iSCSI SAN: Deploy High-Availability iSCSI SAN through Docker Container

Wednesday, Oct 20, 2021

KernSafe Technologies, Inc.

www.kernsafe.com

Table of Contents

Overview

Docker is an open source application container engine that allows developers to package their applications and dependencies into a portable image that can then be distributed to any popular Linux , Windows or macOS machine, as well as virtualization. Containers are completely sandboxed and have no interface with each other.

Start the container by running images. An image is an executable package that contains everything you need to run the application - code, runtime, libraries, environment variables, and configuration files.

Containers are examples of images runtime - while being executed (that is, images state, or user process) in memory, you can use commands to view a list of docker ps that are running containers, just as in Linux.

Docker is now very important in the cloud and edge computing, through Docker images and containers, user can deploy KernSafe iSCSI SAN service very easily, by leveraging Docker technology, users can very quickly to provide iSCSI service from Windows, Linux or macOS server.

The document provides step-by-step guide for user to deploy iSCSI service through Docker container.

Deploy SuperSAN Docker Image

We have created an existing Docker image where the KernSafe iSCSI SAN server has been per-installed. User can use the image to quickly deploy iSCSI SAN in 1 minute. Or user can create their own defined Docker image.

Pull the existing SuperSAN Docker Image

In the Docker host machine, issue the following command to pull the existing Docker image:

#docker pull kernsafe/supersan

After downloading completed, issue the command to see if the image placed there:

#docker images

```
[root@localhost ~]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
[root@localhost ~]# docker pull kernsafe/supersan
Using default tag: latest
Trying to pull repository docker.io/kernsafe/supersan ...
latest: Pulling from docker.io/kernsafe/supersan
f34b00c7da20: Pull complete
2ad0958a073d: Pull complete
45c293c87634: Pull complete
Digest: sha256:766f5069a03c63b9ab75a831886fa9cbfbeb00b26f6e6f891da29da0c47c0457
Status: Downloaded newer image for docker.io/kernsafe/supersan:latest
[root@localhost ~]# docker images
REPOSITORY                    TAG                 IMAGE ID            CREATED             SIZE
docker.io/kernsafe/supersan   latest              9311d52975e6        6 weeks ago         572 MB
[root@localhost ~]#
```

Install SuperSAN into new Docker Image

User can install SuperSAN into any Linux based container, there is no difference between installing into Linux physical machine and container, please refer to the white paper to install KernSafe iSCSI SAN into Docker container.

https://www.kernsafe.com/tech/supersan/install-and-using-kernsafe-iscsi-san-on-linux.pdf

Of choosing the OS, CentOS 7.x and Ubuntu Server 20+ are recommended.

Prepare Datastore

User can use a local storage in the Docker host machine, could be a local HDD/SSD/NVMe, using mkfs to format a suitable filesystem and then mount to a folder, like:

#mkfs.ext4 /dev/nvme0n1
#mkdir /mnt/iscsi
#mount /dev/nvme0n1 /mnt/iscsi

Launch the Docker container

User can issue the following command to start a container, as the SuperSAN need “init” process, user should start the init process.

docker run -tid --privileged=true -p 192.168.80.21:3260:3260 -p 192.168.80.21:3268:3268 -p 192.168.80.21:3261:3261 -v /mnt/iscsi:/iscsi 9311d52975e6 /sbin/init

We need -p to specify port forwarding from the container to the host, specify IP address to let Docker to select which network to offer iSCSI SAN service, we need to open the following ports:

3260, iSCSI service.
3261, iSCSI management console port.
3268, iSCSI Web based management console port (optional).

Use -v to passthrough host datastore path into the Docker container.

9311d52975e6 is the image id, use need to replace one that shown in the docker images command.

Then issue the command to enter the new created container:

#docker ps
#docker exec -it 6df08fc706d4 /bin/bash

6df08fc706d4 is the Docker container ID, user need to replace it as shown in the “docker ps” command.

In the docker container, issue the command to see if KernSafe iSCSI SAN service is running well:

```
[root@6df08fc706d4 /]# service supersand status
● supersand.service - LSB: supersan
   Loaded: loaded (/etc/rc.d/init.d/supersand; bad; vendor preset: disabled)
   Active: active (running) since Tue 2021-10-19 06:31:31 UTC; 1min 23s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 408 ExecStart=/etc/rc.d/init.d/supersand start (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/docker-6df08fc706d4d212c1c2201a85634c7147a687db16c8ee7e0a1d70a483848b14.scope/system.slice/supersand.service
           └─412 /opt/SuperSAN/supersan

Oct 19 06:31:28 6df08fc706d4 systemd[1]: Starting LSB: supersan...
Oct 19 06:31:29 6df08fc706d4 supersand[408]: Starting KernSafe SuperSANpid: 411, hpage size: 2097152
Oct 19 06:31:31 6df08fc706d4 systemd[1]: Started LSB: supersan.
[root@6df08fc706d4 /]#
```

Set up High Availability iSCSI SAN

HA iSCSI SAN is important service for providing Uninterrupted service. User can skip this step if user do not need to create HA iSCSI SAN service.

Launch the Docker container

Now we use 192.168.0.101, 192.168.0.102 as the storage network IP, please see above topics for details.

Launch the first Docker container.

docker run -tid --privileged=true -p 192.168.0.101:3260:3260 -p 192.168.0.101:3268:3268 -p 192.168.0.101:3261:3261 -v /mnt/iscsi:/iscsi 9311d52975e6 /sbin/init

Enter into the first container, this step is optional.

#docker ps

#docker exec -it 6df08fc706d4 /bin/bash

Launch the second Docker container (should be on another host).

docker run -tid --privileged=true -p 192.168.0.102:3260:3260 -p 192.168.0.102:3268:3268 -p 192.168.0.101:3261:3261 -v /mnt/iscsi:/iscsi 9311d52975e6 /sbin/init

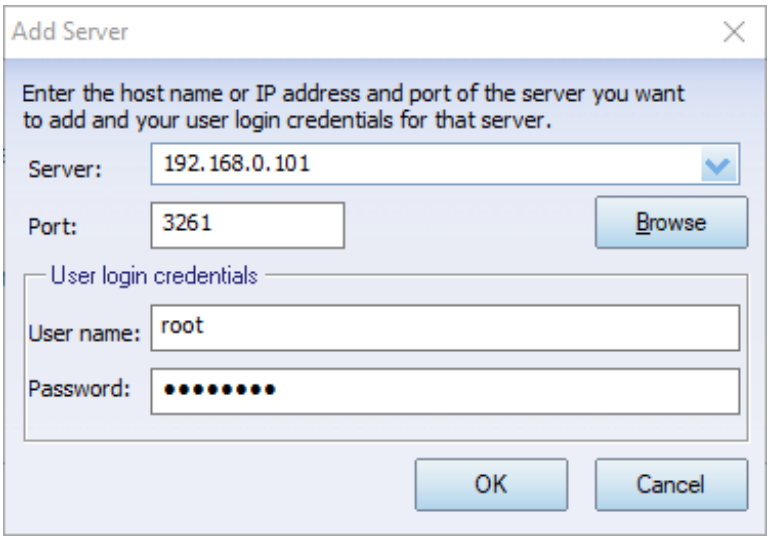
Enter into the first container, this step is optional.

