File System Snapshot with bsnapshot

Bacula Systems Documentation

Contents

Contents

Note

You can download this article as a PDF

Since Bacula Enterprise version 8.2, it is possible for Clients to perform automatic File System Snapshots backup. It is also possible to manage Snapshots from Bacula's bconsole tool through a unique interface.

The plugin provides file-level restore at the snapshot state.

The following Snapshot backends are supported:

- BTRFS
- ZFS
- LVM
- Quobyte

Note

Some restrictions described *here* apply to the LVM backend.

By default, Snapshots are mounted (or directly available) under .snapshots directory on the root filesystem (On ZFS, the default is .zfs/snapshots).

The Snapshot backend program is called bsnapshot and is available in the bacula-enterprise-snapshot package. In order to use the Snapshot Management feature, the package must be installed on the Client.

The bsnapshot program can be configured using `/opt/bacula/etc/bsnapshot.conf` file. The following parameters can be adjusted in the configuration file:

- `sudo=<yes|no>` Use sudo to run commands
- `disabled=<yes|no>` Disable snapshot support

- `retry`= Configure the number of retries for some operations
- `snapshot_dir`= Use a custom name for the Snapshot directory. (.SNAPSHOT, .snapdir, etc.)
- `lvm_snapshot_size`= Specify a custom snapshot size for a given LVM volume
- `mountopts`= Specify a custom mount option for a given device (available since 10.0.4)
- `trace`= Specify a trace file
- `debug`= Specify a debug level

There is an example as follows:

```
# cat /opt/bacula/etc/bsnapshot.conf
trace=/tmp/snap.log
debug=10
lvm_snapshot_size=/dev/ubuntu-vg/root:5%
mountopts=nouuid
mountopts=/dev/ubuntu-vg/root:nouuid,nosuid
```

1 Application Quiescing

When using Snapshots, it is very important to quiesce applications that are running on the system. The simplest way to quiesce an application is to stop it. Usually, taking the Snapshot is very fast, and the downtime is only about a couple of seconds.

If downtime is not possible and/or the application provides a way to quiesce, directives RunBeforeJob and RunAfterJob can be used.

2 Director Directives

The use of the Snapshot Plugin on the FileDaemon is determined by the Enable Snapshot Fileset directive. The default is no.

```
Fileset {
  Name = LinuxHome
  Enable Snapshot = yes
  Include {
    Options = { Compression = LZO }
      File = /home
  }
}
```

By default, Snapshots are deleted from the Client at the end of the backup. To keep Snapshots on the Client and record them in the Catalog for a determined period, it is possible to use the Snapshot Retention directive in the Client or in the Job resource. The default value is 0 seconds. If, for a given Job, both Client and Job Snapshot Retention directives are set, the Job directive will be used.

```
Client {
  Name = linux1
  ...
  Snapshot Retention = 5 days
}
```

To automatically prune Snapshots, it is possible to use the following RunScript command:

```
Job {
  ...
  Client = linux1
  ...
  RunScript {
    RunsOnClient = no
    Console = "prune snapshot client=%c yes"
    RunsAfter = yes
  }
}
```

In RunScripts, the AfterSnapshot keyword for the RunsWhen directive will allow a command to be run just after the Snapshot creation. AfterSnapshot is a synonym for the AfterVSS keyword.

```
Job {
  ...
  RunScript {
    Command = "/etc/init.d/mysql start"
    RunsWhen = AfterSnapshot
    RunsOnClient = yes
  }
  RunScript {
    Command = "/etc/init.d/mysql stop"
    RunsWhen = Before
    RunsOnClient = yes
  }
}
```

3 Job Output Information

Information about Snapshots are displayed in the Job output. The list of all devices used by the Snapshot Engine is displayed, and the Job summary indicates if Snapshots were available.

```
JobId 3:    Create Snapshot of /home/build
JobId 3:    Create Snapshot of /home/build/subvol
JobId 3:    Delete snapshot of /home/build
JobId 3:    Delete snapshot of /home/build/subvol
...
JobId 3: Bacula 127.0.0.1-dir 7.2.0 (23Jul15):
  Build OS:               x86_64-unknown-linux-gnu archlinux
  JobId:                  3
  Job:                    Incremental.2015-02-24_11.20.27_08
  Backup Level:           Full
...
  Snapshot/VSS:           yes
...
  Termination:            Backup OK
```

