

# Install Backup Server

This section is the part of the OnApp installation procedure.

[Install Control Panel Server](#) > [Install Compute Resources](#) > [Install Data Stores](#) > **Install Backup Server** > [Configure vCloud Director Integration](#) > [Configure Cloud](#)

Follow one of two processes to set up a backup server in your cloud:

- Install Static Backup Server
- Install CloudBoot Backup Server

Choose the one that suits you best.

## On this page:

- [Install Static Backup Server](#)
- [Install CloudBoot Backup Server](#)

## See also:

[Install Control Panel Server](#)

[Install Data Stores](#)

[Technical Details](#)

[Preparation Guide](#)

Search for other docs:

## Install Static Backup Server

To install static backup server, run the following procedure. Skip this section if you are using a CloudBoot method. We recommend installing CentOS from the minimal CentOS ISO for static backup servers.

1. Add a backup server via the Control Panel user interface:
  - a. Go to your Control Panel's **Settings** menu, then press **Backup servers** icon.
  - b. Click the **Create Backup Server** button.
  - c. Fill in the form that appears:
    - *Label* - give your backup server a label
    - *IP address* - enter the backup server IP address (IPv4)
    - *Backup IP address* - add a provisioning network IP address
    - *Capacity* - set the backup server capacity (in GB)
    - *Backup server zone* - select the backup server zone to which this backup server will be assigned.
  - d. Move the **Enabled** slider to the right to enable the backup server.
  - e. Click the **Add Backup Server** button.

2. Update your server:

```
bash# yum update
```

3. Download the OnApp repository:

```
bash# rpm -Uvh http://rpm.repo.onapp.com/repo/onapp-repo-5.4.noarch.rpm
```

4. Install the OnApp Backup Server installer package:

```
bash# yum install onapp-bk-install
```

5. Check and set Backup Server default settings:

**OnApp BK tools custom version**

```
BK_VERSION= " "
```

#### OnApp StorageAPI custom version

```
API_VERSION= " "
```

#### Default server to synch time on the HV

```
NTP_TIME_SERVER='pool.ntp.org'
```

#### The number of retries for WGET to download the file

```
WGET_TRIES=5
```

#### OnApp templates directory.



Please refer to the corresponding settings at OnApp Control Panel web interface

```
TEMPLATES_DIR='/onapp/templates'
```

#### OnApp backups directory.



Please refer to the corresponding settings at OnApp Control Panel web interface

```
BACKUPSS_DIR='/onapp/backups'
```

```
bash# vi /onapp/onapp-bk.conf
```

6. Run the installer. It is recommended to download Base, Load Balancer and CDN templates while running the installer. You may rerun the installer later with the -t option.

```
bash# sh /onapp/onapp-bk-install/onapp-bk-install.sh
```

#### Usage:

```
/onapp/onapp-bk-install/onapp-bk-install.sh [-c CONFIG_FILE] [-a] [-y] [-t] [-h]
```

#### Where:

-c CONFIG_FILE	Custom installer configuration file. Otherwise, preinstalled one is used.
-a	Do NOT be interactive. Processe with automatic installation.
-y	Update OS packages (except of OnApp provided) on the box with 'yum update'.
-t	Initiate Base, Load Balancer and CDN templates download. The download is initiated if '-a' option is used.
-h	Print this info

7. onfigure the backup server for your cloud. This step is also required for the SNMP statistics receiver configuration:

```
bash#> /onapp/onapp-bk-install/onapp-bk-config.sh -h <CP_HOST_IP> -p [BK_HOST_IP]
```

**Usage:**

```
/onapp/onapp-bk-install/onapp-bk-config.sh [-h CP_HOST_IP] [ -p BK_HOST_IP] [-a|-i [USER:PASSWD]] [-s] -?
```

**Where:**

-h CP_HOST_IP	FQDN or IP Address of the management server which should receive all status reports and is authoritative for this backup server.
-p BK_HOST_IP	FQDN or IP Address of Backup Server which will serve all stats related and other requests send by the CP_HOST_IP. Used by snmpd and StorageAPI.
-a	Install AoE
-i [USER:PASSWD]	Install iSCSI utils and configure with USER and PASSWD (if specified)
-s	Install sshfs
-?	Print this help info

8. Run the following commands:

```
# yum install gdisk fuse fuse-libs fuse-devel qemu-img libvmdk libvmdk-tools
```

## Install CloudBoot Backup Server

CloudBoot backup servers are CloudBooted KVM Compute resources that can be used as backup servers. Follow the step-by-step instructions provided in this chapter to configure CloudBoot backup servers in your cloud.



- You should configure some local or remote attached storage for persistent backups on the provisioning/backup server. We strongly recommend you to deploy one or more backup servers for backups and VS provisioning when using a CloudBoot functionality.
- We strongly recommend you to deploy one or more backup servers on your cloud. Incremental backups are only supported with a dedicated backup server.
- CentOS now defaults to NFSv4. This is known to cause compatibility issues so we strongly recommend that you use NFSv3 for all mounts. This can be done by passing `-t nfs -o vers=3` in any mount commands.
- To use the backup server, you have to add it to a [backup server zone](#) and assign it either to a [compute resource](#) or a [compute zone](#).