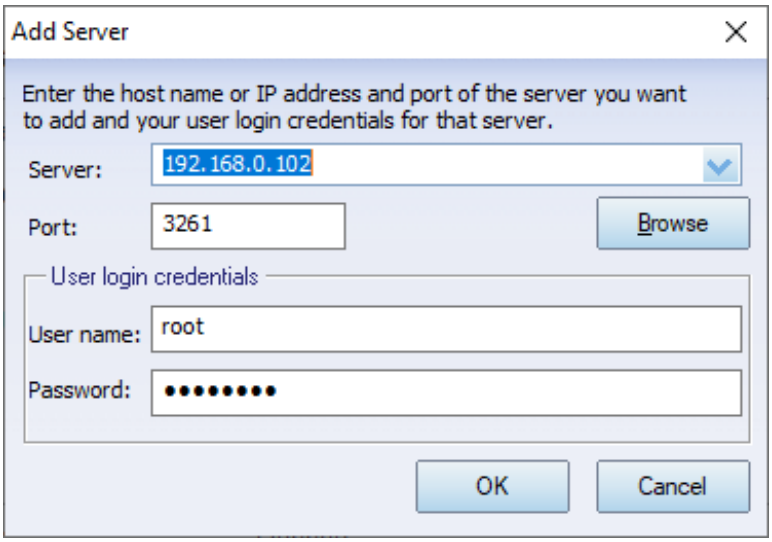
#docker ps

#docker exec -it 6df08fc706d4 /bin/bash

Configuring in the Management Console

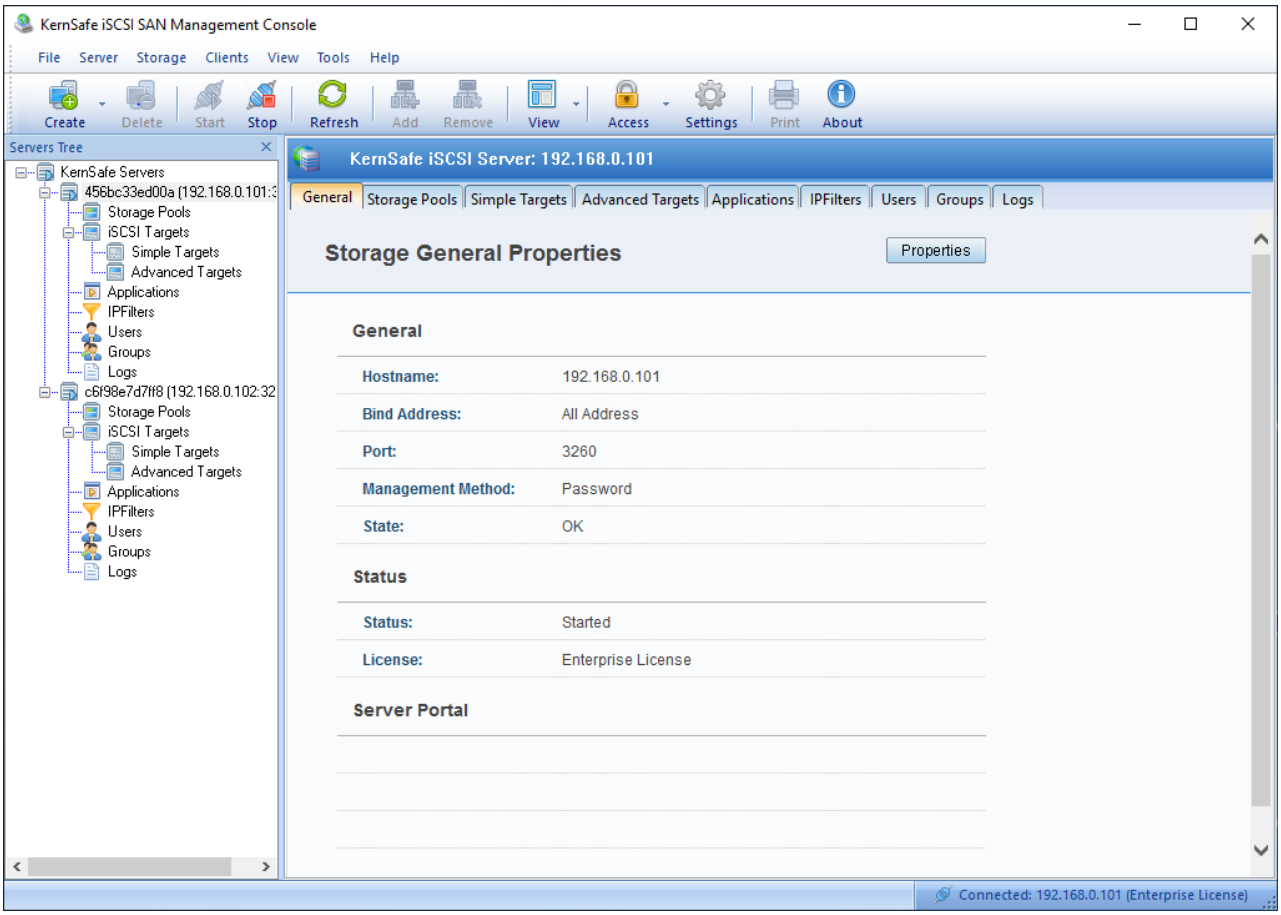
Open **KernSafe iSCSI SAN Management Console.** Click Server->Add Another Server.





Type the two IP address on each Add Server dialog, and then press the OK button.

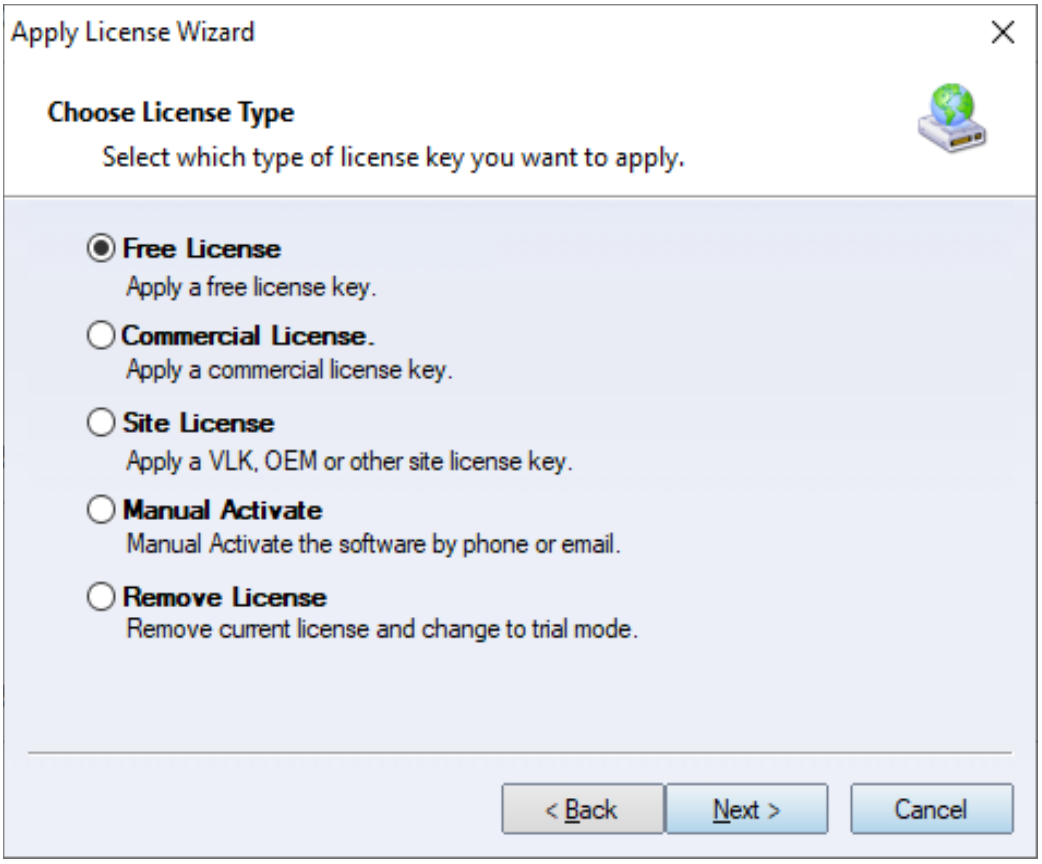
The main interface is shown as this.



Acquire and install license keys

Users need license key for each KernSafe iSCSI SAN instance, a trial license key will be automatically acquired through internet when first run, if the progress failed, user may ask us to obtain a trial or free license key or purchase commercial license keys.

Select each server node and click on the menu item Help->Apply License, then the Apply License Wizard shows.

