How to build BaaS using OpenStack and Storware Backup & Recovery

www.storware.eu

Table of content

Introduction

This article will present key aspects of building the Backup as a Service (BaaS) solution based on OpenStack and Storware Backup and Recovery. We will cover the general approach of protecting data and how to connect different components to achieve a backup solution that end-users can use without disturbing the cloud itself.

It must be first noted that while, in general, cloud instances should be as stateless as possible, and there is a common approach that you should not backup instances but rather the application itself - the reality tends to be quite different. Quite often, users treat cloud instances in the same way as another virtual machine and change their state manually. This results in an obvious requirement for VM-level backup, which we will analyze in the following sections.

Agents need to be installed and maintained to protect specific applications. Having to protect hundreds or thousands of virtual environments usually makes agent-based solutions unmanageable. Of course, in specific cases, such as databases - it may be required to use them, but generally, you should avoid an agent-based approach whenever possible.

Backup of a single instance

Let's start with the basics - how to protect individual instances in OpenStack. Like any other virtual environment, the instance itself typically has two main components - metadata and data. Metadata, so the instance configuration (CPU, storage, networking, etc.) in OpenStack is offered by multiple services such as Cinder for storage definition or Nova for compute-related aspects. The solution must always collect metadata with each Backup to have a consistent view of the instance. Without this step, one could discover that a new volume has been connected to the instance at some point, and the data was never backed up.

The key element is data, which is stored on the OpenStack volumes. OpenStack supports several types of volumes, including temporary (ephemeral) volumes, nova volumes, and,

finally, cinder-based volumes. While the ephemeral volumes can be skipped, and cinder volumes (keeping real data) need to be protected, nova volumes also may, in rare cases, be of our interest. In rare cases, users may also use (unknowingly) Nova volumes for application data.

Disk-attachment backup strategy

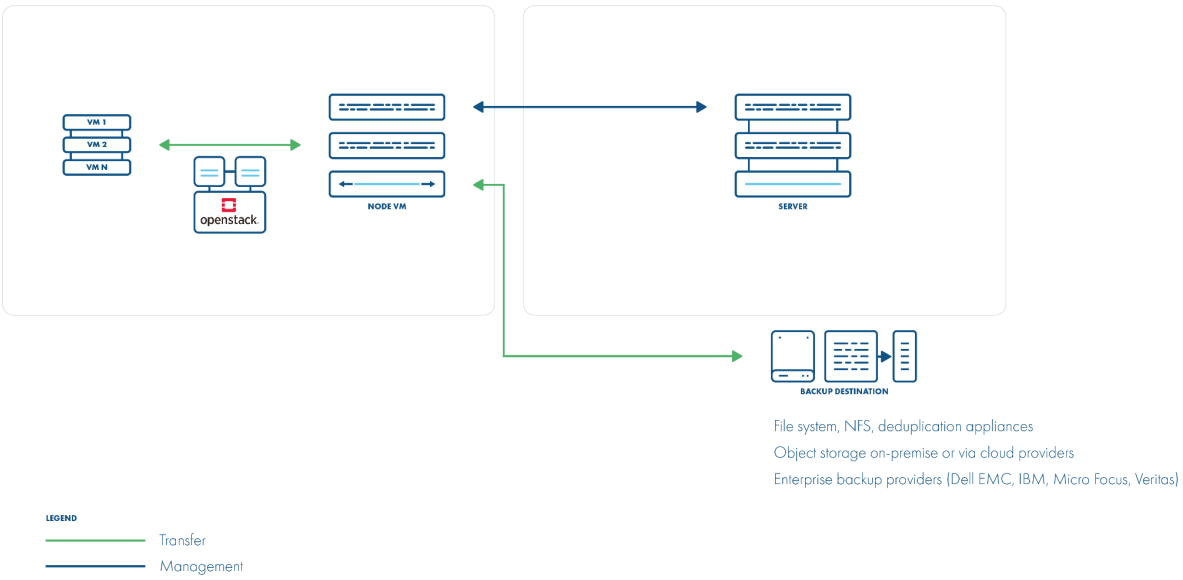
Depending on the storage configuration, we may need to use a different strategy for how the instance will be backed up. Starting with the most common case - just cinder-based volumes and using only the main APIs offered by OpenStack, we can protect instances using the Disk Attachment method.

This method assumes that there is a helper proxy VM with the backup software used to read data from the volume snapshots of the protected instances. The data effectively is read directly from the underlying storage. However, by default, there is no way to have information about the changed blocks since the last backup, so the application would need to scan for the changed blocks. There is, however, an option to use Ceph RBD snapshot-difference API. In such a scenario, creating incremental backup would look almost exactly as with the Disk-Attachment method - volume snapshots would be mounted on the proxy. Still, instead of reading the whole data, we can call Ceph RBD API to fetch changes since the last snapshot directly from Ceph monitors. This requires network access to the monitors and always leaving the last snapshot so that there is a possibility to compute changes since the last Backup.

Some vendors (and this also includes Storware Backup & Recovery) also provide the option to scan for changes using information about checksums of regions in the volumes - if the checksum of fixed size, i.e., 64 MB is different from the previous one - the whole 64 MB needs to be read. This process consumes additional time, but the advantage is that you can run incremental backups on any storage provided by the Cinder.

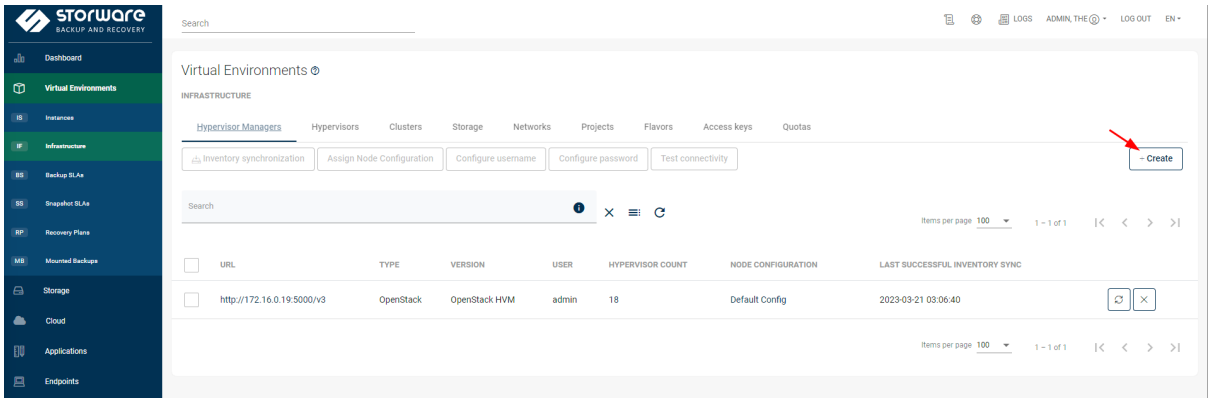
A simpler approach is to just do a full backup instead and leave the deduplication layer to remove duplicated blocks - without needing a separate mechanism for incremental backups.

While CBT is not available in OpenStack, you can see that there are ways to address this issue in one way or the other. Cinder offers a transparent layer that allows using all kinds of disk arrays. The disk-attachment backup strategy covers various scenarios - including non-Ceph storage usage.

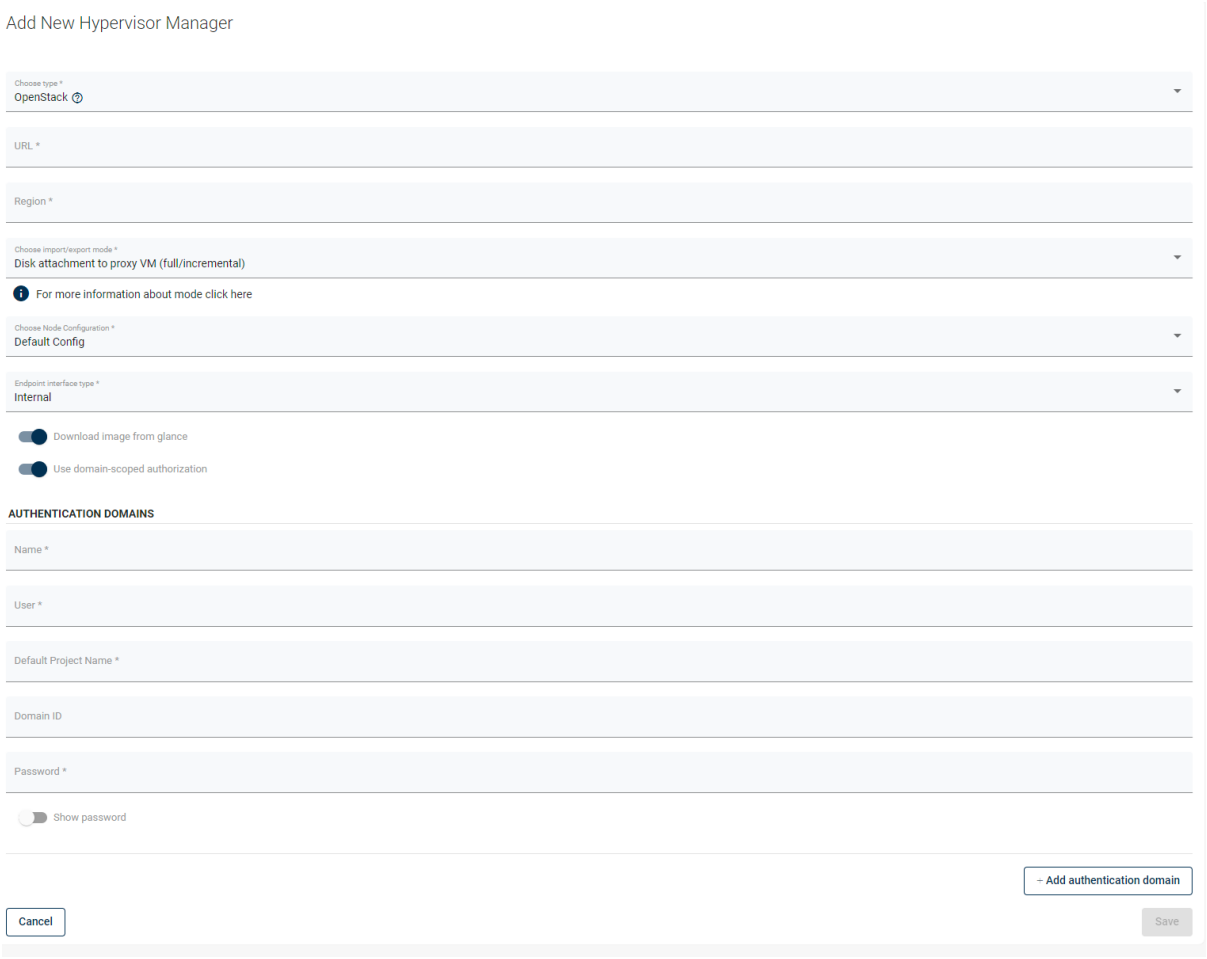


Example

To add OpenStack, first, click Create in the Hypervisor Managers tab of the infrastructure section (virtual environments)



Choose the type OpenStack



Fields:

- **URL** - URL of the Keystone endpoint in version 3. (i.e., `https://keystoneHost:5000/v3`)
- **Region** - Region value configured in the OpenStack
- **Import/export mode** - leave it as a **Disk attachment to proxy VM** or **Disk attachment changed block tracking** (if you want to scan for changes for incremental backups on non-Ceph storage)
- **Endpoint interface type** - choose which endpoint type should the Storware Backup & Recovery communicate with; depending on the environment, it can be Public, Internal, or Admin
- **Download image from Glance** - if turned on, during full Backup, also glance image from which the instance was created is backed up
- **Use domain-scoped authorization** - if turned on, authorization credentials are only used for the domain with which they were provided (you can provide multiple credential sets using the **Add authentication Domain button**)

