Solution Overview

for

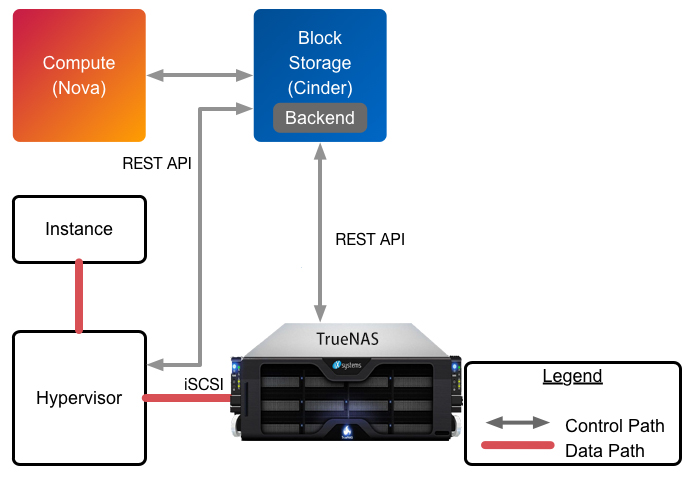
iXSystems Cinder Project

March, 2016

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Name** | Solution Overview for  iXsystems Cinder Project | **Customer Name** | iXsystems |
| **Prepared by** | Jitendra Pawar | **Version** | 1.0 |
| **Approved by** | Sunu Engineer | **Date** | 01-03-2016 |

# Solution Overview

This document provides an overview of iXsystems Cinder Solution. The solution involves implementing Cinder volume driver for backend storage appliance. The driver plug-in enables TrueNAS to serve as backend storage for OpenStack Cinder (block storage service). Current solution will implement driver for iSCSI protocol.



# Cinder Operations:

The driver is targeted for OpenStack Kilo release. For Kilo release, Cinder drivers need to support following listed operations:

1. Volume Create / Delete
2. Volume Attach / Detach
3. Snapshot Create / Delete
4. Create Volume from Snapshot
5. Get Volume Stats
6. Copy Image to Volume
7. Copy Volume to Image
8. Clone Volume
9. Extend Volume
10. Volume Statistics

* driver\_version
* free\_capacity\_gb
* reserved\_percentage
* storage\_protocol
* total\_capacity\_gb
* vendor\_name
* volume\_backend\_name

# Implementation overview

The driver will be implemented in python and will communicate with backend storage using FreeNAS 9.3 STABLE REST api. The driver will handover control / delegate operation to backend storage using the corresponding REST api(s). In case there is no backend support for a specific operation, the driver will handle it gracefully and return an appropriate message.

The driver will define class ‘iXsystemsISCSIDriver’ and export the required functionalities which will override Cinder base class functionalities.

In addition to listed operation list, Cinder driver needs to implement below listed functions by default.

1. create\_export(), ensure\_export() and remove\_export()

To export volume details post create volume operation

1. do\_setup() and check\_for\_setup\_error()

For custom setup and error checking post driver load operation

1. initialize\_connection() and terminate\_connection()

For exporting connection information before volume attach / detach operation

In order to use this solution, Cinder user needs to enable iSCSI driver in Cinder configuration file

Example:

*“volume\_driver=cinder.volume.drivers.ixsystems.iscsi.TrueNASiSCSIDriver”*

User needs to include required parameters in cinder configuration file *(/etc/cinder/cinder.conf).* Configuration parameters include - User credentials, Appliance connection details, Storage parameters, etc.

--oooOooo--