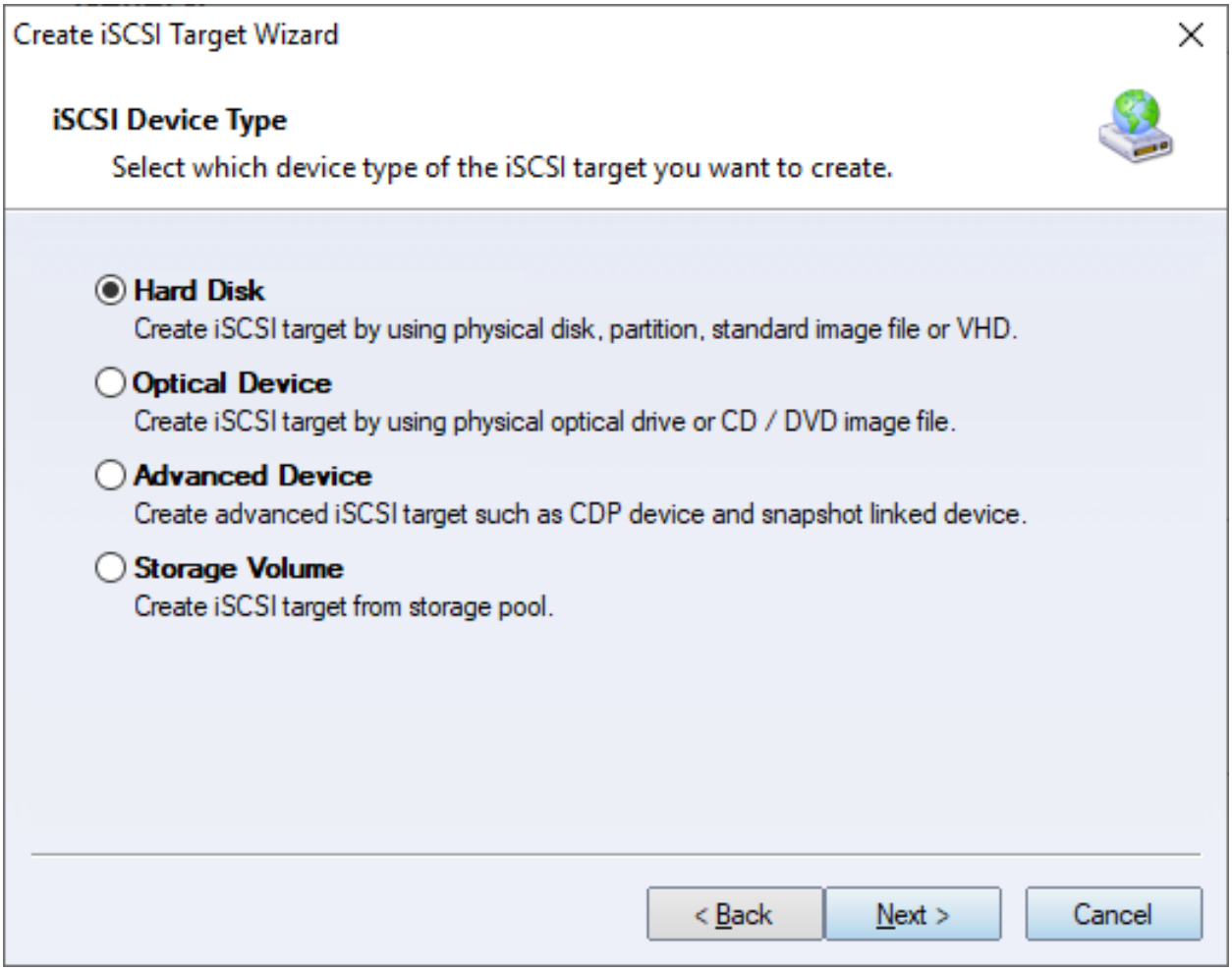


Select the license type and fulfilled the form and then press the Finish button, then user will see license information in the abort box or current server's generic information page.

Create Target on the first server

Select on the first server node, and press the **Create** button on the toolbar of KernSafe iSCSI SAN management console, the **Create Device Wizard** is shown.

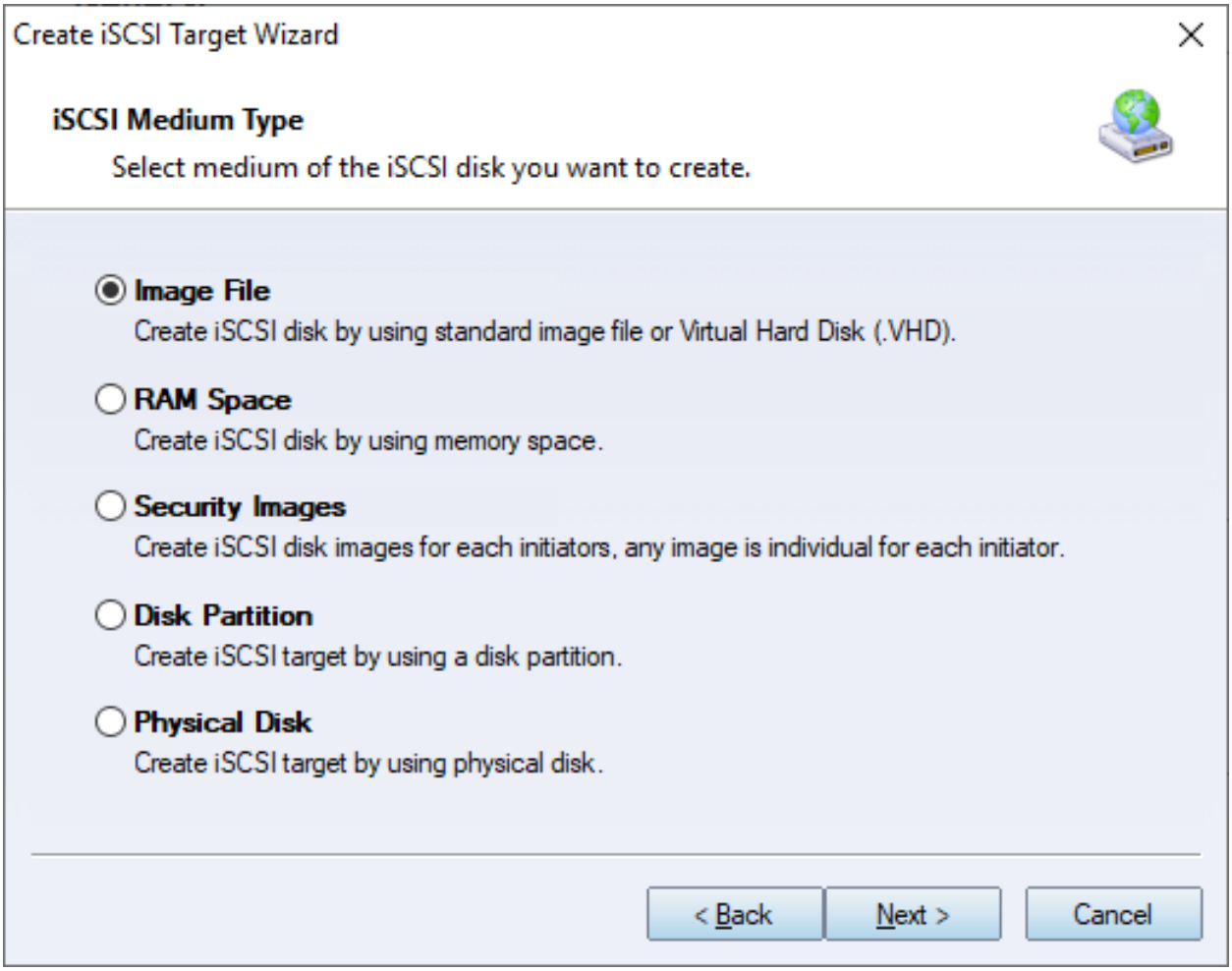
Select a device type



Choose **Hard Disk**.

Press the **Next** button to continue.