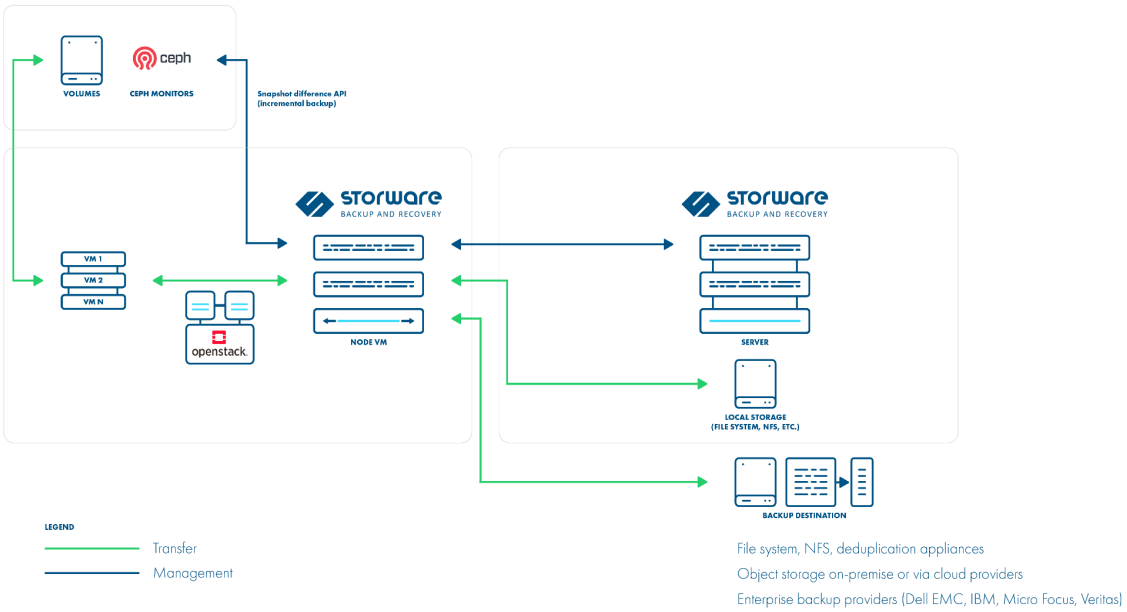
Authentication domain section:

- **Name** - OpenStack domain name
- **User** - username used for authorization
- **Default** project name - default project used for authorization scoping

- **Domain ID** - OpenStack ID of the provided domain
- **Password** - password for the chosen user

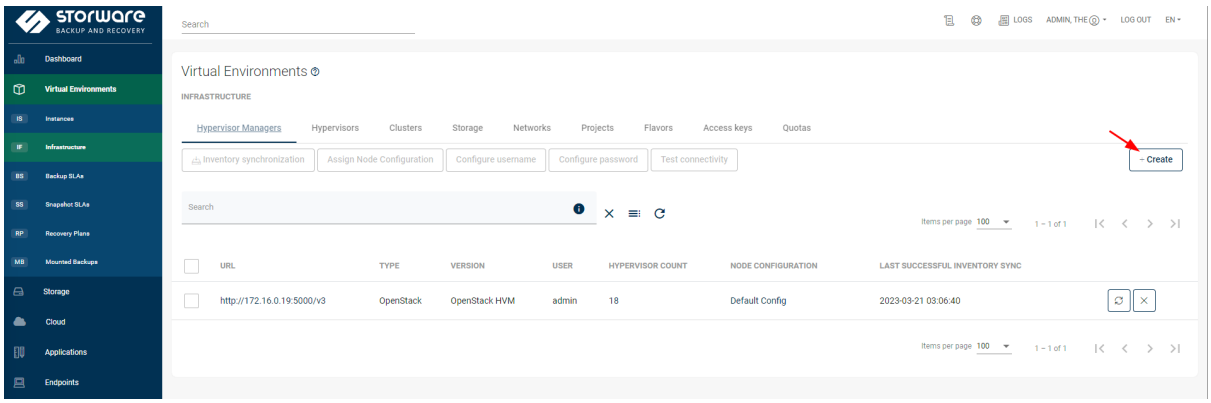
After providing all the values and clicking save, you will be asked whether you want to do the inventory synchronization. It is safe to proceed with this process. Once the task is done, all OpenStack instances should be visible in the Storware Backup & Recovery.

Disk attachment with Ceph RBD

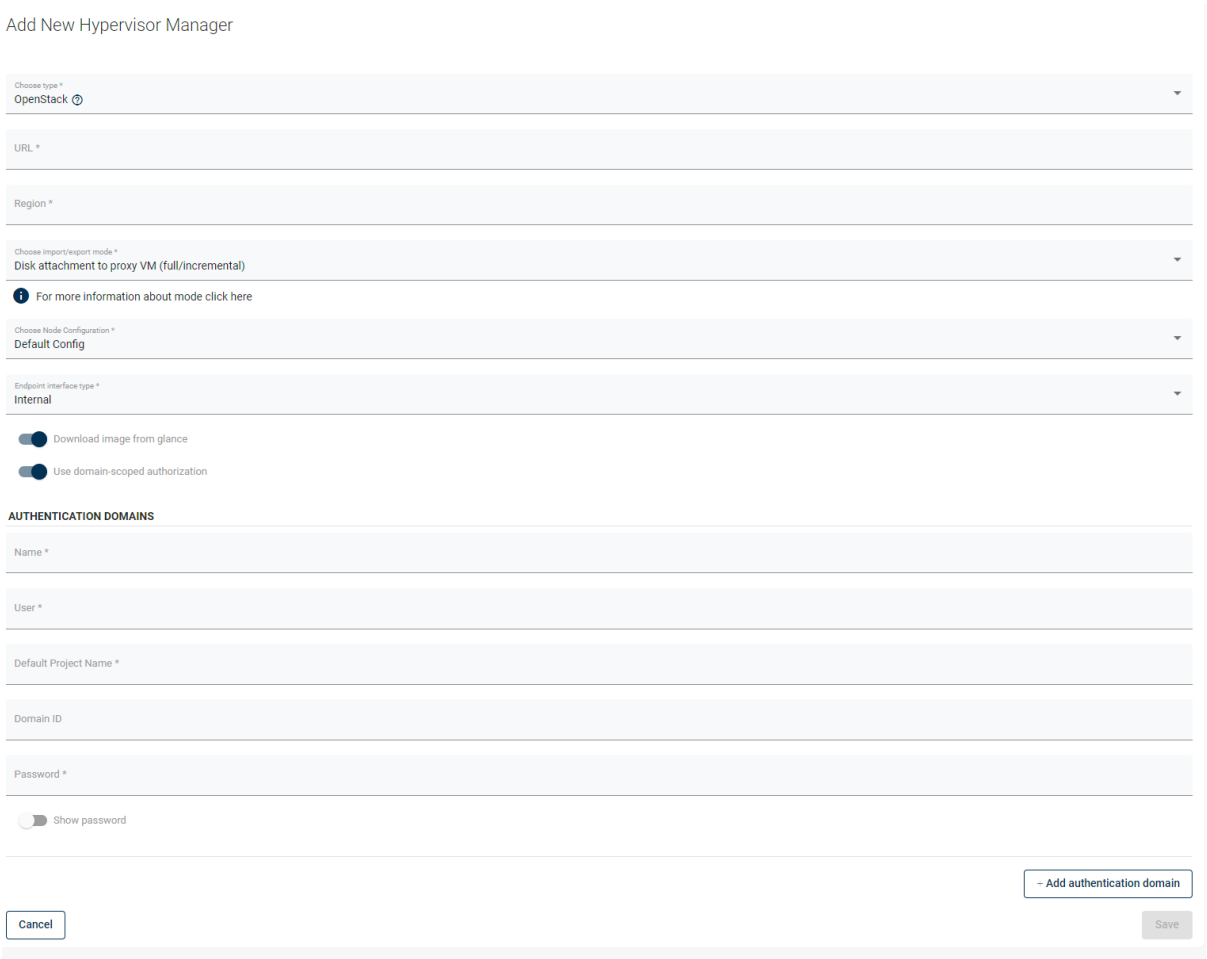


Example

To add OpenStack, first, click Create in the Hypervisor Managers tab of the infrastructure section



Choose the type OpenStack



Fields:

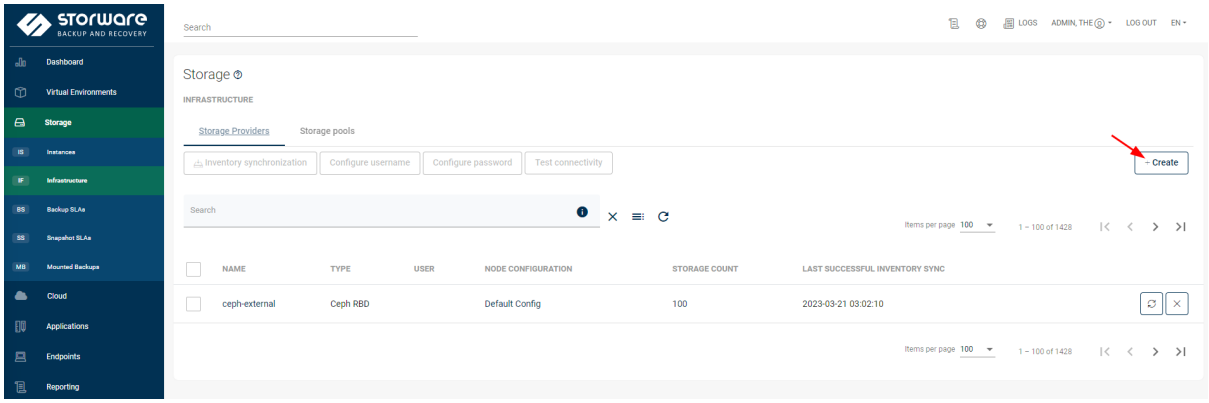
- **URL** - URL of the Keystone endpoint in version 3. (i.e., https://keystoneHost:5000/v3)
- **Region** - Region value configured in the OpenStack
- **Import/export mode** - leave it as a **Distk attachment** to proxy VM
- **Endpoint interface type** - choose which endpoint type should the Storware Backup & Recovery communicate with; depending on the environment, it can be Public, Internal, or Admin
- **Download image from Glance** - if turned on, during full Backup, also glance image from which the instance was created is backed up
- **Use domain-scoped authorization** - if turned on, authorization credentials are only used for the domain with which they were provided (you can provide multiple credential sets using Add authentication Domain button)

Authentication domain section:

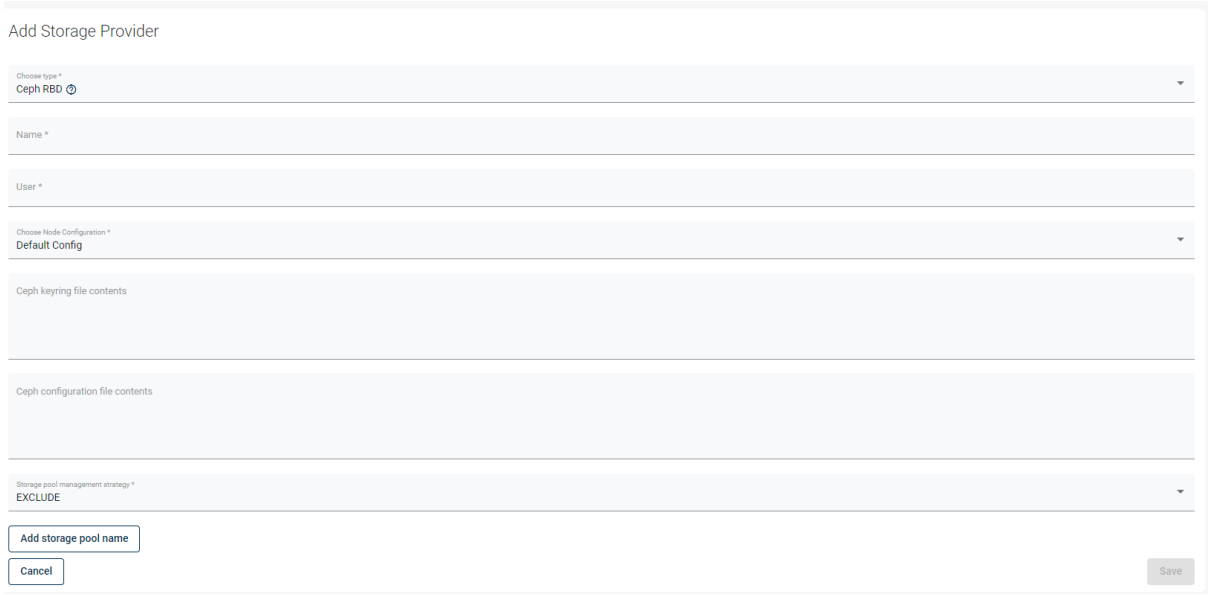
- **Name** - OpenStack domain name
- **User** - username used for authorization
- **Default** project name - default project used for authorization scoping
- **Domain ID** - OpenStack ID of the provided domain
- **Password** - password for the chosen user

After providing all the values and clicking save, you will be asked whether you want to do the inventory synchronization. Click yes, and do not worry if the task fails. This is because there is no Ceph storage added yet.

Next, go to storage, click the infrastructure section, and click Create.



Choose type Ceph RBD and fill in the required fields.



- **Name** - anything easily recognizable; it is only used in the Storware Backup & Recovery UI
- **User** - Ceph username used for authorization
- **Choose Node Configuration** - choose nodes from which node configuration should be used to work with Ceph

WARNING: Nodes used for Ceph communication must have the following rpm packages installed: `ceph-common` and `rbd-nbd`. They are available from the Ceph repository. **Ceph keyring file contents** - provide the content of the chosen Ceph user keyring file. Please leave the empty line at the end.

