

IT and DevOps

- [OpenZFS Issue #15526 Patch and Mitigation for Older Versions - SUBSTANTIAL REVISION](#)
- [Docker and Docker Compose v2 in Fedora CoreOS](#)
- [Importing VMs from TrueNAS Core \(Bhyve\) to Proxmox](#)
- [Image Credit - Book Cover Art](#)

OpenZFS Issue #15526

Patch and Mitigation for Older Versions -

SUBSTANTIAL REVISION

SUBSTANTIAL REVISION: Issue #15526 has been patched in ZFS versions 2.2.2 and 2.1.14, according to open source reporting. The mitigation below only applies to older versions of ZFS.

Summary

A silent data corruption bug exists in ZFS versions 2.1.4 up to 2.1.13, 2.2.0 and 2.2.1, per [this GitHub issue](#). Versions 2.2.2 and 2.1.14 have patched the issue, according to [open source reporting](#).

A ZFS scrub will not identify any data corrupted by this bug. The only high-assurance method to check if any files were corrupted is to compare files within ZFS to their copies stored outside of ZFS.

If you are still using an older version of ZFS that is impacted by this issue, you may significantly lower the chance that this issue impacts your file system by setting the ZFS parameter `zfs_dmu_offset_next_sync` to 0. Note that this does not prevent the issue from occurring, per [this GitHub comment](#), but it does lower the chance that silent data corruption occurs.

Details

Mitigation

Linux

Runtime

To apply the mitigation in runtime, run the following command as the root user:

```
echo 0 > /sys/module/zfs/parameters/zfs_dmu_offset_next_sync
```

Permanent

To apply the mitigation permanently, create a file in `/etc/modprobe.d/` such as:

```
/etc/modprobe.d/mitigation.conf
```

Containing the following:

```
options zfs zfs_dmu_offset_next_sync=0
```

FreeBSD

Runtime

To apply the mitigation in runtime, run the following command as the root user:

```
sysctl -w vfs.zfs.dmu_offset_next_sync=0
```

Permanent

To apply the mitigation permanently, append the following line to `/etc/sysctl.conf`:

```
vfs.zfs.dmu_offset_next_sync=0
```

Reproducing the Bug

Linux

To reproduce the bug in Linux, use the script below (copy pasted from the following [gist](#)):

```
#!/bin/bash
#
# Run this script multiple times in parallel inside your pool's mount
# to reproduce https://github.com/openzfs/zfs/issues/15526. Like:
#
# ./reproducer.sh & ./reproducer.sh & ./reproducer.sh & ./reproducer.sh & wait
#

# if [ $(cat /sys/module/zfs/parameters/zfs_bclone_enabled) != "1" ] ; then
#   echo "please set /sys/module/zfs/parameters/zfs_bclone_enabled = 1"
#   exit
# fi

prefix="reproducer_${BASHPID}_"
dd if=/dev/urandom of=${prefix}0 bs=1M count=1 status=none

echo "writing files"
end=1000
h=0
for i in `seq 1 2 $end` ; do
  let "j=i+1"
  cp ${prefix}$h ${prefix}$i
  cp --reflink=never ${prefix}$i ${prefix}$j
  let "h++"
done

echo "checking files"
for i in `seq 1 $end` ; do
  diff ${prefix}0 ${prefix}$i
done
```

FreeBSD

To reproduce the bug in FreeBSD, use the script below (copy pasted from the following [post](#)):

```
#!/bin/bash
#
# Run this script multiple times in parallel inside your pool's mount
# to reproduce https://github.com/openzfs/zfs/issues/15526. Like:
```

```
#
# ./reproducer.sh & ./reproducer.sh & ./reproducer.sh & ./reproducer.sh & wait
#

# if [ $(cat /sys/module/zfs/parameters/zfs_bclone_enabled) != "1" ] ; then
#     echo "please set /sys/module/zfs/parameters/zfs_bclone_enabled = 1"
#     exit
# fi

prefix="reproducer_${BASHPID}_"
dd if=/dev/urandom of=${prefix}0 bs=1M count=1 status=none

echo "writing files"
end=1000
h=0
for i in `seq 1 2 $end` ; do
    let "j=$i+1"
    cp ${prefix}$h ${prefix}$i
    cp ${prefix}$i ${prefix}$j
    let "h++"
done

echo "checking files"
for i in `seq 1 $end` ; do
    diff ${prefix}0 ${prefix}$i
done
```

Commentary

I was unable to reproduce this issue in TrueNAS Core 13.0-U5.3 (FreeBSD) but I was able to reproduce it in Proxmox 8.0.4 (Debian).

