

Backing up and restoring snapshots on Amazon EC2 machines - TechRepublic

By Nick Hardiman

Published: 2012-03-21 · Archived: 2026-04-05 18:50:01 UTC

Nick Hardiman describes several methods of backups of your Amazon EC2 machines. He also details the steps for restoring snapshot backups.

As part of my [operational readiness](#) preparation, I want to make sure my Internet service is recoverable. I covered the developers' favourite method – [application-level backup](#) – in my last post.

OS-level backup and recovery

Sysadmins prefer to back up chunks of the file system. Many [outstanding Linux backup utilities](#) can help you manage a file system, the complexity of versioning, automation, backups spread across different media and rolling back to just the right files.

OS level backups got all the attention for decades. Now it's a bit old and the younger model VM snapshots are getting all the attention. Don't be dazzled: OS-level backup is the best solution.

VM-level backup and recovery

The elephant gun approach – when you absolutely positively have to back up everything – is to copy the contents of an EC2 machine. You don't really copy the machine itself: you copy a disk volume. Behind the scenes, the volume is reduced to one fat image file and some meta-data.

AWS make it easy to litter their storage with your backups. If an enterprise uses these volumes as its backup strategy, storage use spirals. The enterprise ends up with thousands of files in many cloud storage areas and has quite a job keeping track of the what is being stored, how much is being stored, and how much it is costing. The enterprise's spanner-wielding officers archive their work using a storage service like [AWS S3](#), and bean-counting officers monitor the financial implications using a cost monitoring service like [Uptimecloud](#).

Make a backup snapshot using the AWS console

This is a simple backup procedure for taking one snapshot and storing it in [EBS](#). A snapshot is a copy of a disk volume. A snapshot is not a complete copy – it is only the bits that have changed since the last snapshot.

This procedure is similar to [launching a new VM](#).

Find the name of the volume to back up.

1. [Open the AWS console](#).

2. Navigate to the list of EC2 machines. The breadcrumbs to follow are *EC2 tab > INSTANCES > Instances*.
3. Display the root device name of the machine to back up. Tick the row in the top pane, click the *sda1* link in the bottom pane and a popup box appears with the volume's *EBS ID* name, along the lines of *vol-12345678*. Make a snapshot.
4. Open the snapshot dialog box. *ELASTIC BLOCK STORE | Snapshots | Create Snapshot* button in the top pane.
5. Fill in the form. I entered the name as *baseline* with description *apps installed, no content yet*.
6. Click the *Create* button. The AWS apps behind the scenes create the backup in a couple minutes.

Restore a backup snapshot by launching a new EC2 machine

One method of restoring is to recreate the entire EC2 machine. This restore procedure is similar to my initial [EC2 machine creation](#).

1. Make an image from the snapshot. I called mine *img01*. *EC2 tab | ELASTIC BLOCK STORE | Snapshots | baseline row | Create Image*.
2. View the image. *IMAGES | AMIs | Refresh*.
3. Make an instance from the image. *INSTANCES | Instances | Launch Instance | Create a New Instance*. The *My AMIs* tab shows the name *123456789012/img01*. That long number is my AWS account number.

Restore a backup snapshot by mounting on an existing EC2 machine

Another way is to make a volume from the snapshot to mount the volume on an existing EC2 machine and copy files from it. I have done this when my new EC2 machine's kernel refused to mount an old snapshot's volume.

The first thing to do is a little research. You can't mount a volume from one zone on a machine in another zone.

Check the EC2 machine.

1. Pick an instance. *EC2 tab | INSTANCES | Instances*.
2. Make a note of the EC2 machine's availability zone.

Create a volume.

1. Find the snapshot you want to copy files from and tick the box. *ELASTIC BLOCK STORE | Snapshots*
2. Click the *Create Volume* button and fill in the fields.
 - The *Size* must be bigger than the snapshot size (free micro-instances get an 8GB volume).
 - The *Availability Zone* must be the same as the EC2 machine's.
 - The *Snapshot* is already selected, more or less like *snap12345678 – my description*.
3. Click the *Yes, Create* button. A new line appears in the *Volumes* table. *ELASTIC BLOCK STORE | Volumes*

Attach the volume.

1. Click the *Attach Volume* button and fill in the fields.
2. The *Volume* value is already there.

3. Pick your machine name *i-12345678 (running)* from the drop-down list of Instances.
4. The Devices field shows the first available device name, like */dev/sdf*. Does anyone bother changing this value?
5. Click the Yes, Create button. A new device magically appears on the EC2 machine.
6. Close the AWS console.

Copy files.

1. [Open a CLI](#).
2. Mount the new volume.

```
sudo mount /dev/sdf /mnt/
```
3. Copy any files you want to the old volume.

```
cp /mnt/home/ec2-user/my-old-file.txt /home/ec2-user/
```

Clean up.

1. Unmount the new volume.

```
sudo umount /mnt/
```
2. Close the CLI.

Share Article



Nick Hardiman

Nick Hardiman builds and maintains the infrastructure required to run Internet services such as the web. Nick deals with the lower layers of the Internet - the machines, networks, operating systems and applications. Nick's job stops there and he hands over to the designers and developers who build the top layer that customers use.

Source: <https://www.techrepublic.com/blog/the-enterprise-cloud/backing-up-and-restoring-snapshots-on-amazon-ec2-machines/>