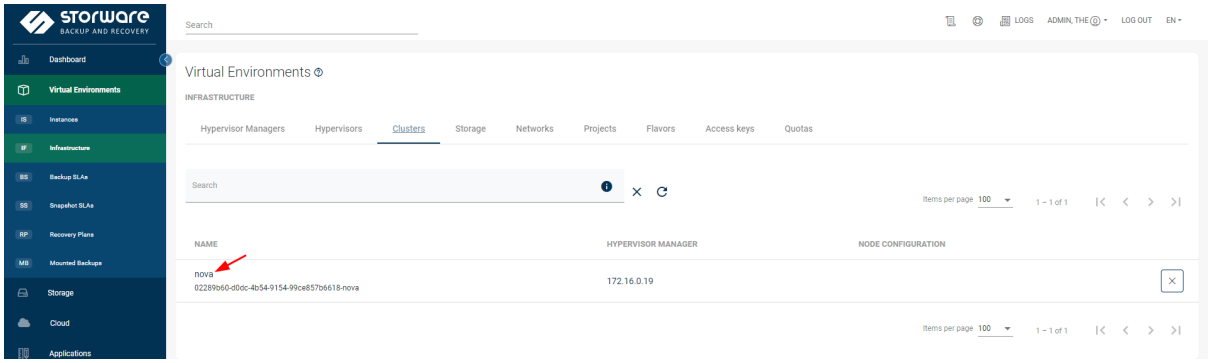
Ceph configuration file contents - provide the content of the Ceph config file. Please leave the empty line at the end.

Storage pool management strategy - if you choose to Exclude, you can add storage pools that should be excluded from the Ceph inventory synchronization. If you choose to Include, you can provide storage pools that should be scanned during the Ceph inventory synchronization (not added pools will not be scanned).

Add storage pool name - allows you to add pools that should be excluded or included in the Ceph inventory synchronization, depending on the option chosen above.

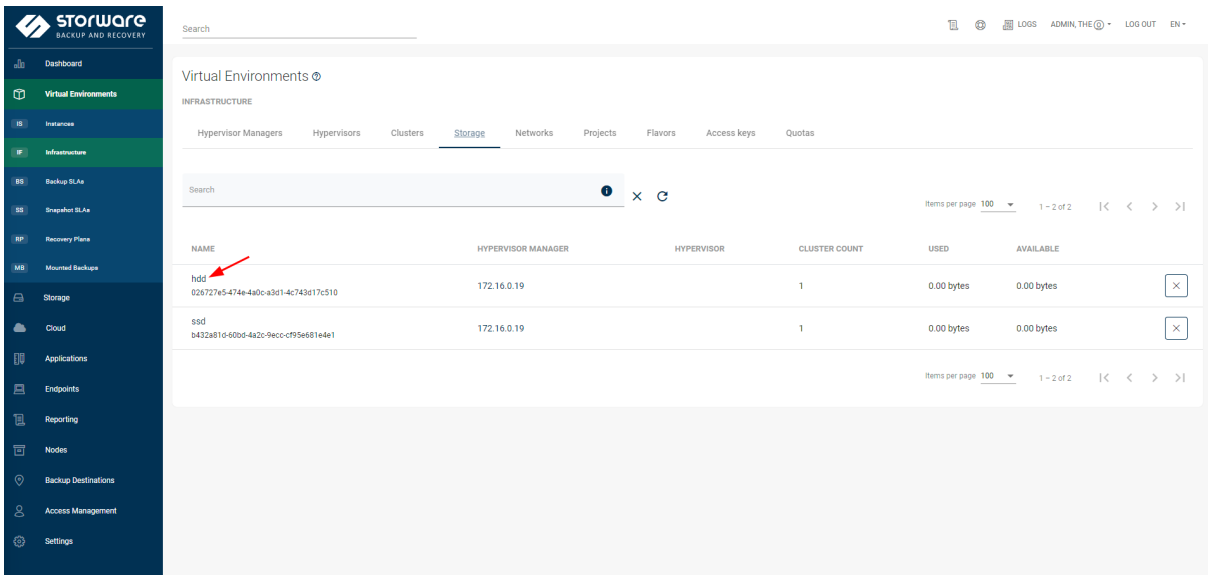
After providing all required values, you can save and proceed with the Ceph inventory synchronization. Once the task is done, the last step is configuring the relation between OpenStack and Ceph.

Go back to the Virtual Environments Infrastructure section and open the Clusters tab. Click on the name of the OpenStack Availability Zone



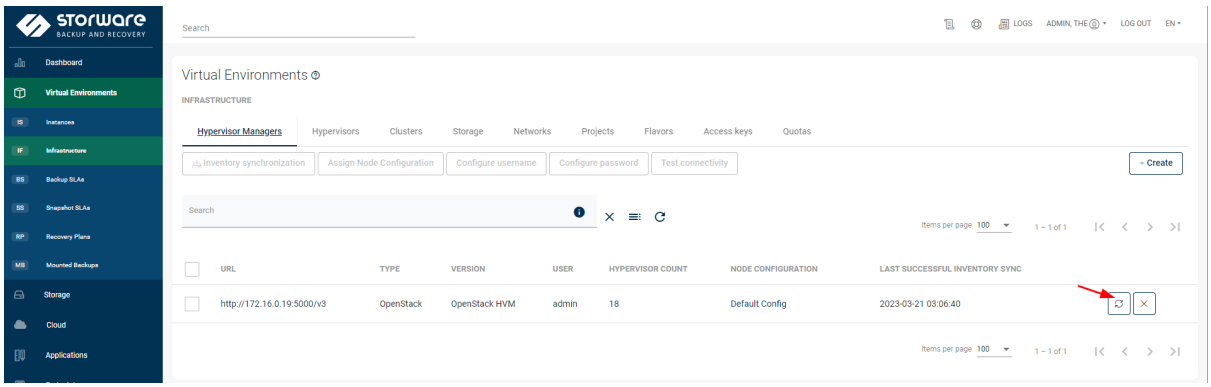
On the opened page, click the Storage Provider field and select the name of the previously added Ceph storage; next, click save. Repeat the procedure for all OpenStack Availability Zones.

Once all Availability Zones are configured, the next step is to enter the Storage tab and select volume types that use Ceph backend on the OpenStack side.



In the Storage pool name field, provide the name of the Ceph pool used by the selected volume type and click save. Repeat the process for all volume types that use Ceph.

The last step is to go to the hypervisor managers tab and run the inventory synchronization task. Once it is done, all is set.



SSH transfer backup strategy

Nova volumes backup is one aspect that can't be addressed with a disk-attachment backup strategy. While it is not recommended to keep application data on the same volume as the OS, it is certainly possible, and some users may configure their applications incorrectly. Nova volumes, provisioned for OS, are not visible as a cinder volume in the OpenStack, so they can't be mounted using cinder API. These are usually kept as QCOW2 files (a chain) on the host or dedicated storage underneath and require direct access to fetch data.

SSH transfer backup strategy works on a different layer than Disk-attachment and directly talks to the hypervisors - in other words - libvirt. This allows us to find appropriate QCOW2 files and transfer data directly from the host.

Note that the snapshot is done directly on QCOW2 files, not via API. Incremental backups are possible, and again the last snapshot needs to be kept in the environment to fetch delta changes since the previous Backup.

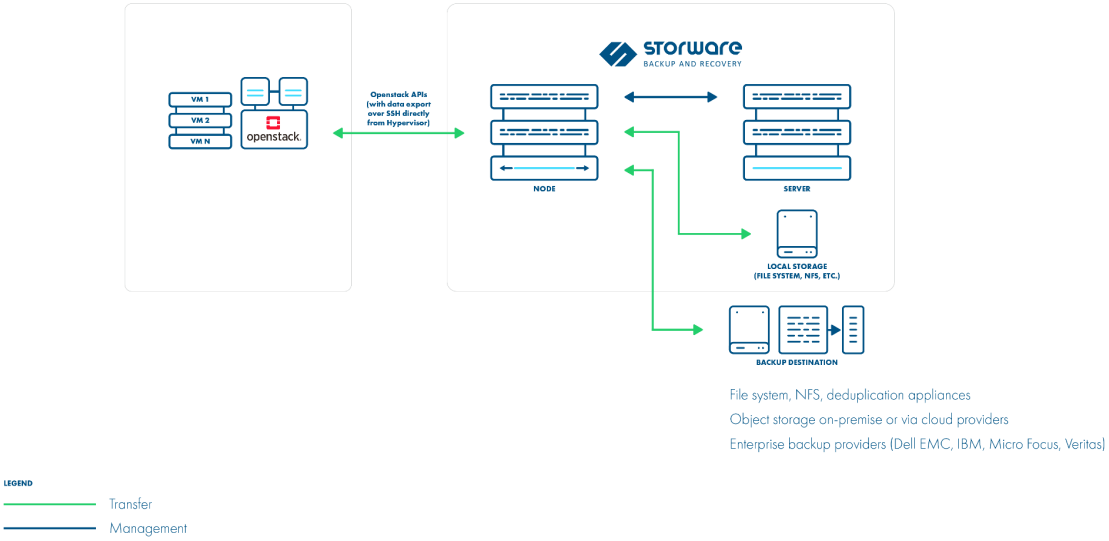
A side effect of this strategy is that you don't need any backup proxy to be installed as all of the data is transferred over LAN, so only network connectivity needs to be provided between hosts and the data mover.

When security requirements can be lowered, you can also optimize transfer by using Netcat instead of SSH, which can significantly improve transfer rates.

SSH backup strategy doesn't mean that Ceph RBD can't be used - actually, one can transfer data directly from Ceph, which should perform far better than reading data from the qcow2 files over SSH. Incremental backups can be in such a scenario implemented by mounting RBD devices (with RBD or RBD-NBD) so that we would have the same behavior as with disk attachment, and only changed blocks could be read from such mounted devices. It is also worth mentioning that Ceph RBD also has a mechanism to export data differences directly (without the need to mount such volume on the data mover).

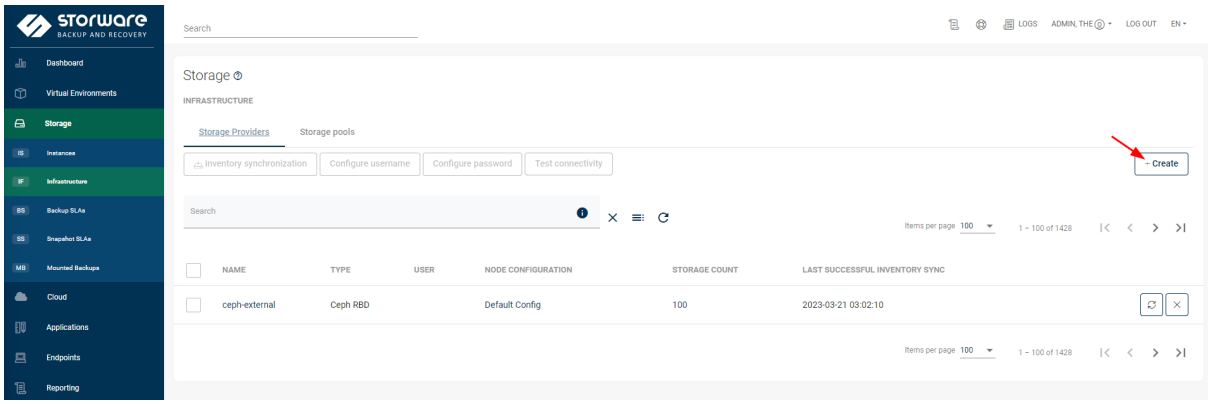
One thing to mention - the metadata always needs to be fetched from the OpenStack API, as during the recovery, we need to rebuild the OpenStack instance, not just a libvirt VM. You may also need to transfer Nova volumes using Glance to create an image before the instance can be booted.

This brings us to another issue - having backups of multiple VMs with the same image, which you may not want to restore with every instance, as this would occupy a lot of storage. Storware Backup & Recovery detects if the image you want to restore is already available in the environment and allows you to specify different images if you've lost your primary data center and the same image in the second DC has a different ID.