Source Description Block

Multiple sources:

Issue tracking in OpenZFS: <https://github.com/openzfs/zfs/issues/15526>

Mitigation: <https://github.com/openzfs/zfs/issues/15526#issuecomment-1823737998>

Linux reproducer script:

<https://gist.github.com/tonyhutter/d69f305508ae3b7ff6e9263b22031a84>

FreeBSD reproducer script: <https://www.truenas.com/community/threads/truenas-13-0-u6-is-now-available.114337/page-3>

TrueNAS Core (FreeBSD) issue forum thread:

<https://www.truenas.com/community/threads/silent-corruption-with-openzfs-ongoing-discussion-and-testing.114390/>

Documentation on `dmu_offset_next_sync`: <https://openzfs.github.io/openzfs-docs/Performance%20and%20Tuning/Module%20Parameters.html#zfs-dmu-offset-next-sync>

Data corruption bug occurs even with `zfs_dmu_offset_next_sync` set to 0:

<https://github.com/openzfs/zfs/issues/15526#issuecomment-1826348986>

Reddit thread on the bug:

https://old.reddit.com/r/DataHoarder/comments/1821mpr/heads_up_for_a_data_corruption_bug_in_zfs_few/

Reddit thread on the bug:

https://old.reddit.com/r/zfs/comments/1826lgs/psa_its_not_block_cloning_its_a_data_corruption/

Issue fixed in versions 2.2.2 and 2.1.14: <https://www.phoronix.com/news/OpenZFS-2.2.2-Released>

Licensing

This page (**not including the code snippets**) is licensed under a [Creative Commons Universal \(CC0 1.0\) Public Domain Dedication](#). For code snippet licensing, please contact the original authors.

Docker and Docker Compose v2 in Fedora CoreOS

Summary

If you prefer to use Docker over Podman in Fedora CoreOS, use the Butane file below to add the latest version of Docker and Docker Compose v2 to your system.

Details

Butane

```
variant: fcos
version: 1.4.0
passwd:
  users:
    - name: core
      ssh_authorized_keys:
        - ssh-[Your SSH key]
storage:
  files:
    - path: /etc/yum.repos.d/docker-ce.repo
      overwrite: true
      contents:
        inline: |
          [docker-ce-stable]
          name=Docker CE Stable - $basearch
          baseurl=https://download.docker.com/linux/fedora/$releasever/$basearch/stable
          enabled=1
          gpgcheck=1
```

gpgkey=https://download.docker.com/linux/fedora/gpg

systemd:

units:

Removing unofficial copies of docker and related packages

- name: rpm-ostree-uninstall.service

enabled: true

contents: |

[Unit]

Description=Docker rpm-ostree install

Wants=network-online.target

After=network-online.target

We run before `zincati.service` to avoid conflicting rpm-ostree

transactions.

Before=zincati.service

ConditionPathExists=!/var/lib/%N.stamp

[Service]

Type=oneshot

RemainAfterExit=yes

ExecStart=/usr/bin/rpm-ostree override remove docker containerd runc

ExecStart=/bin/touch /var/lib/%N.stamp

[Install]

WantedBy=multi-user.target

Installing Docker as a layered package with rpm-ostree

- name: rpm-ostree-install.service

enabled: true

contents: |

[Unit]

Description=Docker rpm-ostree install

Wants=network-online.target

Requires=rpm-ostree-uninstall.service

After=rpm-ostree-uninstall.service

We run before `zincati.service` to avoid conflicting rpm-ostree

transactions.

