CSCvd42177 is one of the common causes for copy run to fail on the N6k and N5k switches.

<https://bst.cloudapps.cisco.com/bugsearch/bug/CSCvd42177>

**Symptom: -**Copy run start fails  
# copy running-config startup-config  
[########################################] 100%  
Jun 30 14:33:54 %KERN-2-SYSTEM\_MSG: EXT3-fs error (device sda5): ext3\_readdir: directory #2 contains a hole at offset 0 - kernel  
Configuration update aborted: request was aborted  
  
Following messages will also be seen in the log

           2017 Feb 21 12:50:16 nexus\_lvb Feb 21 12:50:16 %KERN-3-SYSTEM\_MSG: [8994819.472641] Buffer I/O error on device sda3, logical block 0 - kernel

          2017 Feb 21 12:51:35 nexus\_lvb Feb 21 12:51:35 %KERN-2-SYSTEM\_MSG: [8994897.997856] EXT3-fs error (device sda3): ext3\_find\_entry: reading directory #2 offset 0 - kernel

          2017 Feb 21 12:51:35 nexus\_lvb Feb 21 12:51:35 %KERN-3-SYSTEM\_MSG: [8994897.997872] Buffer I/O error on device sda3, logical block 0 - kernel

          2017 Feb 21 12:51:35 nexus\_lvb Feb 21 12:51:35 %KERN-2-SYSTEM\_MSG: [8994898.126319] EXT3-fs error (device sda3): ext3\_find\_entry: reading directory #2 offset 0 - kernel

          2017 Feb 21 12:51:35 nexus\_lvb Feb 21 12:51:35 %KERN-3-SYSTEM\_MSG: [8994898.126336] Buffer I/O error on device sda3, logical block 0 - kernel

          2017 Feb 21 12:51:36 nexus\_lvb Feb 21 12:51:35 %KERN-2-SYSTEM\_MSG: [8994898.129387] EXT3-fs error (device sda4): ext3\_find\_entry: reading directory #2 offset 0 - kernel

          2017 Feb 21 12:51:36 nexus\_lvb Feb 21 12:51:35 %KERN-2-SYSTEM\_MSG: [8994898.132190] EXT3-fs error (device sda4): ext3\_find\_entry: reading directory #2 offset 0 - kernel

Symptom: **Bootflash** also appears to operating in a read only mode. Files cannot be written or output redirected to the bootflash.

 According to the workaround above, Reload the box after the configuration is backed up to a remote server/location.

Option1: reload the box, by saving configuration to some external TFTP or NMS server if not customer can lose the config.

Option2: Use the steps to mount/unmount devices to restore the write functionality on the file system and reload when possible.

**Permanent fix:** Upgrade to latest version 7.3(5)N1(1), once Cisco provides Proactive Software Risk Analysis.

**Steps to Validate defect: (\*\*We will need the dplugin installed for this. Refer to instructions at the bottom of the document)**

We can verify this using the command fdisk -l from linux prompt:

Linux(debug)# fdisk -l

Disk /dev/sdb: 8053 MB, 8053063680 bytes

16 heads, 63 sectors/track, 15603 cylinders

Units = cylinders of 1008 \* 512 = 516096 bytes

Disk identifier: 0x900f6a6d

Device Boot Start End Blocks Id System

/dev/sdb1 1 32 16096+ 83 Linux

/dev/sdb2 33 498 234864 5 Extended

/dev/sdb3 733 15603 7494984 83 Linux

/dev/sdb4 499 732 117936 83 Linux

/dev/sdb5 33 266 117904+ 83 Linux

/dev/sdb6 267 498 116896+ 83 Linux

Partition table entries are not in disk order

Linux(debug)#

If the output shows sdb instead of sda as highlighted above, that validates that the issue is due to the defect.

After this glitch, File-system goes to read-only mode. We cannot list the contents in bootflash directory & we can't save running configuration ('copy r s' command fails).

**Workaround:**

Option 1: Reload will resolve this issue.

A reload is a cleaner way to recover but a workaround exists in debug plugin mode.

However, there is a workaround to fix this without the reload also & the steps are mentioned below:

Option 2:

(All the below commands should be issued from linux debug prompt)

Step 1: Unmount all the old devices/partitions by issuing following commands

umount /dev/sda5 #/mnt/cfg/0

umount /dev/sda6 #/mnt/cfg/1

umount /dev/sda3 #/bootflash

umount -l /dev/sda4 #/mnt/pss

Step 2: mount it to new device sdb

mount -t ext3 -o rw,nodev,noatime /dev/sdb4 /mnt/pss

mount -t ext3 -o rw,nodev,noatime /dev/sdb5 /mnt/cfg/0

mount -t ext3 -o rw,nodev,noatime /dev/sdb6 /mnt/cfg/1

mount -t ext3 -o rw,nodev,noatime /dev/sdb3 /bootflash

Step 3: create new symlinks to match to /etc/fstab file

rm /dev/hd-cfg0

rm /dev/hd-cfg1

rm /dev/hd-bootflash

rm /dev/hd-pss

Step 4: Creating new symlinks to sdb...

ln -s /dev/sdb5 /dev/hd-cfg0

ln -s /dev/sdb6 /dev/hd-cfg1

ln -s /dev/sdb3 /dev/hd-bootflash

ln -s /dev/sdb4 /dev/hd-pss

After doing above commands, the problem is resolved but every-time 'copy run start' command is issued, following syslog message is seen. The message seems to be harmless and will be gone after next reload.

esc-6001-aa9# copy run startup-config

[########################################] 100%

Copy complete, now saving to disk (please wait)...

esc-6001-aa9# 2017 Mar 6 08:00:14 esc-6001-aa9 %PSS-0-PSS\_WRITE\_LOG\_FAILURE: res\_mgr: failed to write log: Read-only file system

If all the above fails or if the validation outputs don’t match the defect, please contact TAC for further Assistance.

**\*\*\*Dplugin is needed to get into the Linux mode and there is a seperate dplugin for every release version and hence defect calls out to reach out to TAC for it. If you can share the list of NXOS releases currently in production we can share the dplugins for those releases\*\*\***

**Procedure to install the dplugin:**

1. Copy the plugin to your device. You can use ftp, tftp, scp, sftp servers or USB flash drive.
2. You need to copy the plugin to either the bootflash of the device or if not possible (because of the issue), you can use the “volatile” memory.

The commands to do so will be something like this:

**copy tftp: bootflash:**

**copy tftp: volatile:**

- when you run those commands, you need to follow the instructions after that and chose the filename, hostname of the server (IP), vrf, username, password.

1. After you have the plugin on the device’s bootflash or volatile memory, you need to run it.

The commands to do so will be:

**load bootflash:nuova-or-dplug-mzg.7.1.4.N1.1.bin (plugin name)**

OR

**load volatile:nuova-or-dplug-mzg.7.1.4.N1.1.bin**

1. After you’re in the linux kernel, you will need to run the following command: fdisk  -l

Linux(debug)# **fdisk  -l**