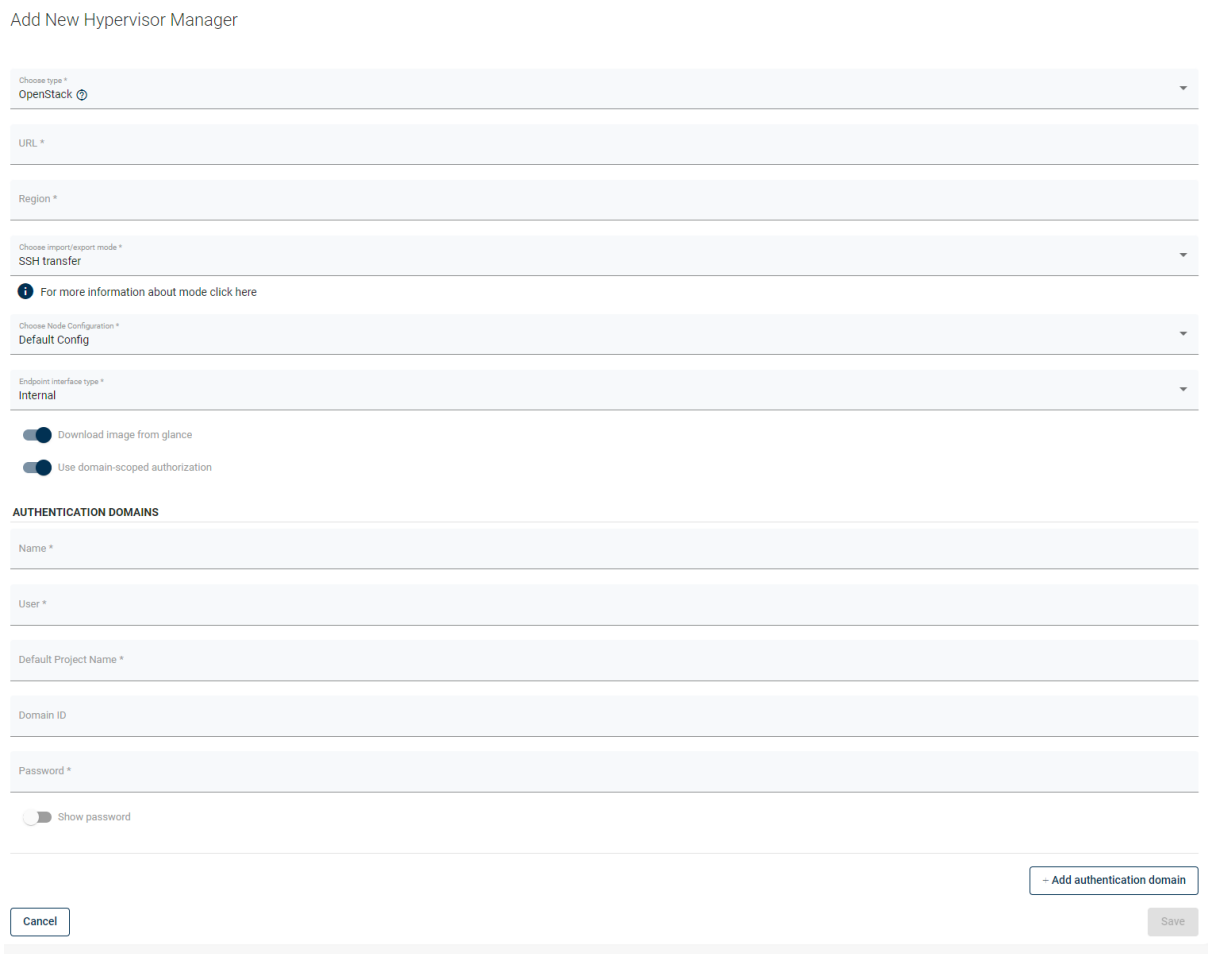


Example

To add the OpenStack, first, click Create in the Hypervisor Managers tab of the infrastructure section (virtual environments)



Choose the type OpenStack



Fields:

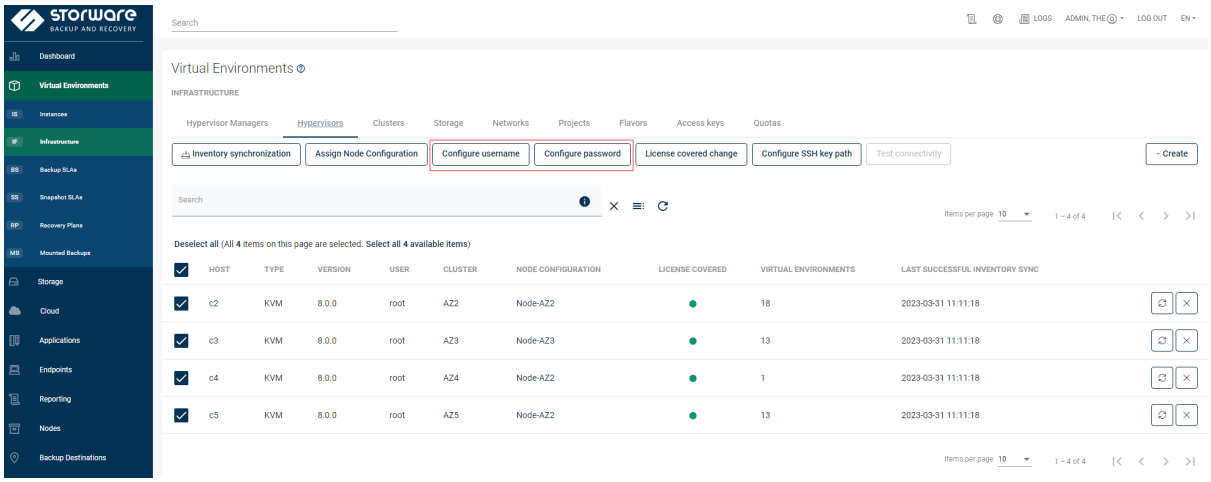
- **URL** - URL of the Keystone endpoint in version 3. (i.e., `https://keystoneHost:5000/v3`)
- **Region** - Region value configured in the OpenStack
- **Import/export mode** - set it to the **SSH Transfer**
- **Endpoint interface type** - choose which endpoint type should the Storware Backup & Recovery communicate with; depending on the environment, it can be Public, Internal, or Admin
- **Download image from Glance** - if turned on, during full Backup, also glance image from which the instance was created is backed up, if not the nova volumes will be backed up
- **Use domain-scoped authorization** - if turned on, authorization credentials are only used for the domain with which they were provided (you can provide multiple credential sets using Add authentication Domain button)

Authentication domain section:

- **Name** - OpenStack domain name
- **User** - username used for authorization
- **Default** project name - default project used for authorization scoping
- **Domain ID** - OpenStack ID of the provided domain
- **Password** - password for the chosen user

After providing all the values and clicking save, you will be asked whether you want to do the inventory synchronization. It is safe to proceed with this process. The inventory synchronization process will fail; this is ok for the first time because there is a need to provide the SSH credentials to access compute nodes.

To do this, you need to go to the Hypervisors tab in the Infrastructure section and select the hosts you want to configure by clicking the checkboxes next to them on the list and clicking Configure username and Configure password buttons.

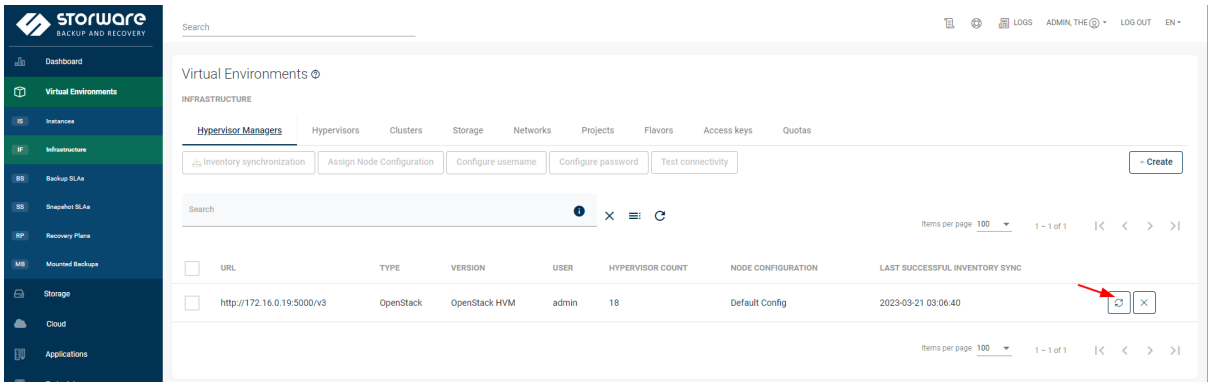


In case of using an SSH key for authorization, you should upload your private key to Storware Backup & Recovery nodes to the directory `/opt/project/.ssh` and change the owner and the permissions of this file.

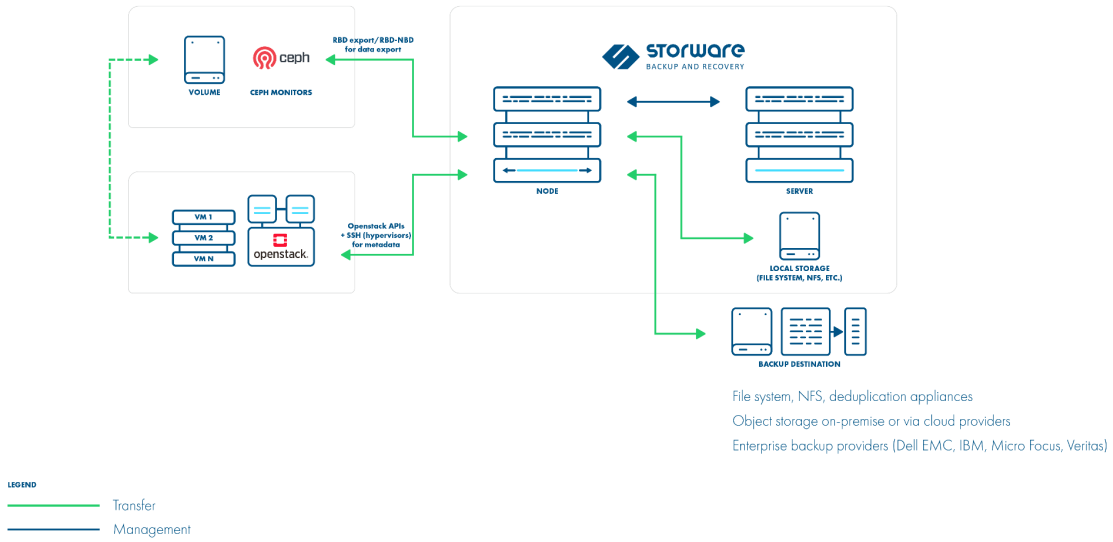
chown vprotect:vprotect /opt/vprotect/.ssh/YourKey
chmod 0600 /opt/vprotect/.ssh/YourKey

Next, go to the Hypervisors tab in the Infrastructure, select compute nodes, click `Configure SSH key path,` and provide the path to the previously uploaded key.

If the OpenStack installation was done inside containers, another step needs to be done to access the required containers. It is described in our knowledge base Openstack in containers
Once the host configuration is done, you can go back to the Hypervisor Managers tab and click the Inventory Synchronization icon next to the added OpenStack