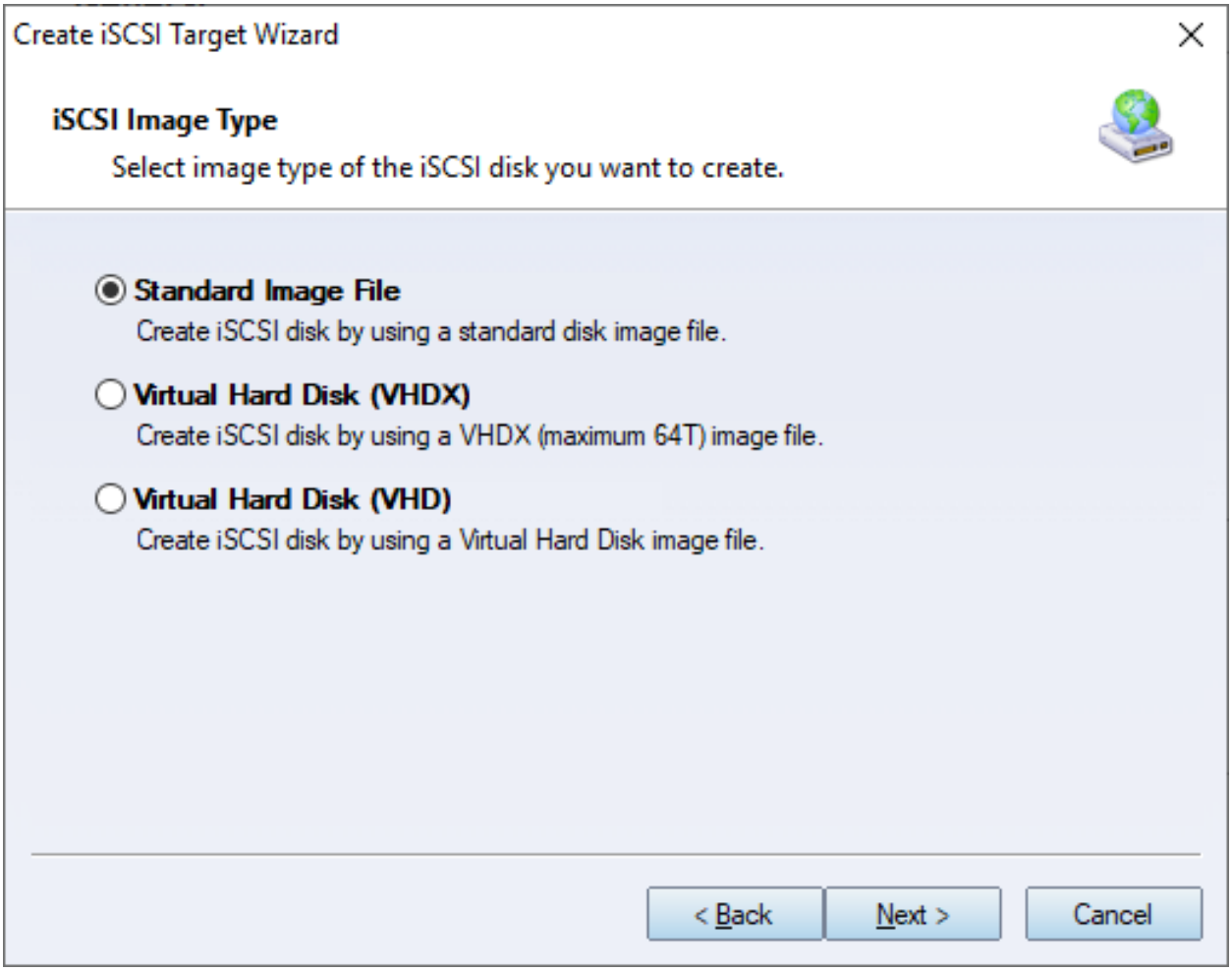
Select a medium type.



Choose **Image File** in **iSCSI Medium Type** window.

Then press **Next** button to continue.

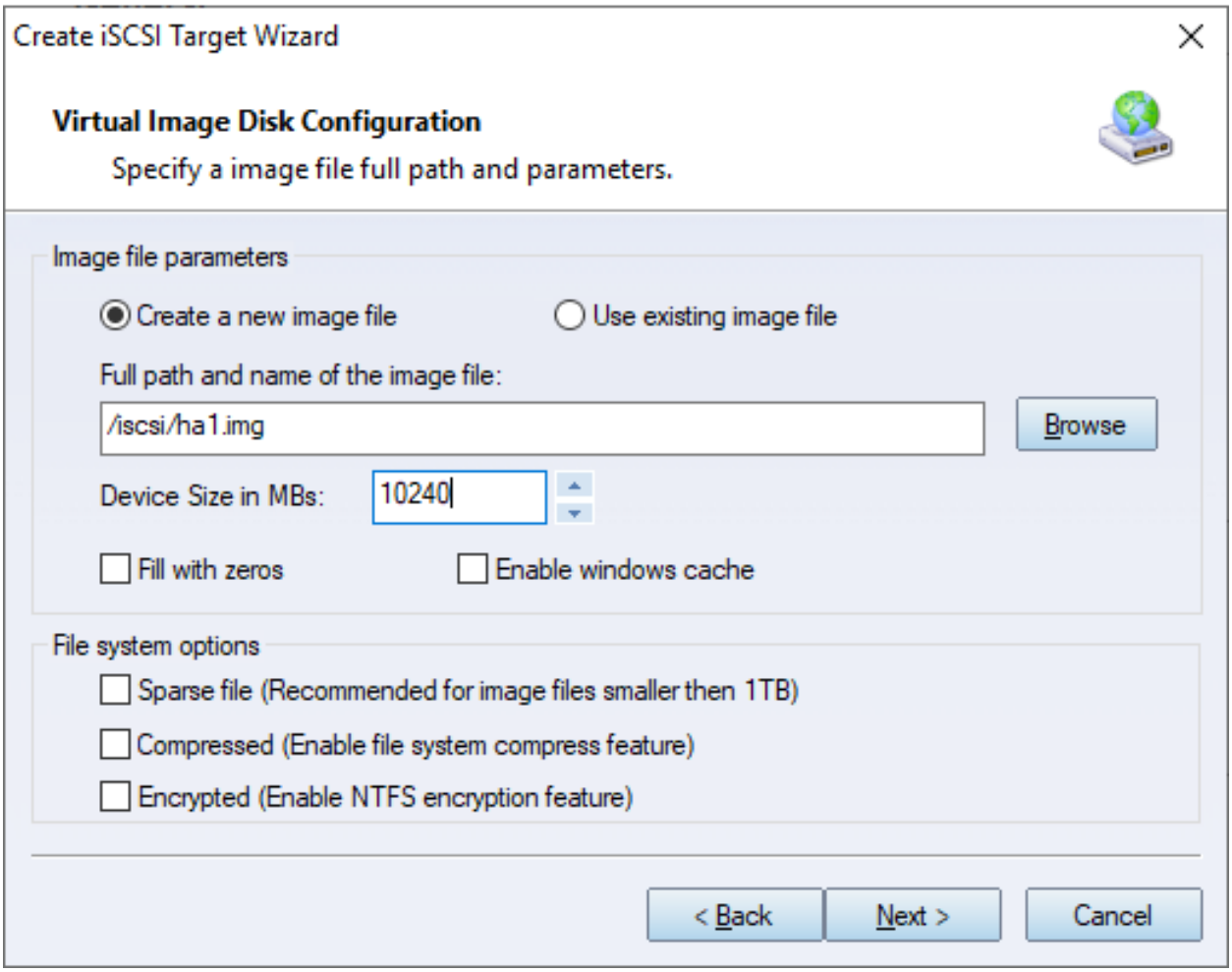
Select an Image type.



Choose **Standard Image File.**

Press the **Next** button to continue.

Specify image file path and size.