4 Snapshot Bconsole Commands

The snapshot command will display by default the following menu:

```
*snapshot
Snapshot choice:
     1: List snapshots in Catalog
     2: List snapshots on Client
     3: Prune snapshots
     4: Delete snapshot
     5: Update snapshot parameters
     6: Update catalog with Client snapshots
     7: Done
Select action to perform on Snapshot Engine (1-7):
```

The snapshot command can also have the following parameters:

```
[client=<client-name> | job=<job-name> | jobid=<jobid>]
 [delete | list | listclient | prune | sync | update]
```

It is also possible to use traditional list, llist, update, prune or delete commands on Snapshots.

```
*llist snapshot jobid=5
 snapshotid: 1
       name: NightlySave.2015-02-24_12.01.00_04
 createdate: 2015-02-24 12:01:03
     client: 127.0.0.1-fd
    fileset: Full Set
      jobid: 5
     volume: /home/.snapshots/NightlySave.2015-02-24_12.01.00_04
     device: /home/btrfs
       type: btrfs
  retention: 30
    comment:

* snapshot listclient
Automatically selected Client: 127.0.0.1-fd
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
Snapshot      NightlySave.2015-02-24_12.01.00_04:
  Volume:     /home/.snapshots/NightlySave.2015-02-24_12.01.00_04
  Device:     /home
  CreateDate: 2015-02-24 12:01:03
  Type:       btrfs
  Status:     OK
  Error:
```

In order to update the catalog with Client snapshots option (or snapshot sync), the Director contacts the FileDaemon, lists snapshots of the system and creates catalog records of the Snapshots.

```
*snapshot sync
Automatically selected Client: 127.0.0.1-fd
Connecting to Client 127.0.0.1-fd at 127.0.0.1:8102
Snapshot      NightlySave.2015-02-24_12.35.47_06:
  Volume:     /home/.snapshots/NightlySave.2015-02-24_12.35.47_06
```

(continues on next page)

(continued from previous page)

```
  Device:     /home
  CreateDate: 2015-02-24 12:35:47
  Type:       btrfs
  Status:     OK
  Error:
Snapshot added in Catalog

*llist snapshot
 snapshotid: 13
       name: NightlySave.2015-02-24_12.35.47_06
 createdate: 2015-02-24 12:35:47
     client: 127.0.0.1-fd
    fileset:
      jobid: 0
     volume: /home/.snapshots/NightlySave.2015-02-24_12.35.47_06
     device: /home
       type: btrfs
  retention: 0
    comment:
```

5 LVM Backend Restrictions

LVM Snapshots are quite primitive compared to ZFS, BTRFS, NetApp and other systems. For example, it is not possible to use Snapshots if the Volume Group (VG) is full. The administrator must keep some free space in the VG to create Snapshots.

The amount of free space required depends on the activity of the Logical Volume (LV). bsnapshot uses 10% of the LV by default. This number can be configured per LV in the bsnapshot.conf file.

```
[root@system1]# vgdisplay
  --- Volume group ---
  VG Name               vg_ssd
  System ID
  Format                lvm2
...
  VG Size               29,81 GiB
  PE Size               4,00 MiB
  Total PE              7632
  Alloc PE / Size       125 / 500,00 MiB
  Free  PE / Size       7507 / 29,32 GiB
...
```

It is also not advisable to leave snapshots on the LVM backend. Having multiple snapshots of the same LV on LVM will slow down the system.

Only Ext4, XFS and EXT3 filesystems are supported with the Snapshot LVM backend.

6 Debug Options

To get low level information about the bsnapshot, the debug tag “snapshot” should be used in the `setdebug` command.

```
*setdebug level=10 tags=snapshot client
*setdebug level=10 tags=snapshot dir
```