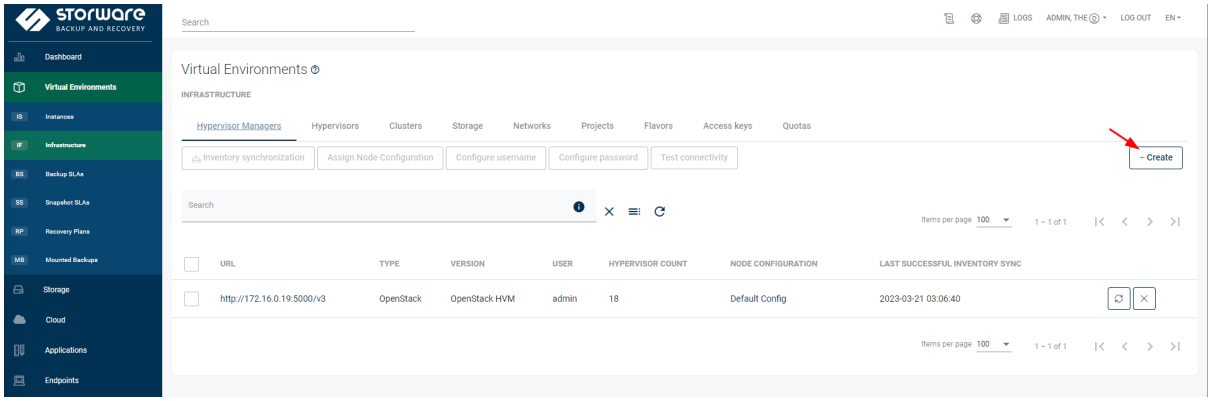


SSH transfer with Ceph RBD

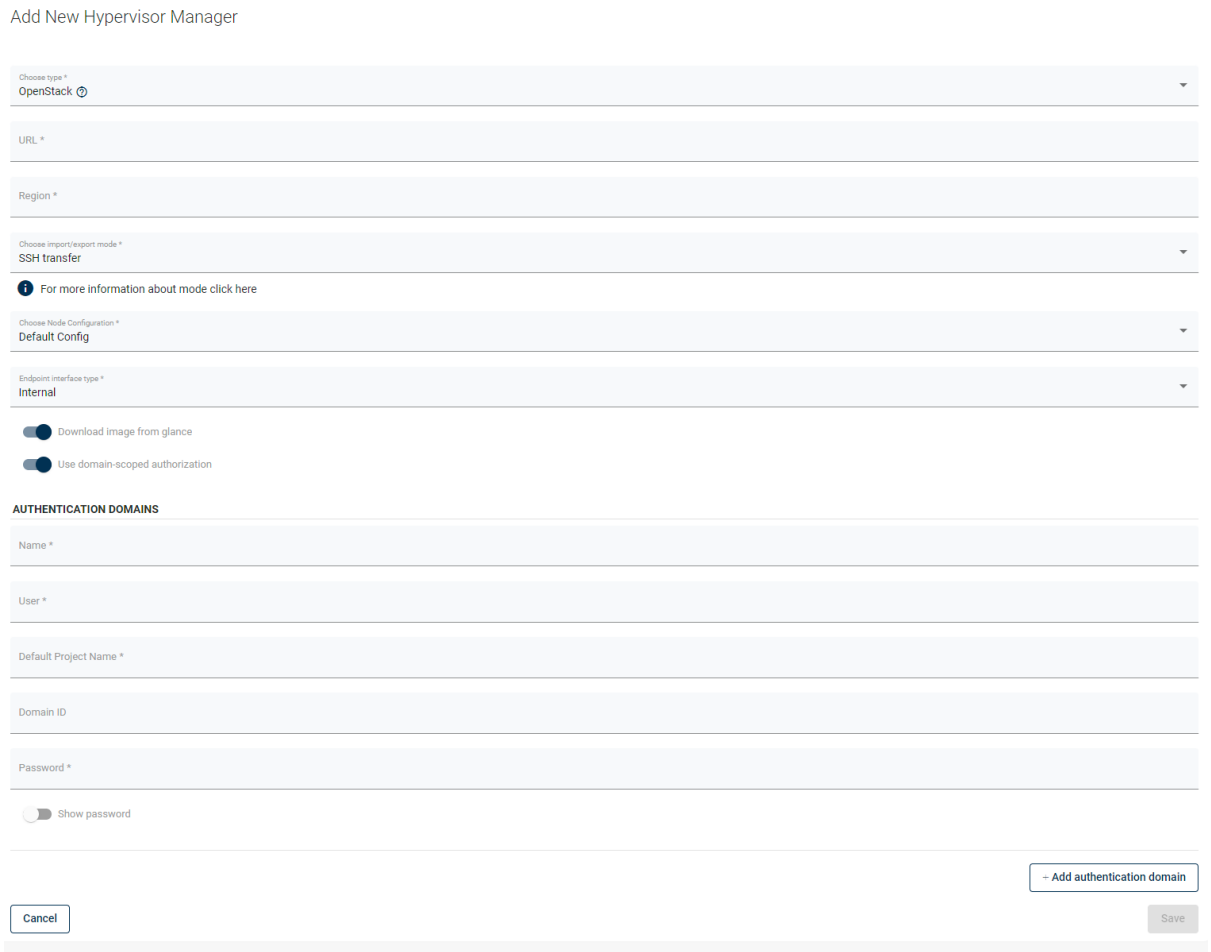


Example

To add the OpenStack, first, click Create in the Hypervisor Managers tab of the infrastructure section (virtual environments)



Choose the type OpenStack



Fields:

- **URL** - URL of the Keystone endpoint in version 3. (i.e., https://keystoneHost:5000/v3)
- **Region** - Region value configured in the OpenStack
- **Import/export mode** - set it to the **SSH Transfer**

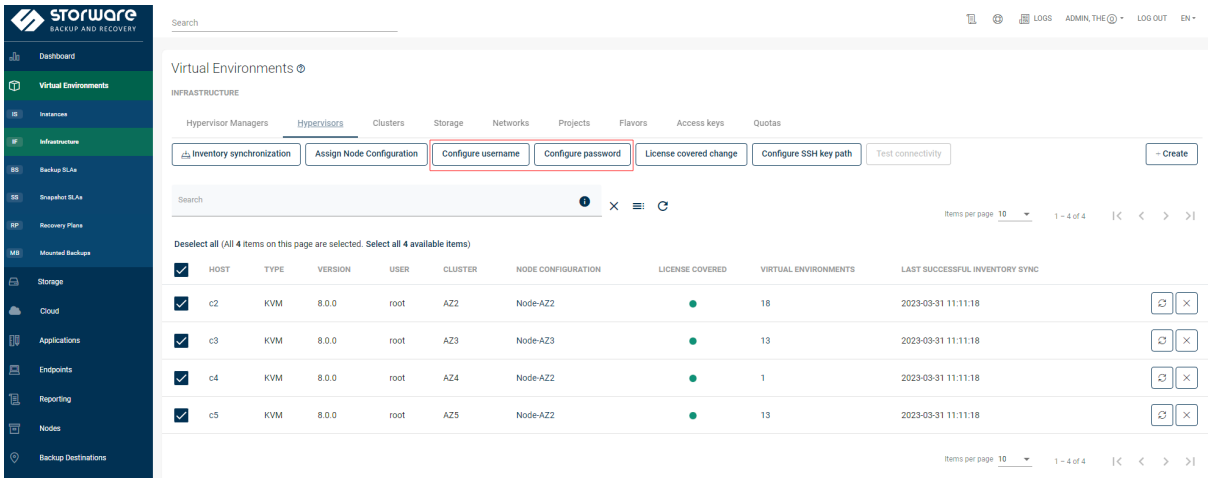
- **Endpoint interface type** - choose which endpoint type should the Storware Backup & Recovery communicate with; depending on the environment, it can be Public, Internal, or Admin
- **Download image from Glance** - if turned on, during full Backup, also glance image from which the instance was created is backed up, if not the nova volumes will be backed up
- **Use domain-scoped authorization** - if turned on, authorization credentials are only used for the domain with which they were provided (you can provide multiple credential sets using Add authentication Domain button)

Authentication domain section:

- **Name** - OpenStack domain name
- **User** - username used for authorization
- **Default** project name - default project used for authorization scoping
- **Domain ID** - OpenStack ID of the provided domain
- **Password** - password for the chosen user

After providing all the values and clicking save, you will be asked whether you want to do the inventory synchronization. It is safe to proceed with this process. The inventory synchronization process will fail; this is ok for the first time because there is a need to provide the SSH credentials to access compute nodes.

To do this, you need to go to the Hypervisors tab in the Infrastructure section and select the hosts which you want to configure by clicking the checkboxes next to them on the list and clicking Configure username and Configure password buttons.



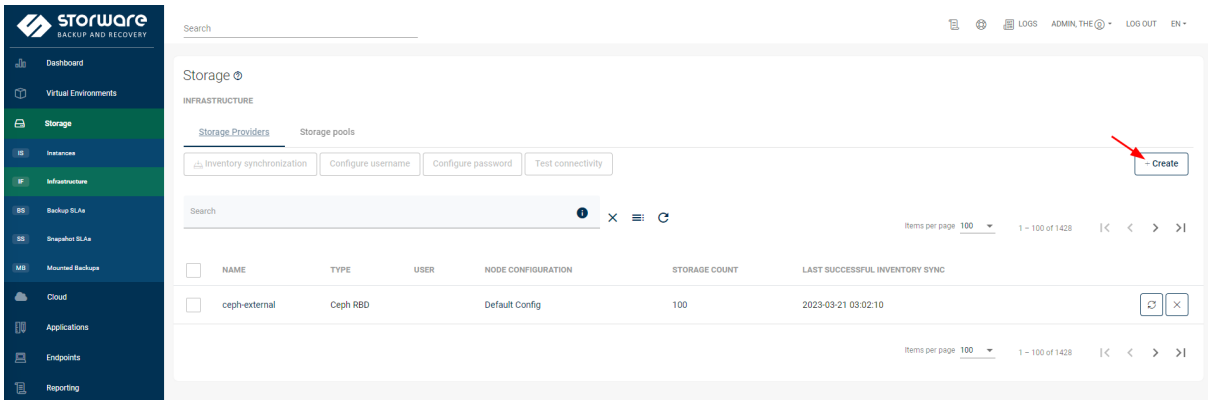
In case of using an SSH key for authorization, you should upload your private key to Storware Backup & Recovery nodes to the directory `/opt/project/.ssh` and change the owner and the permissions of this file.

chown vprotect:vprotect /opt/vprotect/.ssh/YourKey
chmod 0600 /opt/vprotect/.ssh/YourKey

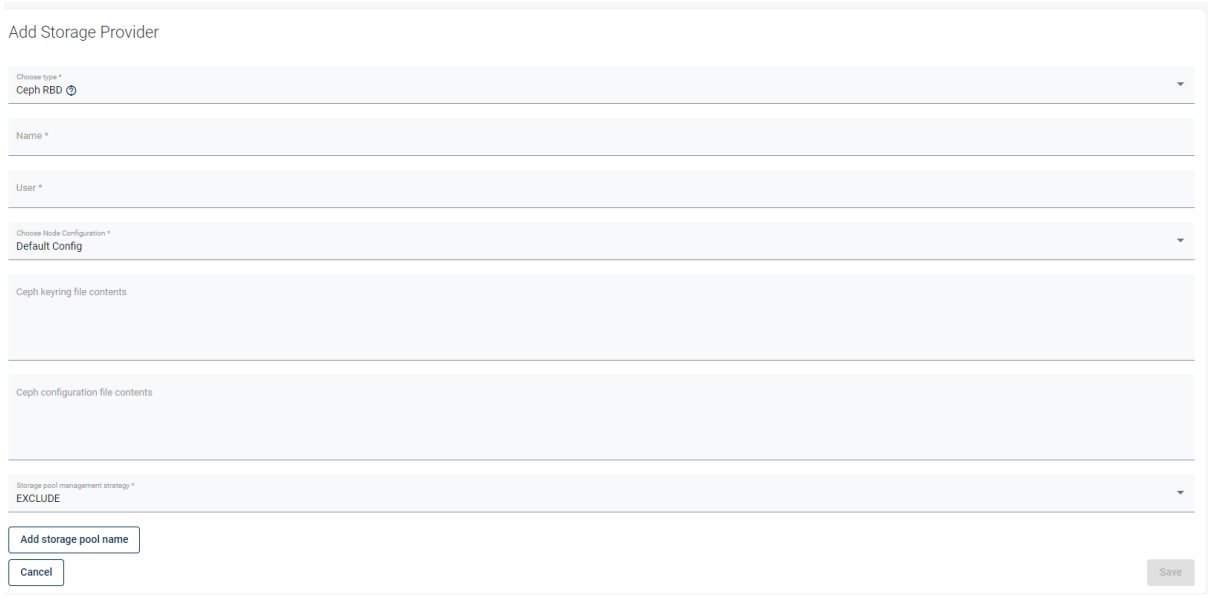
Next, go to the Hypervisors tab in the Infrastructure, select compute nodes, click `Configure SSH key path,` and provide the path to the previously uploaded key.

If the OpenStack installation was done inside containers, another step needs to be done to access the required containers. It is described in our knowledge base [Openstack in containers]

Next is a need to configure Ceph. To do this, go to storage, click the infrastructure section, and click Create.



Choose type Ceph RBD and fill in the required fields.



- **Name** - anything easily recognizable; it is only used in the Storware Backup & Recovery UI
- **User** - Ceph username used for authorization
- **Choose Node Configuration** - choose nodes from which node configuration should be used to work with Ceph

WARNING: Nodes used for Ceph communication must have the following rpm packages installed: `ceph-common` and `rbd-nbd.` They are available from the Ceph repository.

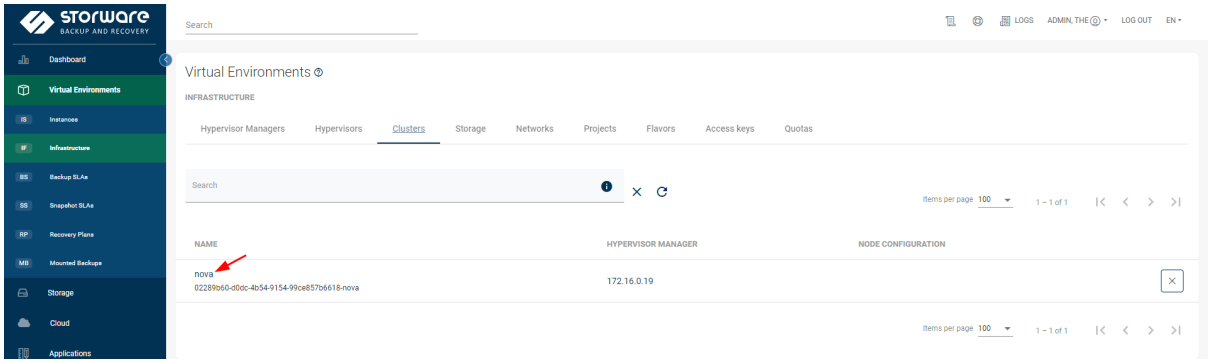
Ceph keyring file contents - provide the content of the chosen Ceph user keyring file. Please leave the empty line at the end.