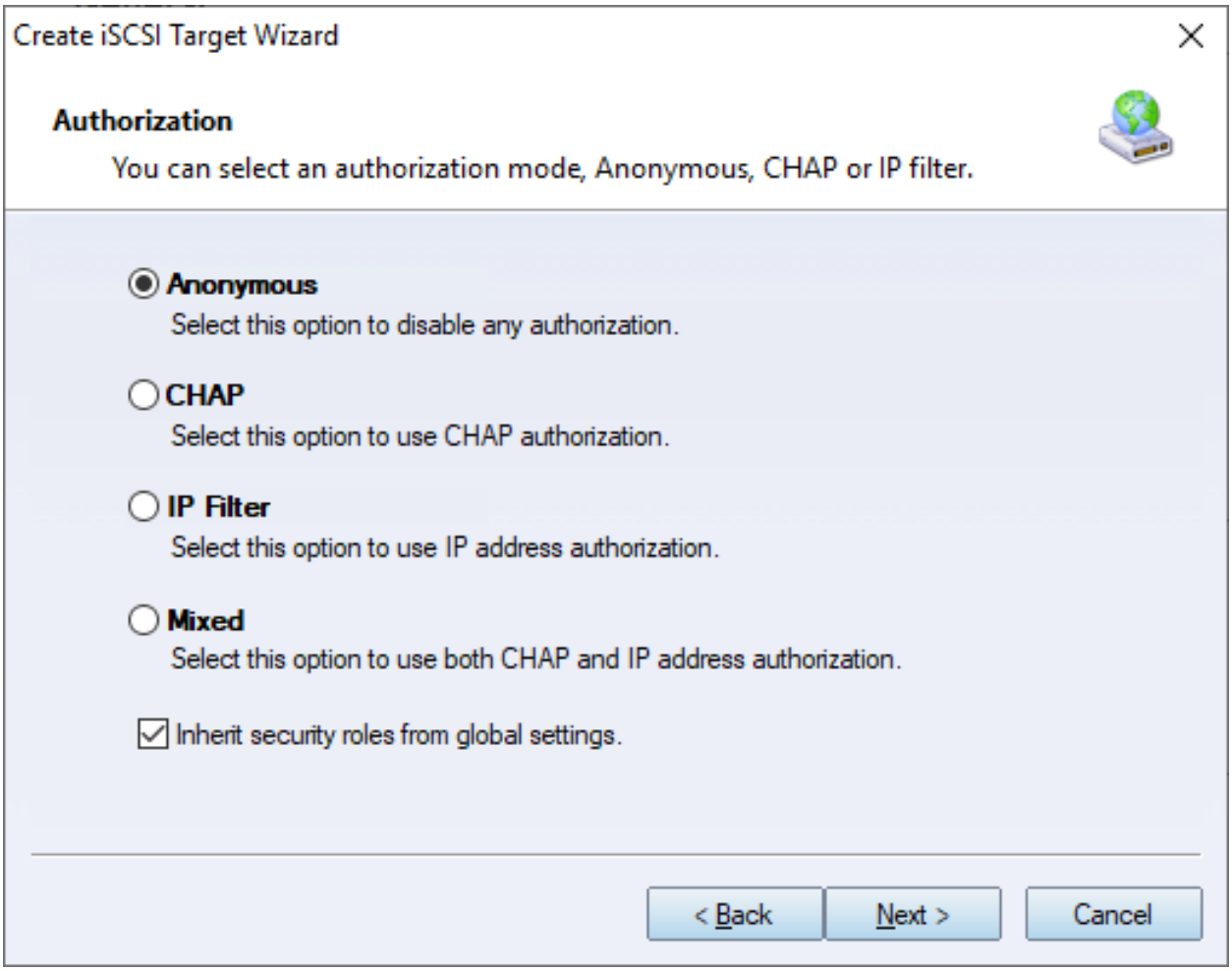
Specify the image file.

Specify the device size.

If you check **Use sparse file on NTFS file system**, the size of disk image file only depends on its content used, it can save your hard disk space.

Press the **Next** button to continue.

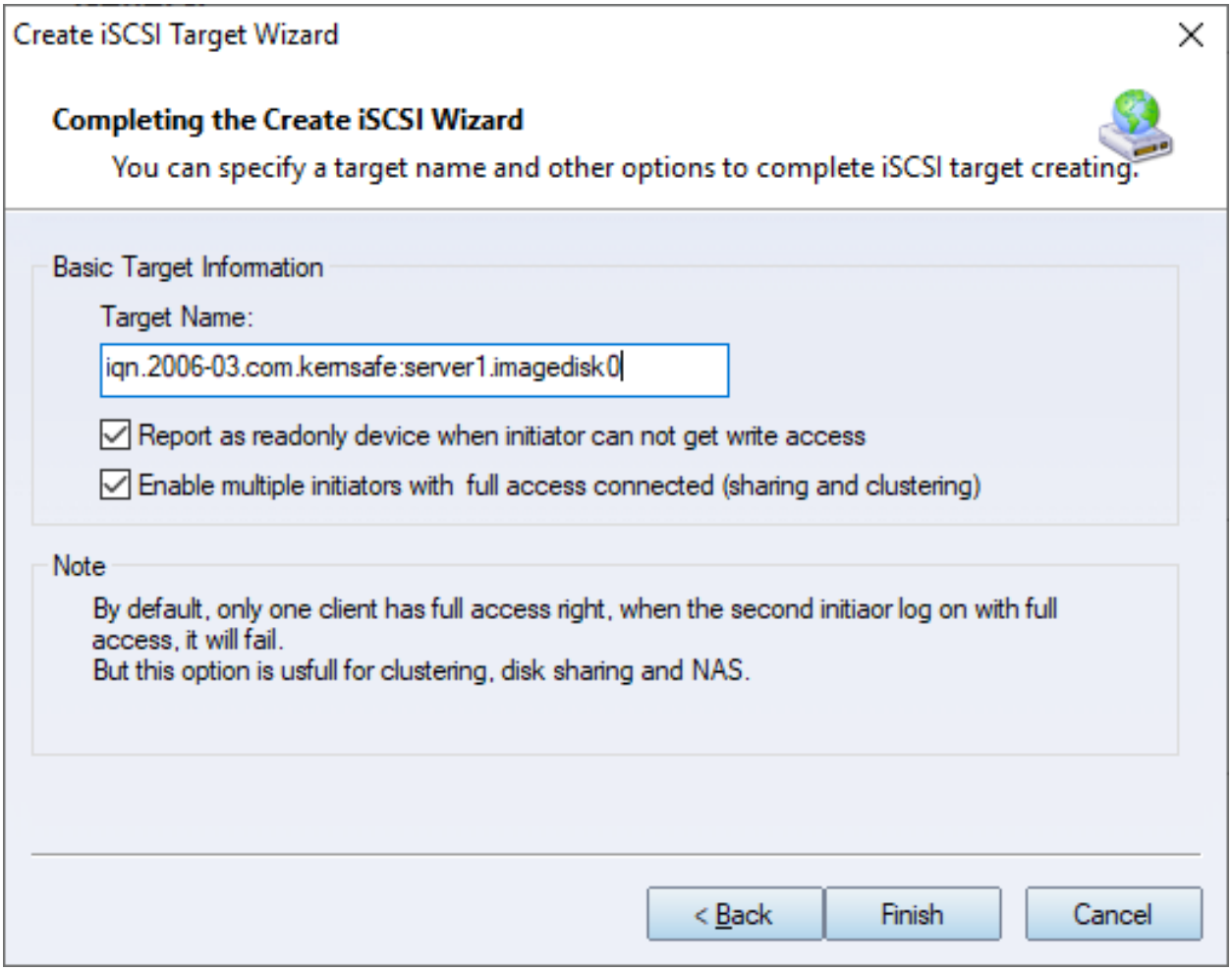
Set authorization mode.



Choose **Anonymous** authorization.

Press the **Next** button to continue.

Finish creating iSCSI Target



Type a target name in the Target Name field, we use **server1.imagedisk0** as an example.

Check the **Enable multiple initiators with full access connected (sharing and clustering)** check box.

