapter, two possible fencing solution

solution 1:

In all NC, implement a cron job (or a fencing deamon) to periodically checking the LV status (this works because LV status can be retrieved from the disk not through the broken network). if it find a LV is deleted or host tag in this LV was deleted, the VM should stop to use this LV.

solution 2:

This need hardware support, once the SC identify a NC is failed, it should send command to the hardware interface (FC switch or ISCSI switch) to disconnect FC/ISCSI connection between SAN and the failed NC.

5. Limitations & Concerns

There are some limitations with this SAN adapter

1) currently this proposal is for KVM and XEN. need to study if it works for VMWare.

2) potential fragment of volume group. as we may frequently create and destroy LV (different size) in one volume group. not sure if this will cause the fragment issue and down the I/O performance. (possible we can set relative large PE size of volume group to avoid this)