Ceph configuration file contents - provide the content of the Ceph config file. Please leave the empty line at the end.

Storage pool management strategy - if you choose to Exclude, you can add storage pools that should be excluded from the Ceph inventory synchronization. If you choose to Include, you can provide storage pools which should be scanned during the Ceph inventory synchronization (not added pools will not be scanned)

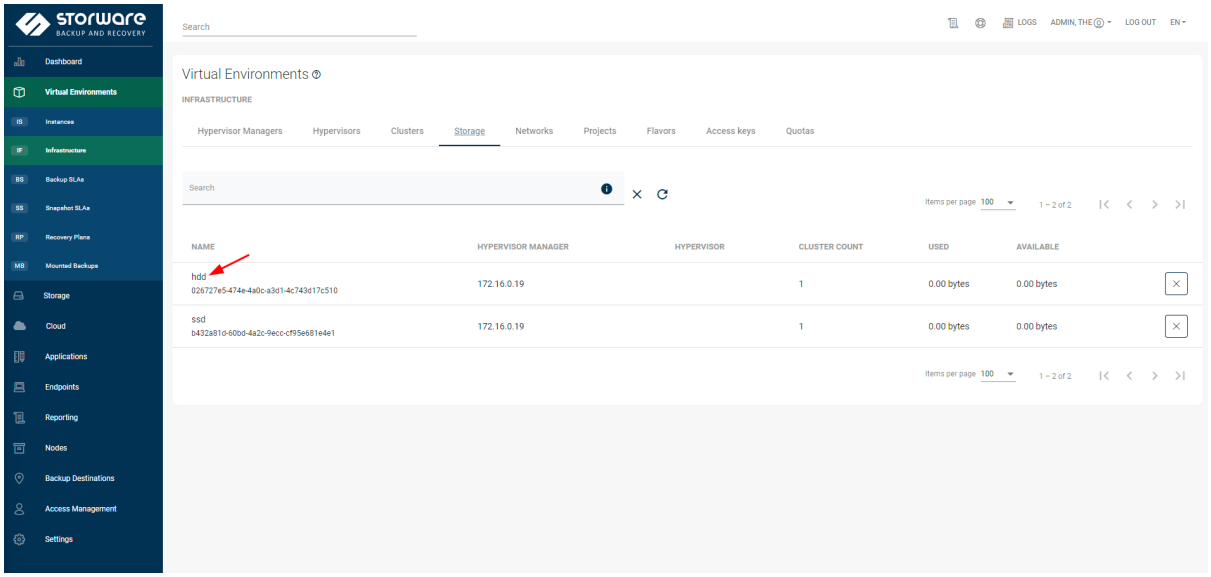
Add storage pool name - allows you to add pools that should be excluded or included in the Ceph inventory synchronization, depending on the option chosen above.

After providing all the required values, you can save and proceed with the Ceph inventory synchronization. Once the task is done, the last step is configuring the relation between the OpenStack and the Ceph. Go back to the Virtual Environments Infrastructure section and open the Clusters tab. Click on the name of the OpenStack Availability Zone



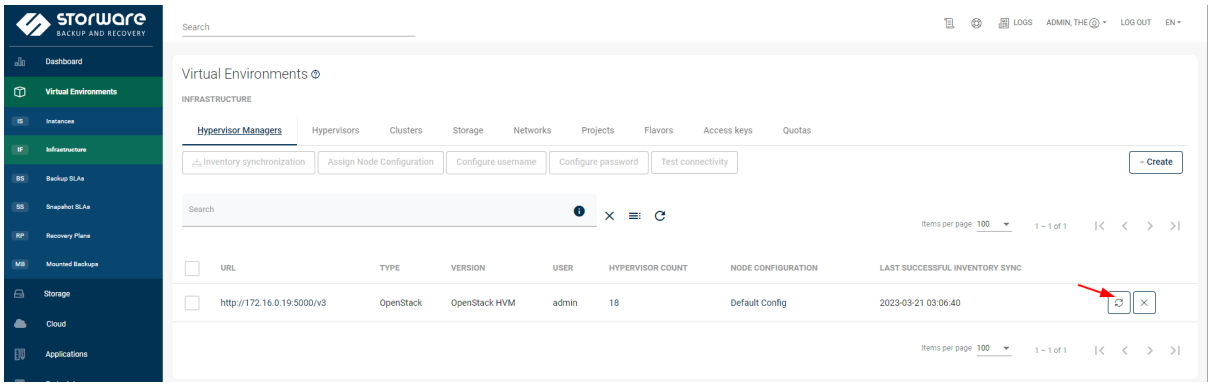
On the opened page, click the Storage Provider field and select the name of the previously added Ceph storage; next, click save. Repeat the procedure for all OpenStack Availability Zones.

Once all Availability Zones are configured, the next step is to enter the Storage tab and select volume types that use Ceph backend on the OpenStack side.



In the Storage pool name field, provide the name of the Ceph pool used by the selected volume type and click save. Repeat the process for all volume types that use Ceph.

Once the host and the storage configuration are done, you can go back to the Hypervisor Managers tab and click the Inventory Synchronization icon next to the added OpenStack



Backup storage

Once the instance data and metadata have been exported, they must be stored safely. One of the commonly used storage types is a file system. It may offer deduplication, snapshots, or support reflinks to build a synthetic backup destination, but in general, it is always worth considering it as a starting point.

However, when we talk about a larger scale, it may be necessary to use scalable storage as the backend, which is usually object storage.

OpenStack even has a dedicated service to build such object storage called Swift. Swift also has support for S3 API, which means that during the implementation phase, you can choose which APIs will be used for the integration.

One aspect worth mentioning is the object size limit; as for Swift, by default, it is 5GB, so even for regular-size disks - object segmentation needs to be used.

Another non-obvious advantage of having scalable storage is that you may use one backup destination, which all data movers can share. This opens a scenario to backup instances from one AZ or Datacenter and restores in the other without the limitations of the file system.

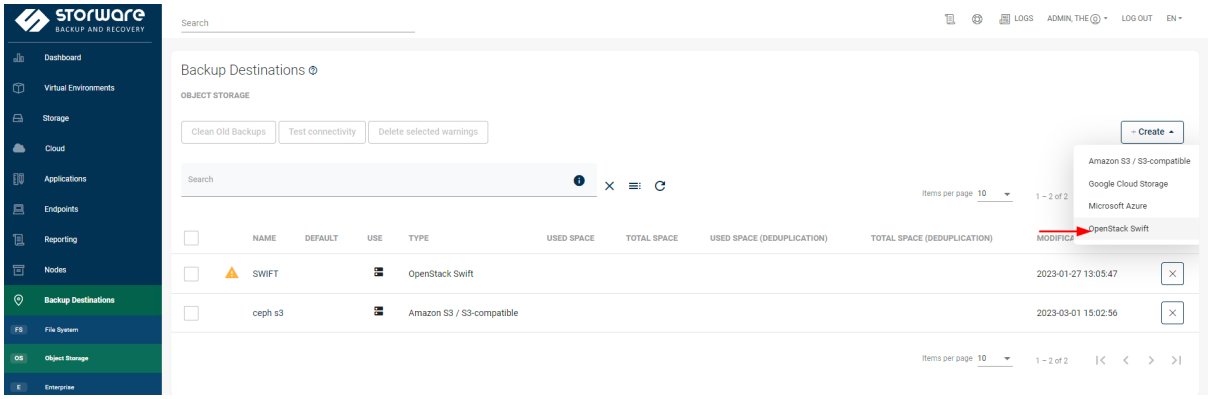
The drawback of object storage is that you may need to rebuild the whole backup chain during a restore, as synthetic backups will be impossible without building a sophisticated data chunking mechanism.

You may also need to consider implementing object compression, as object storage doesn't support sparse disks. It is common to have empty space inside your instances, unnecessarily consuming a lot of space.

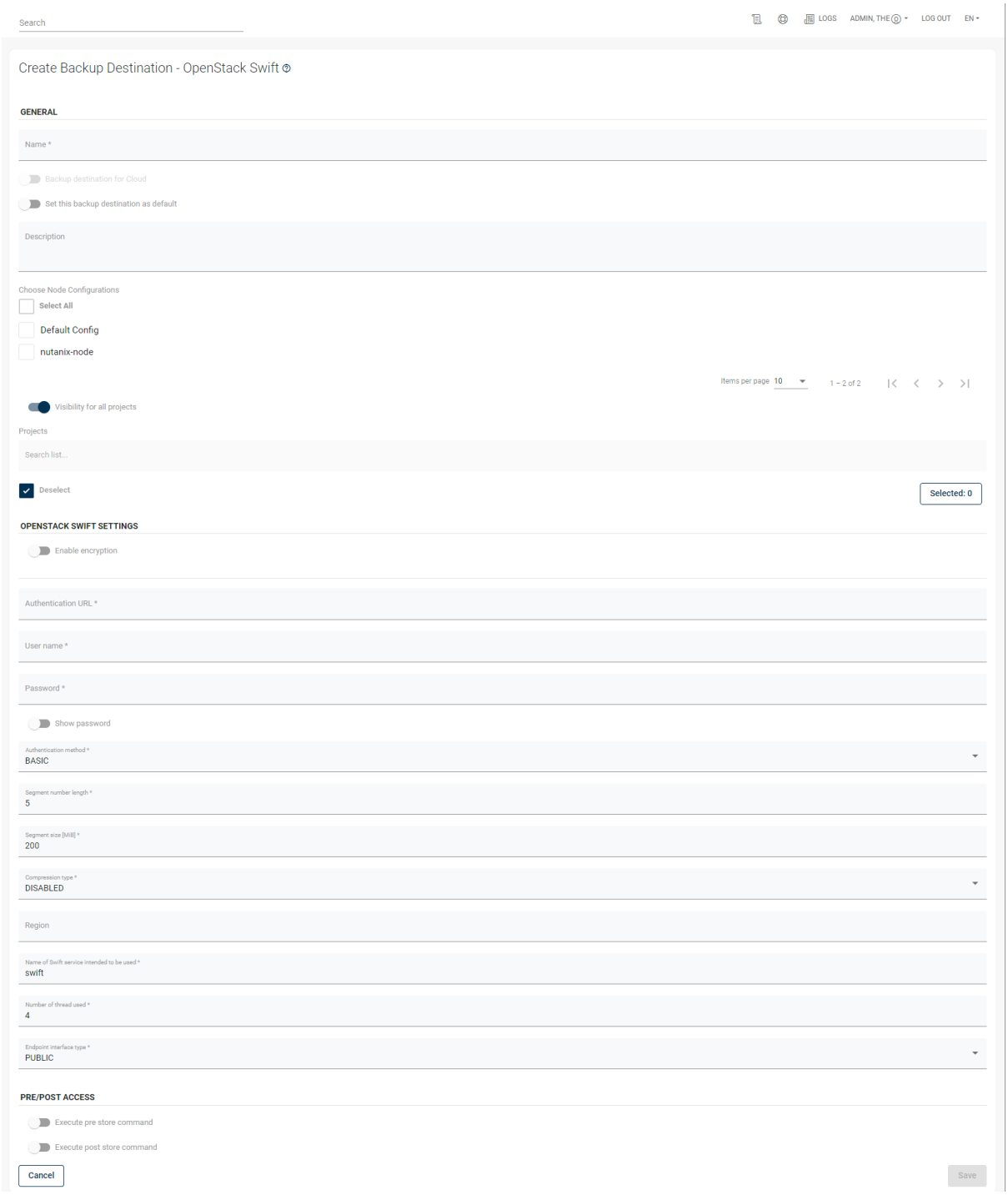
Storware Backup and Recovery support object compression and data segmentation for Swift, so the instance disk size is not an issue.

Example

To use OpenStack Swift as a backup destination, go to the Backup Destinations and Object Storage section. Click Create and choose OpenStack Swift.



