

simplevm-backup

Installation, Testing, and Usage Guide

Version 0.1.5 · Technology Preview

© Copyright 2026 Thinware, Inc. All Rights Reserved.

1. Overview

simplevm-backup is a command-line utility that creates a point-in-time backup of a single virtual machine on a SimpleVM host. It combines a libvirt external snapshot with a thin, compressed copy of each virtual disk, so the virtual machine **continues running throughout the backup** — no downtime or restart is required.

Each backup is written to a self-contained, timestamped directory containing the virtual machine's disks, firmware variables (NVRAM), and configuration.

This guide describes how to install the utility on a SimpleVM server, validate it, and run backups.

Technology Preview Version 0.1.5 is a Technology Preview release provided for evaluation. It is feature-complete for full-virtual-machine backup but has not yet completed full production qualification. Before relying on it, confirm its behavior in your environment with a dry run (Section 4.1) followed by a backup of a non-critical virtual machine. Because backups run online, they do not stop or restart your virtual machines.

2. Prerequisites

Before you begin, ensure the following:

- **Location** — These steps are performed on the SimpleVM host itself (the hypervisor), not inside a guest virtual machine.
- **Privileges** — Administrative (root) access to the SimpleVM server. The utility interacts with libvirt and reads virtual-machine storage, so all backup operations must run as root (via sudo).
- **Platform tools** — The virsh and qemu-img commands. These are present by default on every SimpleVM server; no additional packages are required.

- **Backup destination** — A directory with sufficient free space to hold your backups, ideally on storage separate from the virtual machines. Disks are compressed, but plan capacity for the number of backups you intend to retain.

NOTE To list the virtual machines on the host and confirm a machine's exact name, run:

```
sudo virsh list --all
```

3. Installing simplevm-backup

Procedure

1. Create the application directory.

1. Log in to the SimpleVM web console and open **File browser**.
2. Browse to /opt.
3. Click the actions menu (... , top right) and select **Create directory**:
 - **Directory name:** simplevm
 - **Directory owner:** root:root
4. Click **Create**.

2. Upload the executable.

1. In **File browser**, browse to /opt/simplevm.
2. Click **Upload**, select the simplevm-backup file, and click **Open**.

3. Make the application executable.

1. In the SimpleVM web console, open **Terminal**.
2. Run the following commands:

```
cd /opt/simplevm
```

```
sudo chmod +x simplevm-backup
```

Verification

Confirm that the utility runs and displays its help text:

```
./simplevm-backup --help
```

The command prints a usage summary, including all available options and exit codes. (Displaying help does not require administrative privileges; backup operations do — see Section 4.)

TIP To run the utility from any directory without typing its full path, create a symbolic link on the system path:

```
sudo ln -s /opt/simplevm/simplevm-backup /usr/local/bin/simplevm-backup
```

4. Running a backup

The utility takes three required options and several optional flags:

```
sudo ./simplevm-backup -n <vm-name> -r <backup-root> -k <keep-count> [options]
```

Short	Long	Description
-n	--vm-name	Name of the virtual machine to back up. (Required)
-r	--backup-root	Directory under which backups are created. (Required)
-k	--keep-count	Number of backups to retain; older backups are pruned after a successful run. (Required)
-q	--quiesce	Request an application-consistent backup using the QEMU guest agent.
-d	--dry-run	Validate and report the planned actions without making any changes.
-v	--verbose	Produce detailed diagnostic output.
-h	--help	Display the help text and exit.

IMPORTANT Run all backup operations as root (using sudo), including dry runs. Without administrative privileges, the utility cannot reach the host's virtual machines and will report that the machine cannot be found.

The examples below are run from /opt/simplevm; alternatively, use the full path /opt/simplevm/simplevm-backup (or the symbolic link from Section 3).

4.1. Validate with a dry run (recommended first)

A dry run performs all validation and reports exactly what *would* happen — the disks that would be copied, the destination directory, and the consistency level that would be achieved — without modifying anything. Always start here, particularly on a production host.

```
sudo ./simplevm-backup -n web01 -r /backups -k 7 --dry-run
```

4.2. Perform a backup

After reviewing the dry run, create the backup destination (if it does not already exist) and run the backup:

```
sudo mkdir -p /backups
```

```
sudo ./simplevm-backup -n web01 -r /backups -k 7
```

This creates a crash-consistent backup while web01 keeps running, then prunes all but the seven most recent backups of that machine.

4.3. Application-consistent backups (optional)

For transactional workloads such as databases, add `--quiesce` to briefly freeze the guest filesystem during the snapshot, producing an application-consistent backup:

```
sudo ./simplevm-backup -n db01 -r /backups -k 14 --quiesce
```

NOTE `--quiesce` requires the QEMU guest agent (`qemu-guest-agent`) to be installed and running inside the guest. If the agent is not responsive, the backup stops with an error rather than silently falling back to a crash-consistent backup. Omit `--quiesce` for a crash-consistent backup, or when backing up a powered-off machine (which is already fully consistent).

Backup contents

Each run produces a timestamped directory:

```
<backup-root>/<vm-name>/<YYYY-MM-DD_HH-MM-SS>/
```

containing:

- `<vm-name>-<disk>.qcow2` — a compressed copy of each virtual disk (for example, `web01-vda.qcow2`)
- `<vm-name>-nvram.fd` — firmware variables (UEFI virtual machines only)
- `<vm-name>.xml` — the virtual-machine configuration, prepared for restore

- `<vm-name>.xml.backup` — the original configuration, verbatim
 - `<vm-name>-<timestamp>.log` — a detailed log of the run
-

5. Verifying results

Exit codes

`simplevm-backup` returns a standard exit code, suitable for scripting and scheduling:

Code Meaning

- | | |
|---|--|
| 0 | Success, or a completed dry run. |
| 1 | Validation or backup failure. No backup was produced, or a partial backup was rolled back. |
| 2 | The backup completed, but the temporary snapshot could not be removed. The virtual machine may require manual attention (see Section 6). |

Logs

- **Per-backup log** — the `<vm-name>-<timestamp>.log` file inside each backup directory records that run in detail.
 - **Application log** — `/var/log/simplevm/simplevm-backup.log` records one summary line per run (result, consistency level, and duration), providing an at-a-glance history suitable for monitoring.
-

6. Troubleshooting

The next run reports that a snapshot already exists. If a backup is interrupted (for example, the host reboots mid-run), `simplevm-backup` intentionally leaves its temporary snapshot in place as evidence and refuses to start again until it is cleared. Remove the snapshot to merge it back into the base disk:

```
sudo virsh snapshot-delete <vm-name> simplevm-backup-temp
```

You may then delete the corresponding incomplete backup directory.

"VM not found, or libvirt is unreachable." Confirm the machine name with `sudo virsh list -all`, and ensure the command is run with `sudo`.

"Backup root is not writable" or "does not exist." Create the destination directory and verify its permissions:

```
sudo mkdir -p <backup-root>
```

The utility reports that it has expired. This Technology Preview build is time-limited. Contact Thinware for an updated release.

7. Scheduling backups (optional)

Once validated, simplevm-backup can be scheduled with cron. For example, to back up web01 every night at 1:00 AM while retaining seven days of backups, add the following line to root's crontab (sudo crontab -e):

```
0 1 * * * /opt/simplevm/simplevm-backup -n web01 -r /backups -k 7
```

Monitor outcomes through the application log and the command's exit code.

8. Support and feedback

This is a Technology Preview release, and your testing helps shape the final product. Please report issues, unexpected behavior, or suggestions to Thinware support at support@thinware.net.