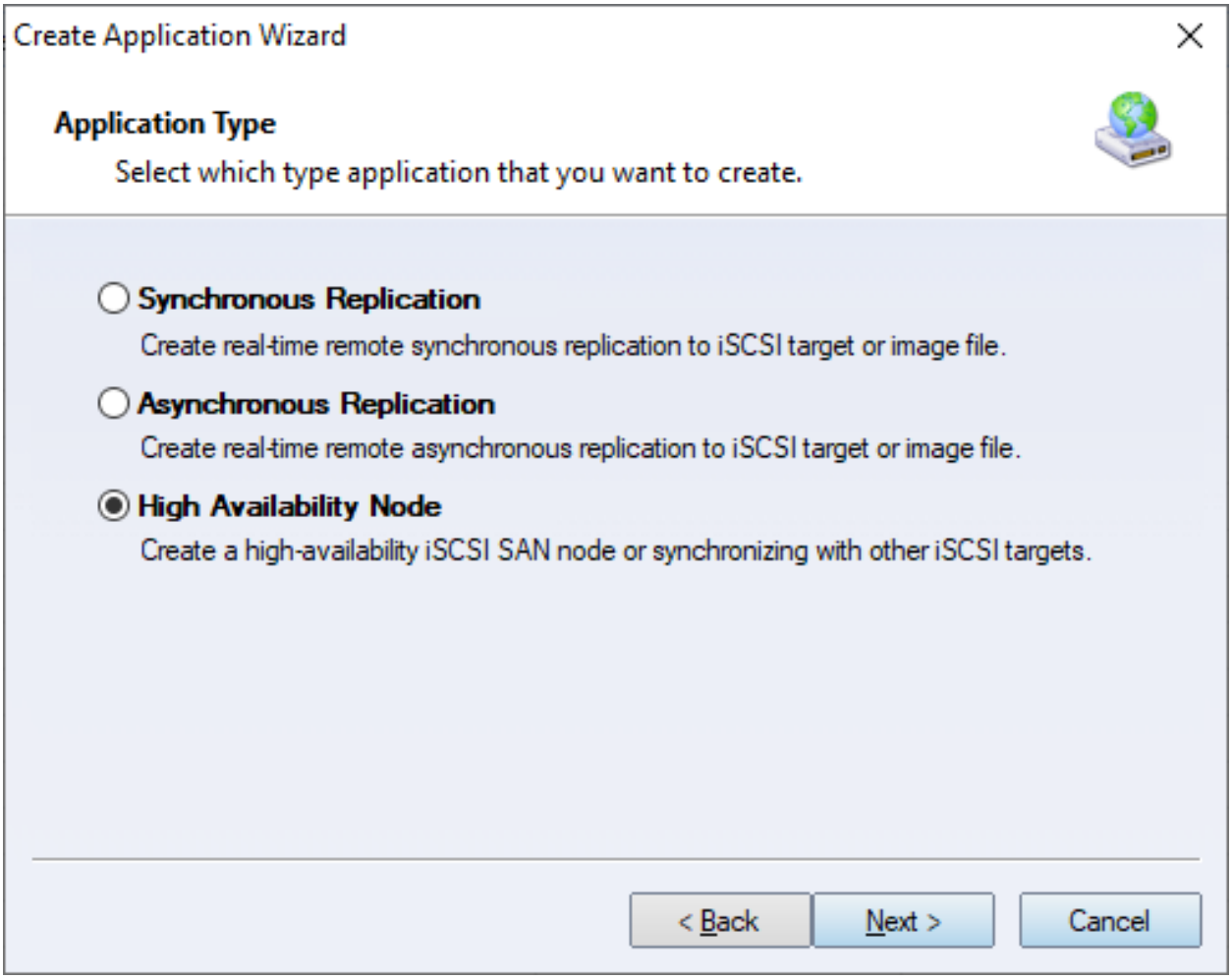
Press the **Finish** button to complete create target.

Create Target on the second server

Select on the second server node in the KernSafe iSCSI SAN management console, and then repeat the above steps to create the second target as the name server2.imagedisk0.

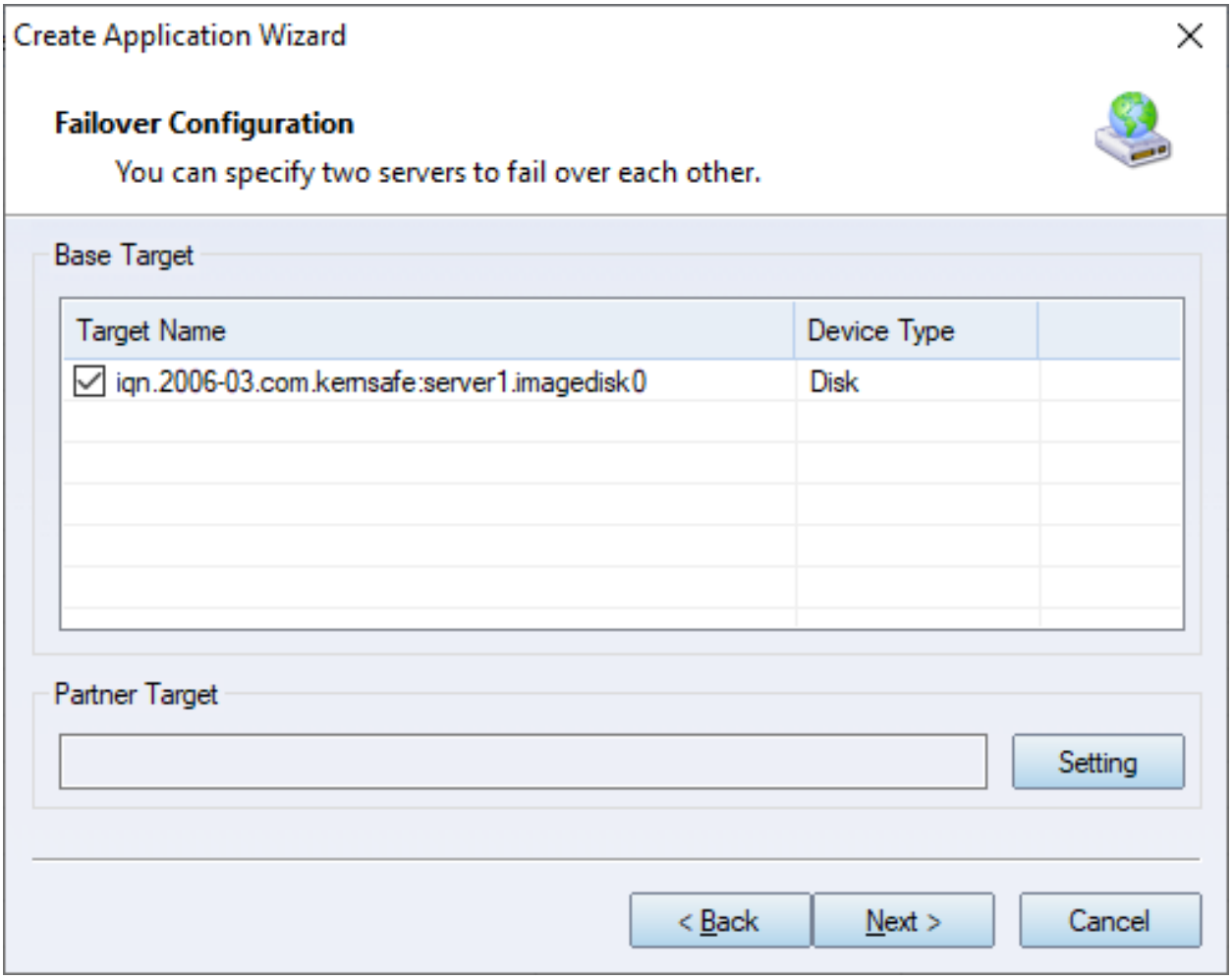
Creating Application on server1

Select on the first server, right click **Applications** on the left tree of the main interface, choose **Create Application** on the pop-up menu, the **Create Application Wizard** widow will be shown.

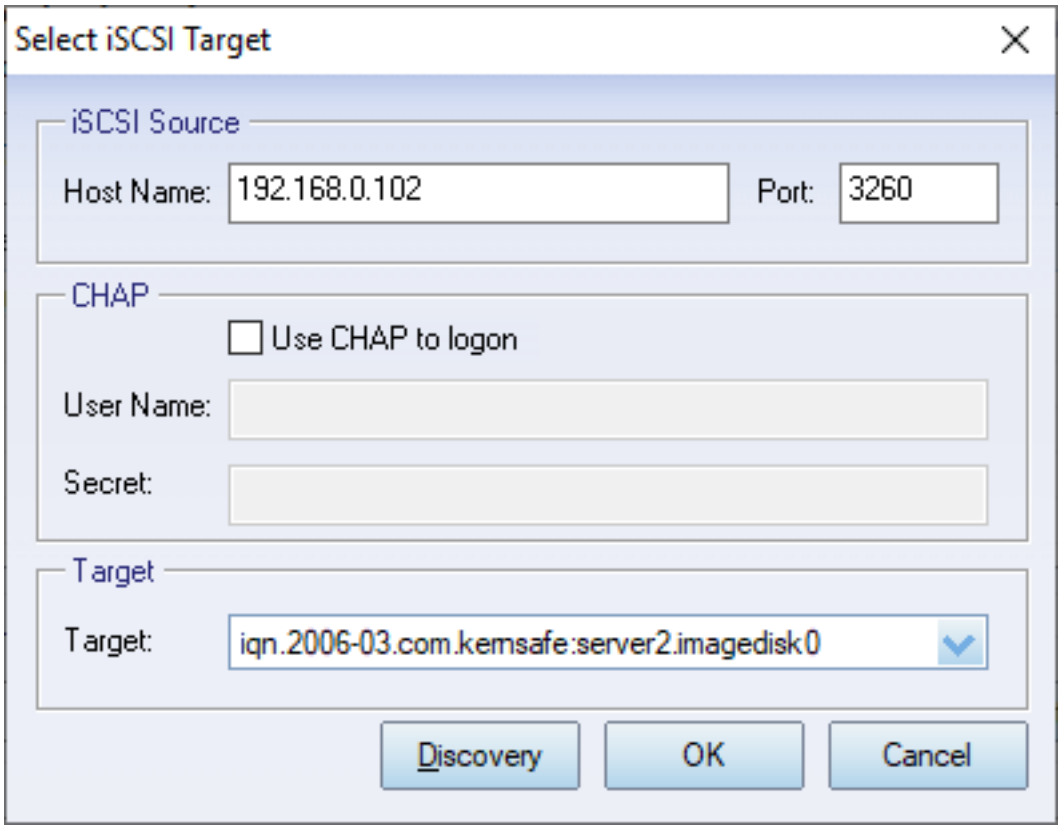


Choose **High Availability Node**.