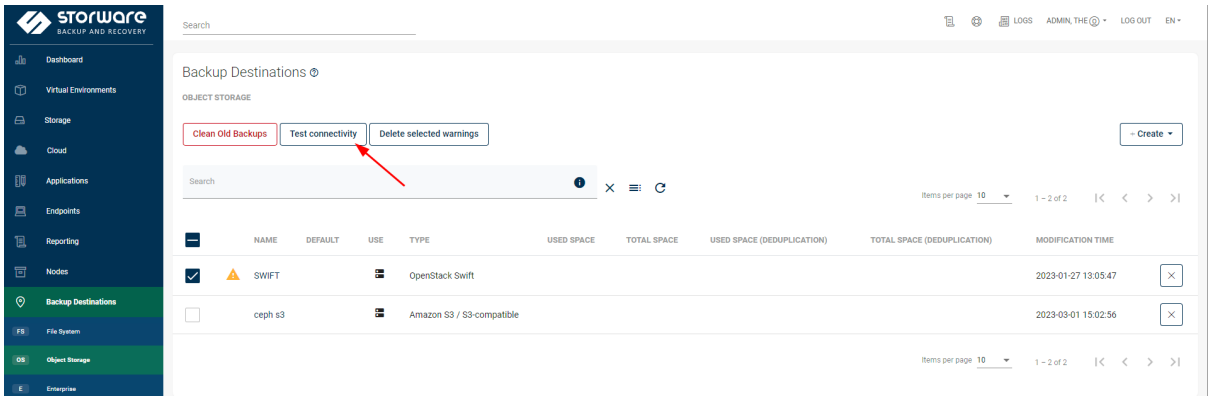
Next, fill in all the required fields



- **Name** - anything easily recognizable; it is only used in the Storware Backup & Recovery UI
- **Choose Node Configurations** - select node configurations that should communicate with Swift
- **Authentication URL** - provide the URL to the OpenStack Keystone responsible for the Swift authorization
- **User name** - provide the user with access to the Swift
- **Password** - provide the password for authorization
- **Authentication method** - choose the method which is used in your Keystone to authenticate; in most cases, this will be `KEYSTONE_V3`

- **Segment number length** - the maximum length of the object segment name, i.e., 5, will allow segment numbers to be up to 99999
- **Segment size** - the size of the data segment into which data will be fragmented. Segment size, together with its number length, limits the maximum backup size (it is segment size multiplied by maximum segment number+1)
- **Compression type** - choose if you want to use gzip compression for the data sent to Swift or not
- **Region** - OpenStack region name in which Swift is running
- **Name of Swift service intended to be used** - Name of the Swift endpoint in the OpenStack, which should be used in the communication.
- **Number of threads used** - number of threads used for simultaneous Swift operations
- **Endpoint interface type** - choose which endpoint type should the Storware Backup & Recovery communicate with; depending on the environment, it can be Public, Internal, or Admin

After clicking save, it is recommended to do the Connectivity test to ensure all chosen Stoware Backup & Recovery nodes have access to Swift.



Scalability

When building a large-scale environment, one aspect to consider is the backup solution's scalability. In Storware Backup & Recovery, data movers, called Nodes, are responsible for actual data movement, and you can deploy multiple nodes to cover larger environments and simultaneously execute backup jobs.

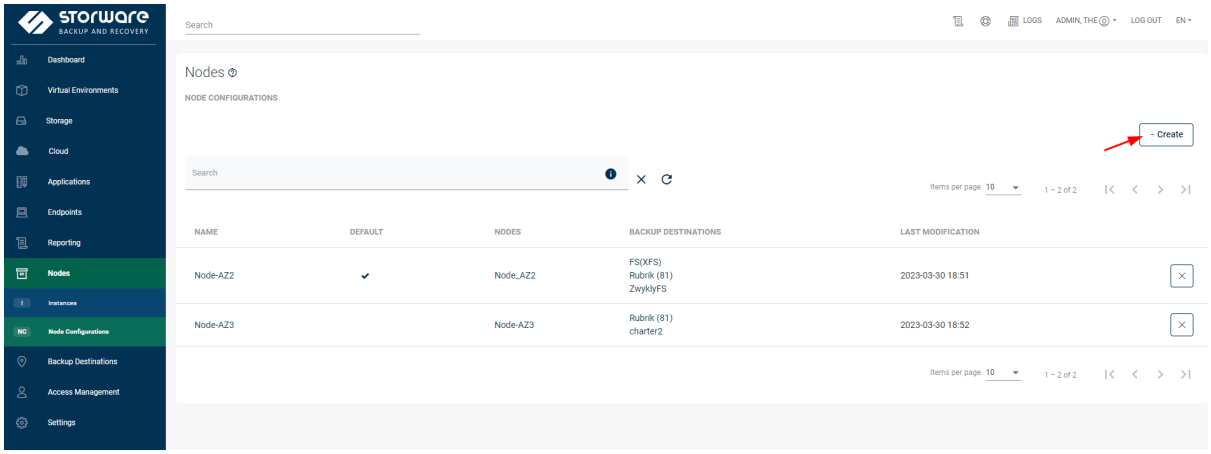
For easier management, nodes can share their settings by assigning the same node configuration. From a scalability perspective, it is worth noting that it allows specifying the number of backup threads executed by each node simultaneously, so that the administrator can very quickly increase the throughput of the backup solution, assuming that the hardware is able to handle the increased workload.

Another setting is the list of available backup destinations for nodes with the specific configuration. This is important not only because it logically limits which node is allowed to backup instances to the specific backup destination (which may be especially important from a networking perspective) but also allows it to limit end users' usage of the same backup destination from a performance perspective.

It is also worth noting that when using the Disk-Attachment method, you need to remember the SCSI device limit per bus, which may limit the number of simultaneous export tasks to around 25 per node. If you have a large availability zone, this could be a problem. However, Storware Backup & Recovery allows you to assign node configuration even to individual hosts so that in a 10-host availability zone, you could have up to 10 nodes, each backing up VMs from that host.

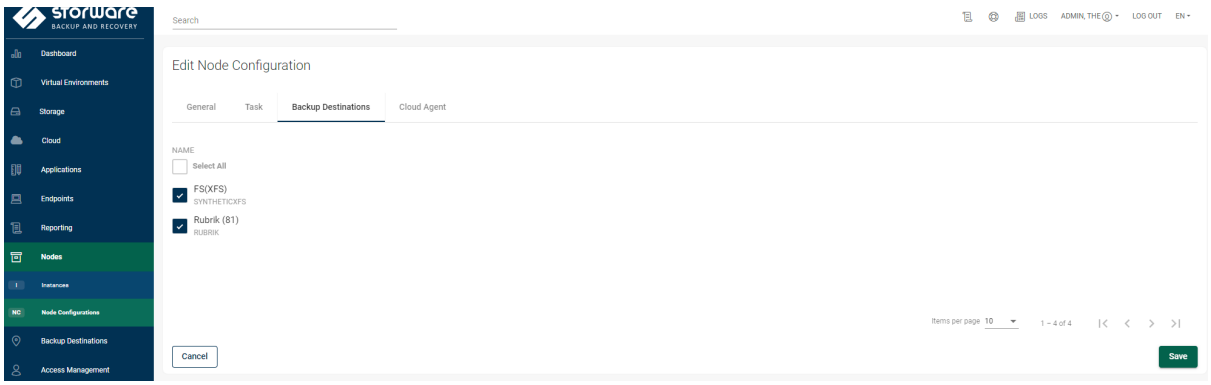
Example

To create more node configurations, you must select node configuration from the Nodes sections. Next, click Create in the top right corner.



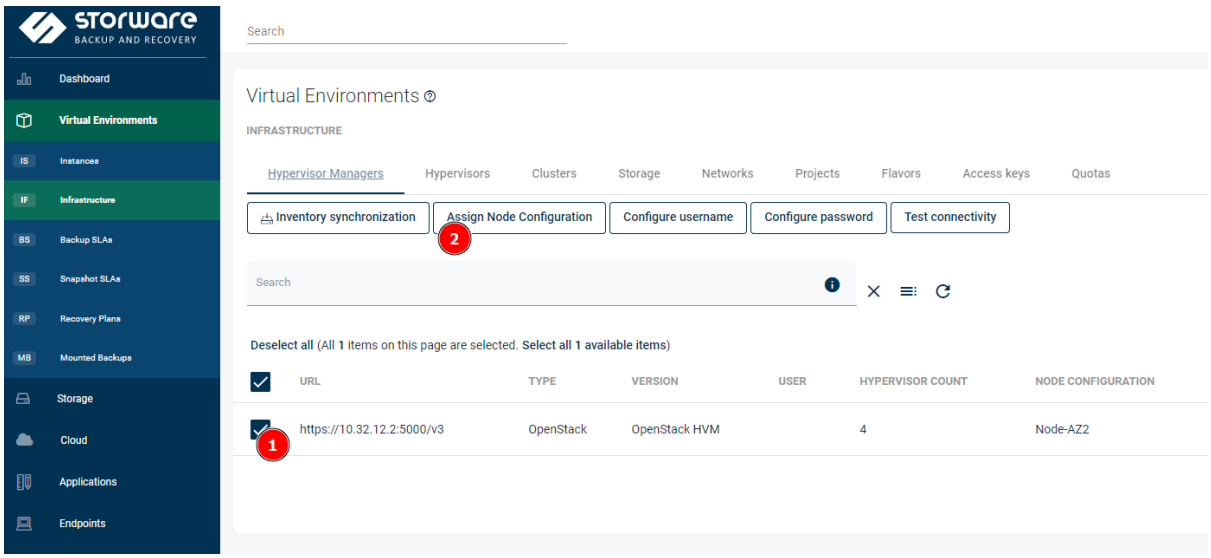
All configurable parameters are described in our documentation Node configuration. From the OpenStack perspective, most of the parameters can be left as default; things to consider are:

- **Export path (staging space)** - path for the staging space must be available in every node assigned to the created node configuration
- **Backup destinations** - the chosen ones must be available for all nodes inside the node configuration



After creating the node configuration, you can assign it to OpenStack or choose only specific compute nodes. This can be done in the Virtual Environments in the Infrastructure section. If you want to set the node config for the whole OpenStack, select the checkbox next to the added OpenStack, click the Assign Node Configuration button, and choose the previously created Node Configuration.

If you want to assign Node Configuration to specific hosts, the procedure is almost the same; the only difference is that you need to go to the Hypervisors tab in the Infrastructure sections, choose desired hosts, and assign Node Configuration to them.



Horizon integration

When we talk about the cloud in general, multi-tenancy is one of the key aspects that also needs to be planned. Each tenant can use a separate project and has a limited view of the project's scope. OpenStack has a service called Horizon, which serves exactly this purpose.

With Storware Backup & Recovery, you can use a Horizon Plugin that enables the Backup & Recovery tab in the Horizon, which also is aware of the project in which the user currently works. This plugin allows almost all key actions to be executed by the end users using an intermediary service that forwards appropriate API calls to the Storware Backup & Recovery server. The idea is that the user only can work on the instances that are (or were previously, before, i.e., accidental removal) within his project.

Using this plugin users can create policies and schedules and point to the appropriate backup destinations. The views and overall UI structure almost directly correspond to the main Storware Backup & Recovery UI but directly from the Horizon.

One aspect which is essential for MSP is the option to limit the scope for which projects specific artifact is visible. A good example is the backup policy, backup destination, or availability zone to which end-users could request the instance to be restored. In the details of such items, there is a section called Projects and a switch that indicates that it is visible for all projects. Administrators can change it and allow it only to be used by specific projects.

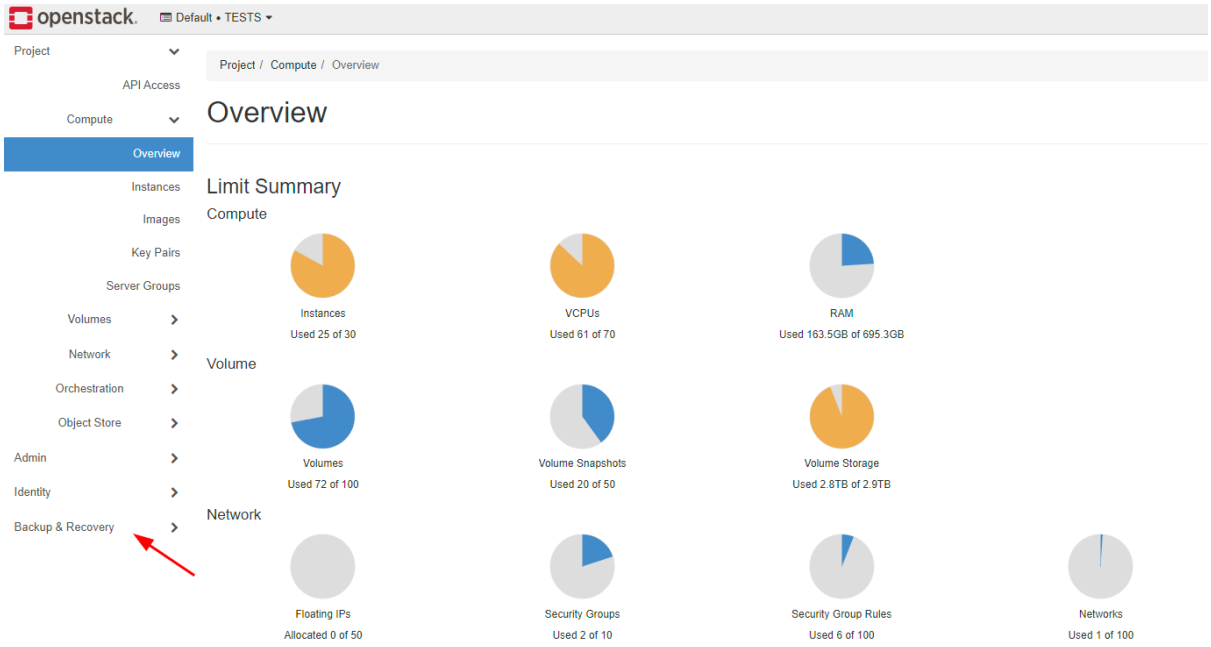
Example

If OpenStack is not in the containerized environment, installing the plugin is straightforward. First, you need to clone the GitHub repository and execute one command to install. The installation has to be performed on the OpenStack node responsible for the Horizon.

```
git clone https://github.com/Storware/openstack-horizon-ui-vprotect-extensions
cd openstack-horizon-ui-vprotect-extensions
python install.py https://Storware.url:8181/api StorwareLogin StorwarePassword
```

In case of using containers in the OpenStack environment, our documentation has more precise instructions, as well as it contains the description of the plugin itself: [Horizon plugin documentation]

Once the plugin is installed after logging into the Horizon, you should see a new section called Backup & Recovery



Backup quotas

Giving end-users the option to specify their backup policies or constantly executing multiple backups can impose a significant load on the infrastructure. Storware Backup & Recovery allows us to define Quotas and assign them to individual projects to limit such activity.