Before=zincati.service

ConditionPathExists=!/var/lib/%N.stamp

[Service]


```
Type=oneshot
RemainAfterExit=yes
ExecStart=/usr/bin/rpm-ostree install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-
compose-plugin
ExecStart=/bin/touch /var/lib/%N.stamp

[Install]
WantedBy=multi-user.target
```

Butane - Explanation

On line 7, add your SSH public key to be able to sign into your Fedora CoreOS machine. We add the Docker repository as a file. Then, we use some systemd trickery to remove docker, runc and containerd. These are installed by default in Fedora CoreOS, but conflict with the up-to-date versions of Docker, so we remove them. The next service waits for the uninstall service to complete, and installs docker per the Fedora installation guide [here](#).

Your Fedora CoreOS system will reboot in 10 minutes after running these systemd services. It's unfortunately impossible to apply software removals live, so a restart is required. If you wish to restart sooner, you can run `systemctl reboot` manually.

Why?

Podman doesn't have the equivalent of Docker Compose. Per the [suggestion](#) of the Podman development team, we can simply use Docker Compose with a Podman backend. There needs to be some trickery done to support building images with a Podman backend, which can be seen [here](#).

Overall, I found Podman to be more trouble than it's worth. As I worked with Podman for nearly a year, I ran into constant incompatibilities and oddities that had me searching for workarounds for things that should *just work*. Simply running the latest version of Docker and Docker Compose not only meets my needs, but is stable—I have yet to have any breaking changes due to automatic updates with Docker and Docker Compose v2.

Licensing

This page is licensed under a [Creative Commons Universal \(CC0 1.0\) Public Domain Dedication](#).

Importing VMs from TrueNAS Core (Bhyve) to Proxmox

Summary

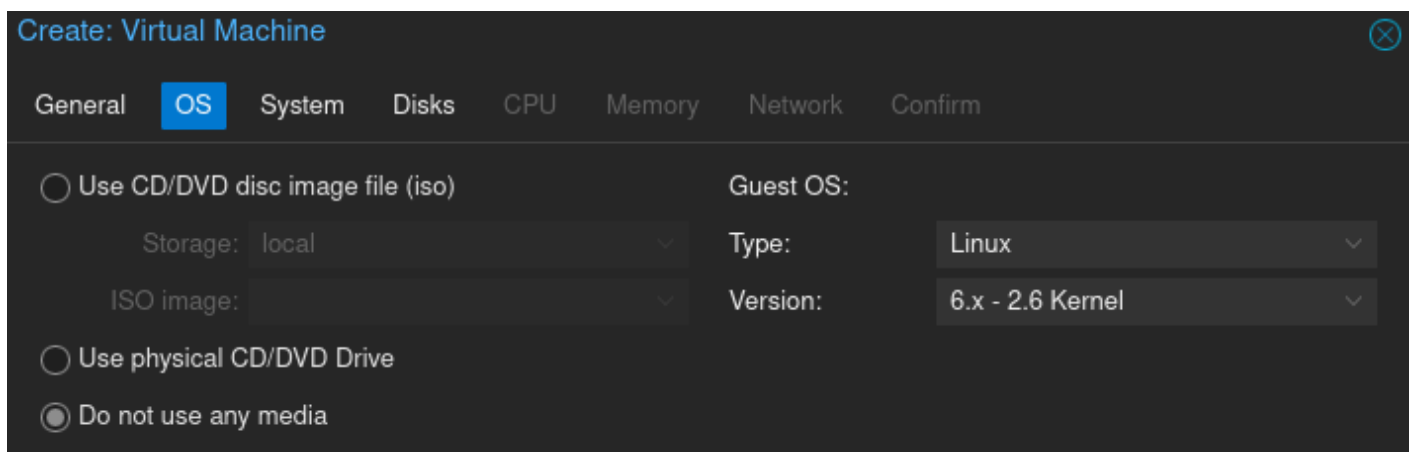
This page explains the process of importing VMs from TrueNAS Core, which uses FreeBSD's [Bhyve](#) for virtualization, to Proxmox.