To create a CloudBoot backup server:

1. Update CloudBoot and CP server RPMs:

```
yum update onapp-store-install yum update onapp-cp-install
```

2. Configure CloudBoot settings:

```
/onapp/onapp-store-install/onapp-store-install.sh
```

3. Create new CloudBoot compute resource: at the first step of the creation wizard choose the type - Backup (CloudBoot Provisioning and Backup Resource). Refer to the [Create CloudBoot Compute Resource](#) section of this guide for details.
4. Go to your Control Panel's **Settings** menu, then press **Compute Resources** icon.
5. Click the **Add New CloudBoot Compute Resource** button.
6. In wizard menu click Backup (CloudBoot Provisioning and Backup Resource).
7. Select right MAC address of it.
8. Fill in the form that appears:
  - *Label* - give the compute resource a name
  - *Pxe IP address* - select an IP address for this compute resource from the address pool available
  - *Enabled* - move the slider to the right to allow VSs to be installed/booted on this compute resource
  - *Compute Zone* - select the compute zone, to which this compute resource will be assigned, from the drop-down list
  - *Custom Config* - specify any custom commands you want to run when compute resource is booted
9. At this step, specify the Backup Server properties:
  - *Label* - give your backup server a label
  - *IP address* - enter the IP address of a Compute resource you have created at step 1
  - *Backup IP address* - add a provisioning network IP address
  - *Capacity* - set the backup server capacity (in GB)
10. Then, you shouldn't assign any drive, but it needs right NIC to be applied to the storage network.
11. After that, assign your backup server to the backup server zone.

If you intend to attach LVM-based storage and create backups, you should also add the IP address of the KVM Compute resource added in step 1 in the 'Backup IP address' field of each of your compute resources.

**Further steps:**

1. Format and mount the local storage:

- a. SSH to the backup server
- b. Format the storage with your preferred filesystem type, e.g.:

```
bash#> mkfs.ext4 /dev/sda
```

- c. Make folder for backups if it does not exist

```
bash#> mkdir /backupstorage
```

- d. Mount the storage to /onapp/backups:

```
bash#> mount /dev/sda /backupstorage
```

- e. Make folder for storing templates:

```
bash#> mkdir /backupstorage/templates
```

- f. Make folder for storing backups:

```
bash#> mkdir /backupstorage/backups
```

- g. Create symbolic links in /onapp:

```
bash#> ln -s /backupstorage/backups /onapp/backups
bash#> unlink /onapp/templates
bash#> ln -s /backupstorage/templates /onapp/templates
bash#> ln -s /onapp/tools/recovery /data
```

- h. Add the following to custom config file:

```
mkdir /backupstorage
mount /dev/sda /backupstorage
ln -s /backupstorage/backups /onapp/backups
unlink /onapp/templates
ln -s /backupstorage/templates /onapp/templates
ln -s /onapp/tools/recovery /data
```

2. Update the database so that the location of the templates is known:

a. Find the database password:

```
cat /onapp/interface/config/database.yml |grep password
```

b. Open the onapp database in MySQL:

```
bash#> mysql -p bash#> use onapp;
```

c. Find the ID of the backup server:

```
bash#> select * from backup_servers;
```

d. For all of the templates, set the required backup\_server\_id:

```
bash#> update templates set backup_server_id='[your_id]';
```

3. To download the base templates during the installation to your Control Panel, download and run the following script:

```
bash#> wget http://downloads.repo.onapp.com/get_template.sh
bash#> /bin/sh get_template.sh
```



If your backups disappear after rebooting the CloudBoot backup server with LVM storage, add mount command to CloudBoot backup server custom config after the reboot. This is a known issue which will be fixed in the future release.

To fix your custom config settings, use one of the following options provided in the examples below (you will have to specify your own device names):

1. If you have a separate partition for backups and templates (/dev/sda1 and /dev/sda2)

```
mkdir /backupstorage
mkdir /backupstorage/templates
mkdir /backupstorage/backups
mount /dev/sda1 /backupstorage/backups
mount /dev/sda2 /backupstorage/templates
ln -s /backupstorage/backups /onapp/backups
unlink /onapp/templates
ln -s /backupstorage/templates /onapp/templates
```

2. If you current array is detected as /dev/sda1 and currently everything is located in /onapp within templates and backup directories within:

```
mkdir /backupstorage
mount /dev/sda1 /backupstorage/
ln -s /backupstorage/backups /onapp/backups

unlink /onapp/templates
ln -s /backupstorage/templates /onapp/templates
```

**This section is the part of the OnApp installation procedure.**

[Install Control Panel Server](#) > [Install Compute Resources](#) > [Install Data Stores](#) > [Install Backup Server](#) > [Configure vCloud Director Integration](#) > [Configure Cloud](#)