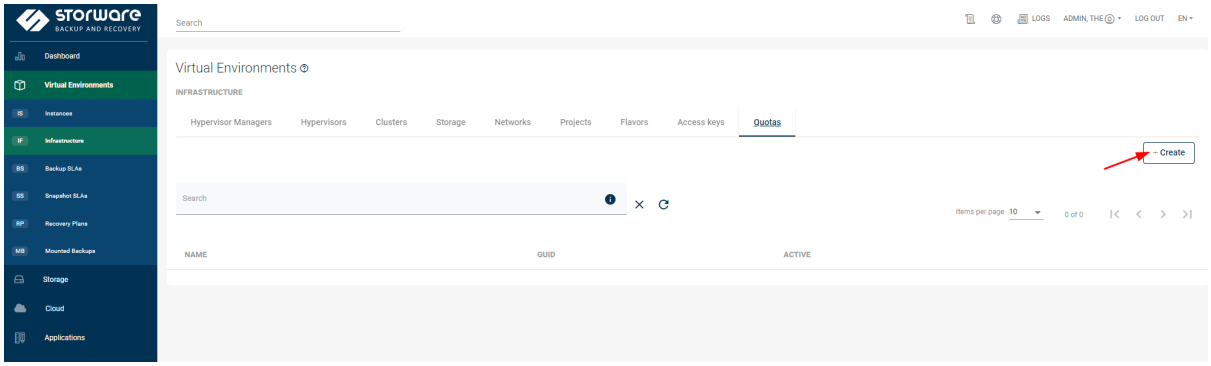
Quotas allow defining limits - soft and hard for Backup and restore activities. A soft limit just lets administrators know that the limit has been reached; a hard prevents Backup or restores from starting.

Administrators can limit the number of backup/restore operations per instance or project within the given time window. It means that if your time window is 1 hour, Storware Backup & Recovery counts all backups that have taken place for this instance or project (depending on the setting) within 1 hour until now.

Similarly, administrators may be rather interested in the volume of data than the number of backups. And again, parameter backup capacity is available for both per instance and project.

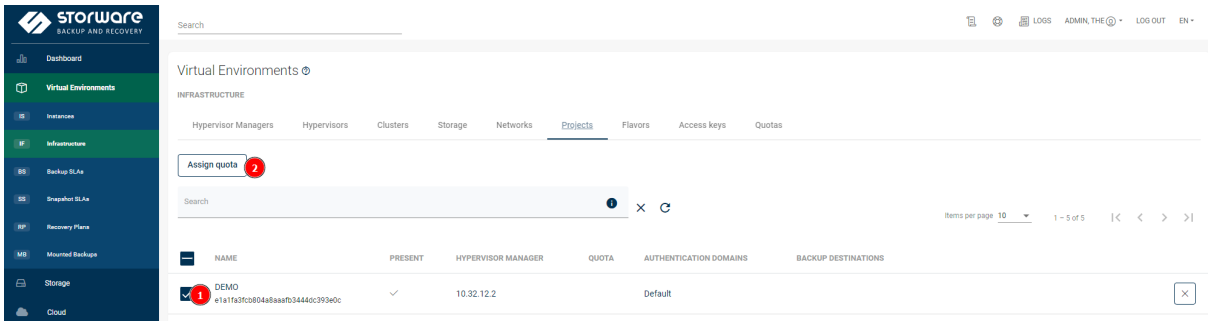
Example

In order to create a quota, you need to go to the Virtual Environments section and choose Infrastructure, next select the last tab, Quotas and click Create.



In the next screen, you must provide the name only used in the Storware Backup & Recovery UI to differentiate quotas and set the limits. For each option, you can set either soft or hard limits. Soft will only notify the administrator, whereas hard will stop the backups and restores from being processed. After setting parameters according to your requirements, click save.

The next step is to assign a quota to the OpenStack projects. To do this, go to the Projects tab in the Infrastructure section and click the check box next to the project name. Then click the Assign quota button and select the previously created quota's name.



Automation and billing

Backup as a Service (BaaS) solution may offer end customers a completely different portal for general billing. It may also cover OpenStack services so that end-users may be unaware that the infrastructure uses OpenStack. Similarly, backup/restore operations may be exposed as a dedicated button in such a portal with limited options.

Both OpenStack and Storware Backup & Recovery rely heavily on the APIs to achieve this. Technically everything you can do using Web UI you can invoke by running appropriate API calls.

In Storware Backup & Recovery, you must create a service account with the appropriate RBAC permissions. This account can later be used to invoke APIs on the server. Usually, the flow is as follows:

- We need to login to API to get a cookie
- We need to create a task of the desired type - such as a backup for our instance
- then we can monitor the progress of a single task, its workflow, or backup entry in the backup history to report progress/status to the end user

In general, Backup can also be a part of a wider process, such as a maintenance window, which automatically can invoke a backup operation. Such workflows can fully be automated using APIs or, for simpler cases using scripts and CLI.

There are also statistics available in the backup history APIs, such as backup time, transfer rate, or backup size, that can be fetched, which are valuable for end-users. For MSPs, however, it is billing that is far more important.

Storware Backup & Recovery offers a dedicated backup size reporting tab, where MSPs can periodically generate a report that groups total backup storage space consumption (or, in other words: total data transferred, as you also can specify a time range) aggregated by instance, cluster, policy, backup destination and more. Such reports can then be filtered so that we can ask for individual instances or want to skip backups residing in a backup destination that we assume is eligible for the free tier.

Reporting APIs also can collect the size of the front-end capacity estimation if required - based on the size of the last full Backup. This is an estimation, as verifying if the occupied block has data that is still used by each file system would make the backup process much more complex and require significantly more resources. It is worth noting that while empty spaces are counted as occupied, incremental backups are excluded, as the front-end capacity can't be larger than the instance itself.

Billing APIs are expected to be invoked periodically, so the billing history will eventually be collected on the Backup as a Service (BaaS) portal side. One reason is that every MSP may use different billing periods and rules, so APIs always return current values.

Example

Since Storware Backup & Recovery uses REST API, it can be easily called in many scripting languages. Below is the sample in Python, which logs in and gets the list of the backup destinations. What needs to be changed is the URL and credentials to the Storware Backup & Recovery.

```
import json
import requests

url = "https://StorwareURL:8181/api/"
s = requests.Session()
payload = { "login" : "admin," "password" : "vPr0tect" }
s.post(url+"session/login", json=payload, verify=False)
#getting the first backup destination from Stoware Backup & Recovery
r = s.get(url+"backup-destinations")
backupDestinations = r.json()
```

Very large scale

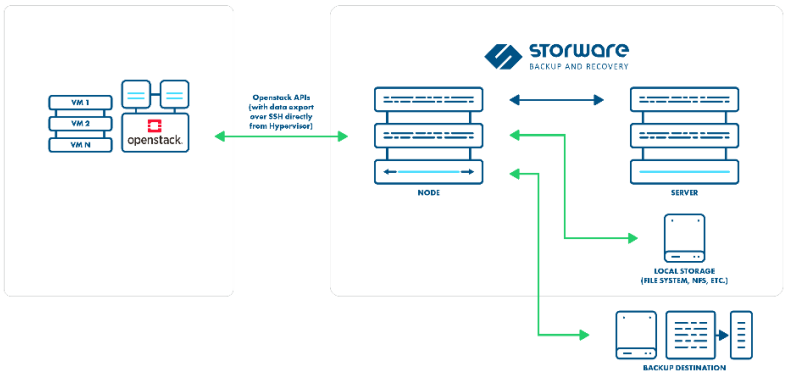
There may be a point where using a single server in Storware Backup & Recovery solution may result in poor performance. In such a scenario, we also recommend splitting the backup infrastructure into smaller parts. In the same way, as large OpenStack environments may have multiple availability zones or regions, you can have dedicated, separate backup servers to cover even the biggest cloud platforms.

You can prepare ahead 24 daily backup schedules for daily backups, but users may quite often choose midnight as the start time for their backups, which may result in failed backups as not all of them will be processed within the given backup window. Suppose your consideration also relates to the cumulation of workload at certain periods. In that case, some MSPs predefine policies with schedules for specific hours and charge customers differently based on the predicted load for the selected schedule or policy. With a simple API call for policy details, you can fetch how much data needs to be processed for each policy and show a different price tag in the Backup as a Service (BaaS) portal.

Example

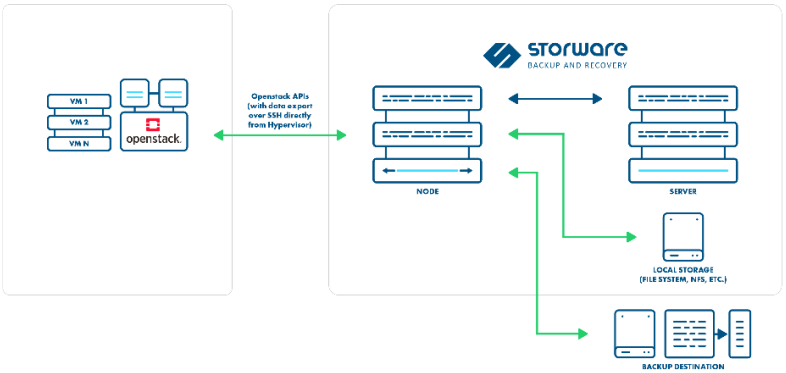
A sample approach to a very large-scale environment

Availability Zone 1



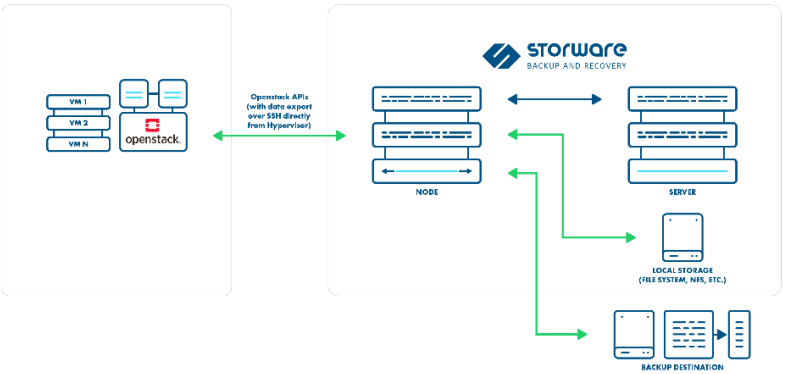
File system, NFS, deduplication appliances
Object storage on-premise or via cloud providers
Enterprise backup providers (Dell EMC, IBM, Micro Focus, Veritas)

Availability Zone 2



File system, NFS, deduplication appliances
Object storage on-premise or via cloud providers
Enterprise backup providers (Dell EMC, IBM, Micro Focus, Veritas)

Availability Zone 3



File system, NFS, deduplication appliances
Object storage on-premise or via cloud providers
Enterprise backup providers (Dell EMC, IBM, Micro Focus, Veritas)

LEGEND

Transfer

Management

The sample API call to fetch the average backup size from policies

```
import json
import requests

url = "https://StorwareURL:8181/api/"
s = requests.Session()
payload = { "login" : "admin," "password" : "vPr0tect" }
s.post(url+"session/login", json=payload, verify=False)
#getting the first backup destination from Stoware Backup & Recovery
r = s.get(url+"policies/vm-backup/detailed")
backupPolicies = r.json()
for backupSLA in backupPolicies:
   print("Average backup size for policy {} is {}".format(backupSLA["name"], backupSLA["averageBackupSize"]))
```

Summary

Building a Backup as a Service (BaaS) in large environments like OpenStack can be challenging. This article covered several areas that should be considered when designing such a solution and how these can be addressed using Storware Backup & Recovery.

www.storware.eu