Details

Proxmox

In Proxmox, create a new VM and note its VM number. When creating the VM, follow these guidelines:

In the OS section, select "Do not use any media"



The screenshot shows the 'Create: Virtual Machine' window in Proxmox, with the 'OS' tab selected. The window has a dark theme and a close button in the top right corner. The 'General' tab is also visible. The 'OS' tab contains the following options:

- ☐ Use CD/DVD disc image file (iso)
 - Storage: local
 - ISO image:
- ☐ Use physical CD/DVD Drive
- ☒ Do not use any media

On the right side, the 'Guest OS' section is visible with the following settings:

- Guest OS: Type: Linux
- Version: 6.x - 2.6 Kernel

In the System section, select "OVMF (UEFI)" for BIOS. Also select EFI Storage to be the same dataset as where you would like your VM's disk to be. We chose the default local-zfs dataset, but you may choose any other dataset, such as an encrypted dataset if you want your VMs to be encrypted.

Create: Virtual Machine

General

OS

System

Disks

CPU

Memory

Network

Confirm

Graphic card:

Default

SCSI Controller:

VirtIO SCSI single

Machine:

Default (i440fx)

Qemu Agent:

☐

Firmware

BIOS:

OVMF (UEFI)

Add TPM:

☐

Add EFI Disk:

☒

EFI Storage:

local-zfs

Format:

Raw disk image (raw)

Pre-Enroll keys:

☒

In the disk section, remove the default disk and do not set a disk.

Create: Virtual Machine

General

OS

System

Disks

CPU

No Disks

Continue with the rest of the sections per your own personal requirements.

TrueNAS

Shutdown the VM in TrueNAS and make a snapshot of the VM dataset in TrueNAS. Login to SSH in TrueNAS as the root user and run the following command to send the dataset using SSH to Proxmox:

```
zfs send [VM_Dataset]@[snapshot_name] | ssh root@proxmox 'zfs receive rpool/[any dataset here]/vm-[num]-disk-1'
```

If you use DHCP in your network and you would like your IP address for the VM to be identical after migration, press the devices button in your virtual machine menu:

Virtual Machines

COLUMNS

ADD

Name	State	Autostart
<div> <div>Virtual CPUs: 1</div> <div>Cores: 2</div> <div>Threads: 1</div> <div>Memory Size: 8.00 GiB</div> <div>Boot Loader Type: UEFI</div> <div>System Clock: LOCAL</div> <div>VNC Port: 31449</div> <div>Com Port: /dev/nmdm2B</div> <div>Description:</div> <div>Shutdown Timeout: 90 seconds</div> </div> <div> <div>▶ START</div> <div>✎ EDIT</div> <div>🗑 DELETE</div> <div>👤 DEVICES</div> <div>📄 CLONE</div> </div>		

Then, press the three dots over the "NIC" device and press Edit.

Device ID	Device	Order
14	NIC	1002
15	DISK	1001
16	VNC	1002

Edit

Delete

Change Device Order

Details

1 - 3 of 3

A new menu should show up which displays the MAC address. Copy this MAC address. In Proxmox, you may edit your Network Device and paste the MAC address there.

Back to Proxmox

Back in Proxmox, login to the root shell and run the `qm rescan` command. Then, go into your VM's hardware menu. The disk should show up as an unattached device. You may now attach it.

Congratulations, you have successfully migrated a virtual machine from TrueNAS Core to Proxmox!

Source Description Block

Multiple Sources:

<https://forum.proxmox.com/threads/adding-existing-disk-from-storage-to-vm.108645/>

https://www.youtube.com/watch?v=yKZ_JJaQHDk

Licensing

This page is licensed under a [Creative Commons Universal \(CC0 1.0\) Public Domain Dedication](#).

Image Credit - Book Cover Art

Photo by Nadin Sh from [Pexels](#).

The photo represents my feelings about DevOps and related tooling.