Then press **Next** to continue.



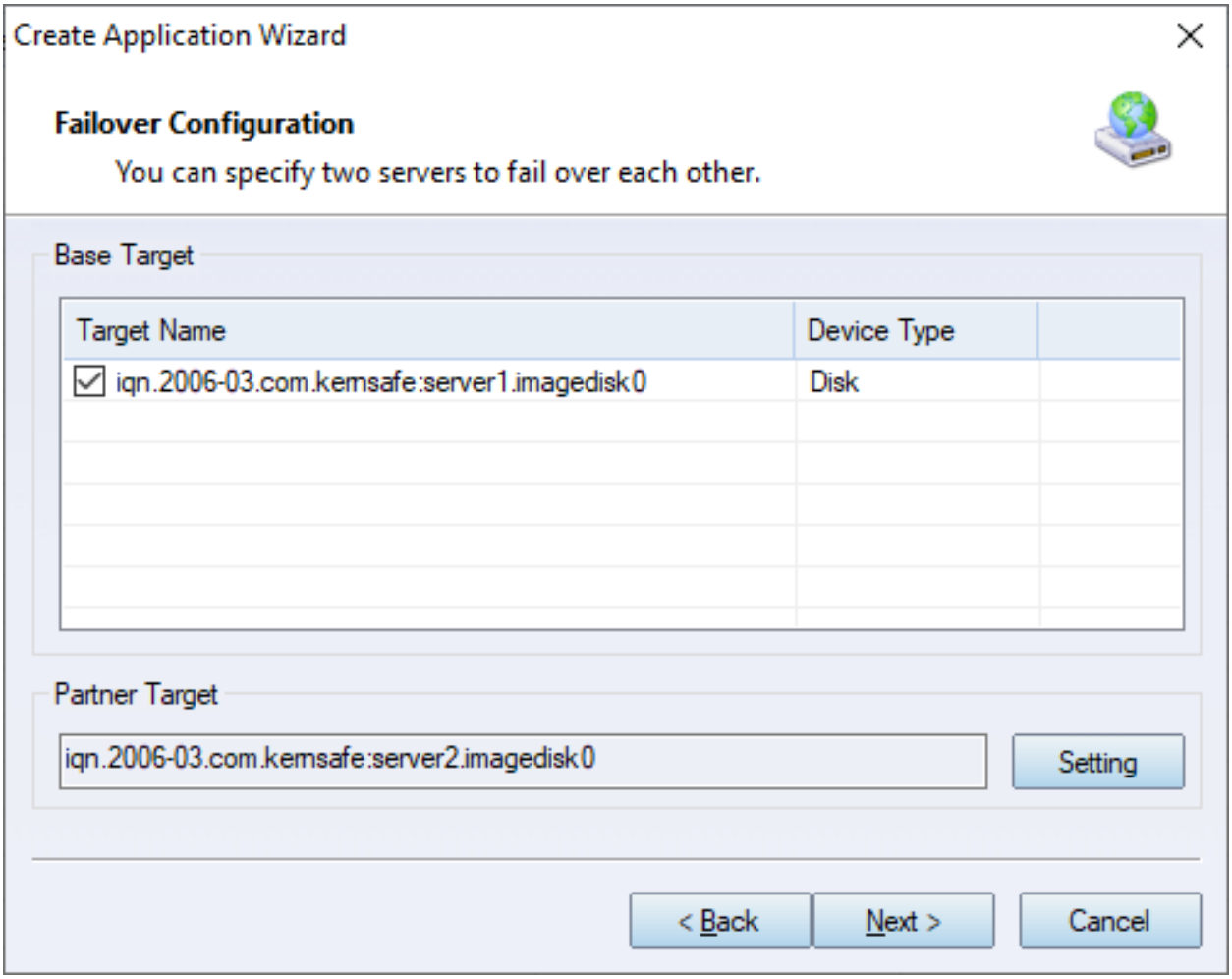
Check to select the existing target storage and click **Edit** to find the remote HA target.



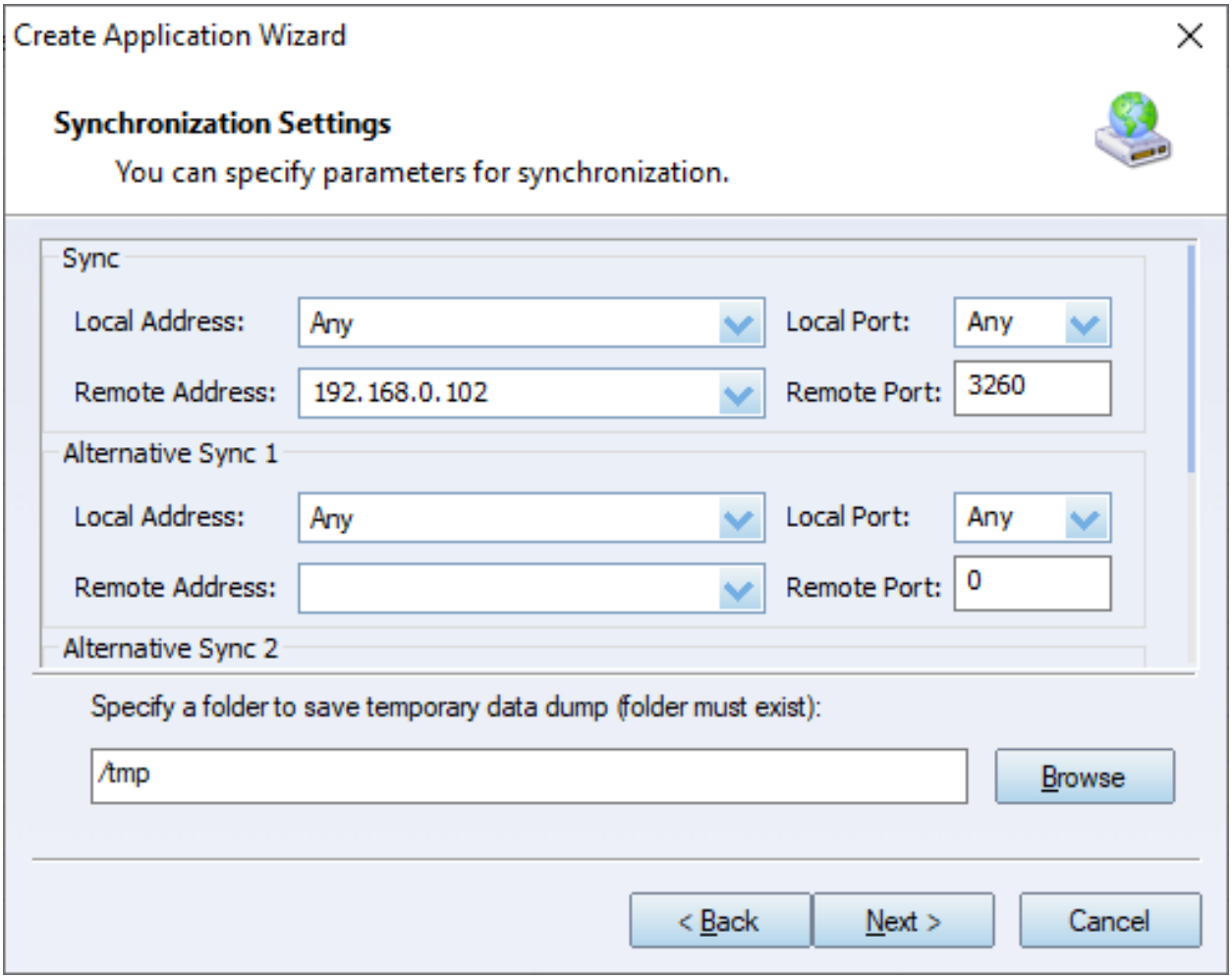
Input the IP and port of server2 in **iSCSI Source** tab, and then click **Discovery** on the bottom of the window to find the mirror target, choose the new created target in the down-list.

Press **OK** button to continue.

Note: If the target needs CHAP authorization, you should provide user name and secret to logon.



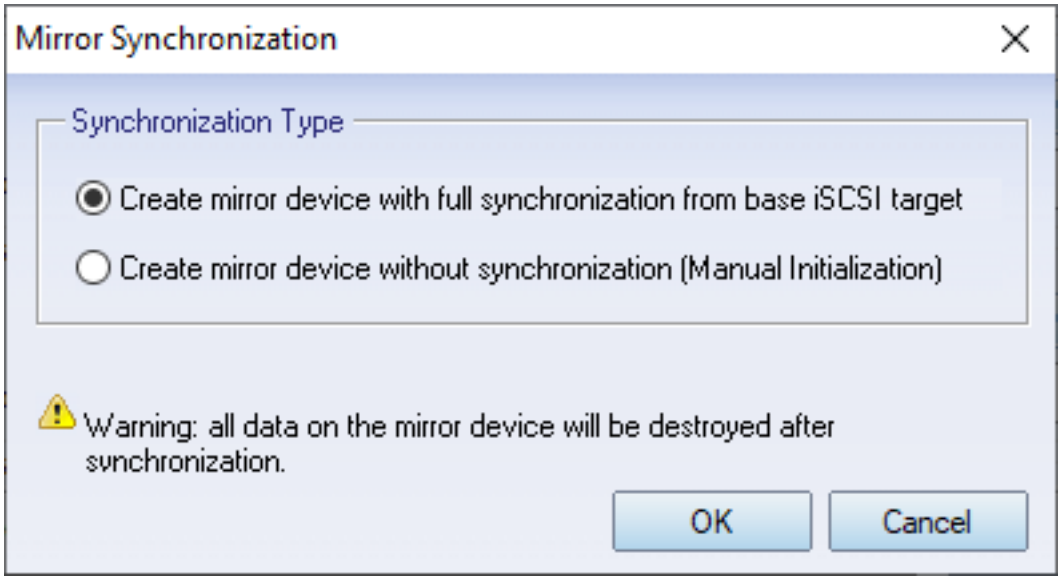
The mirror target will be added to the window, then click **Next** button to continue.



Specify local interface, port for Sybc interface and Heartbeat interface, if you have two NIC for each server ,you can sepcify different address-pair for Sync interface and Heartbeat interface, if you have only one NIC for synchronous, you can use same address for Sync and Heartbeat.

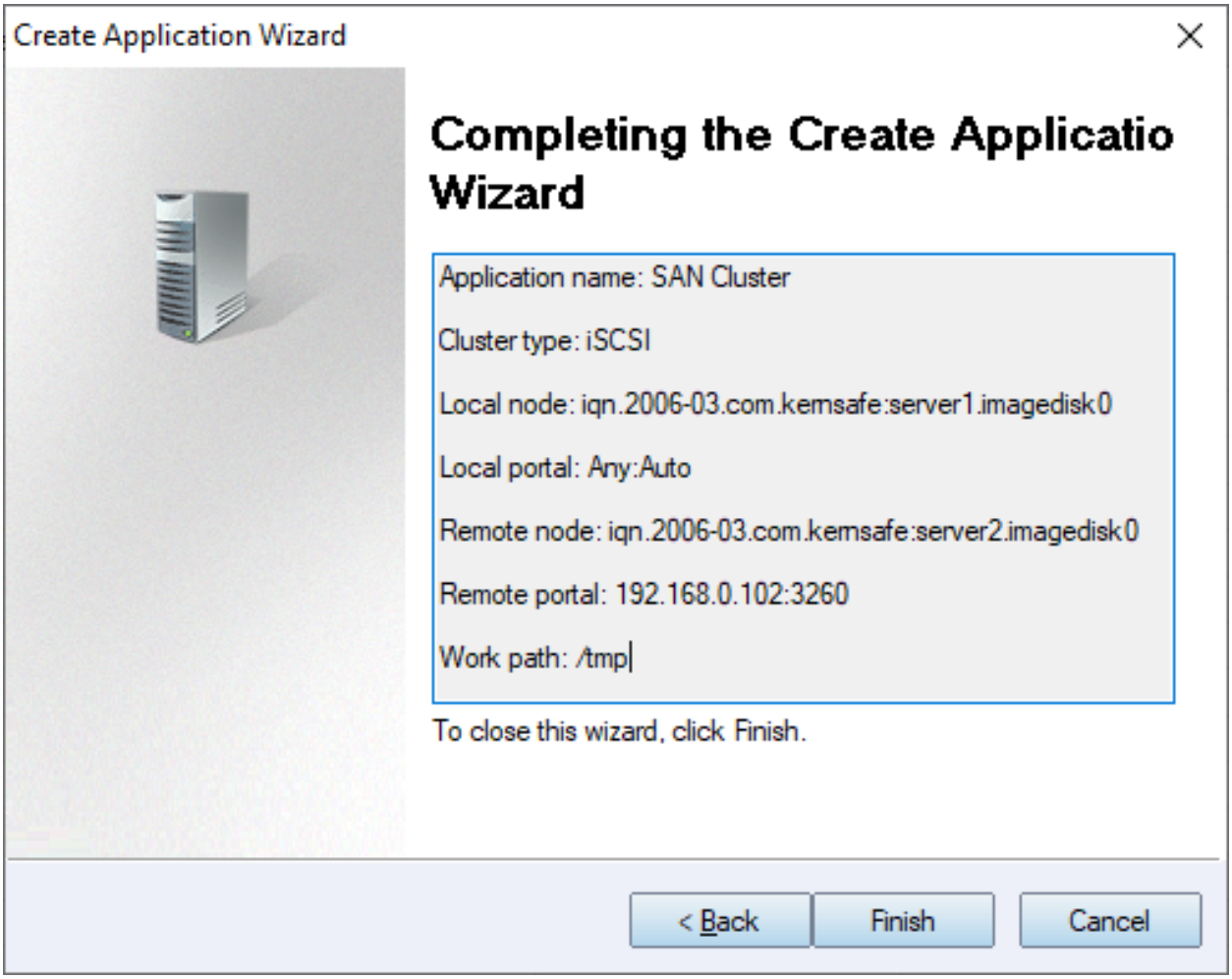
Specify the portal and port.

Press **Next** to continue



Now, the mirror target should be synchronized to the base target, if the two targets are both the new one and do not be initialized, we can choose **Create mirror device without synchronization (Manual Initialization)**, otherwise, we must choose **Create mirror device with full synchronization from base iSCSI target**.

Press **OK** button to continue.



Click **Finish** button to complete the application creation.

Creating Application on server2

Select on the second server, right click **Applications** on the left tree of the main interface, choose **Create Application** on the pop-up menu, repeat the above steps to create the second HA iSCSI SAN application that pointing to the target in the first server.

Then application server (client machine like Windows Server, Linux, ESX/ESXi, XenServer etc.) may use the two iSCSI targets for HA iSCSI device.

Contact

Support:	support@kernsafe.com
Sales:	sales@kernsafe.com
Marketing:	marketing@kernsafe.com
Home Page:	http://www.kernsafe.com
Product Page:	https://www.kernsafe.com/product/iscsi-san-linux.aspx
Licenses	https://www.kernsafe.com/product/iscsi-san-linux/license-compares.aspx
Forum:	http://www.kernsafe.com/forum

KERNSAFE
Secure Storage Solution

KernSafe Technologies, Inc.

www.kernsafe.com