



Handling Core Dump

- [Feature Summary and Revision History, on page 1](#)
- [Feature Changes, on page 2](#)
- [Command Changes, on page 3](#)
- [Performance Indicator Changes, on page 3](#)

Feature Summary and Revision History

Summary Data

Applicable Product(s) or Functional Area	All
Applicable Platform(s)	<ul style="list-style-type: none">• ASR 5500• VPC-DI• VPC-SI
Feature Default	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	<ul style="list-style-type: none">• <i>ASR 5500 System Administration Guide</i>• <i>Command Line Interface Reference</i>• <i>VPC-DI System Administration Guide</i>• <i>VPC-SI System Administration Guide</i>

Revision History



Important

Revision history details are not provided for features introduced before releases 21.2 and N5.1.

Revision Details	Release
With this release, changes are made to the maximum number of core transfers and the format of the core filename.	21.20
First introduced.	Pre 21.2

Feature Changes

Previous Behavior:

- Maximum number of core transfers:

When more than one process crashes, the simultaneous core file transfer was limited to a maximum of two at any given point of time and based on first come first serve basis.

The maximum simultaneous core file transfer was increased from one to two cores as VPP crashes could induce simultaneous crashes in the box. At least one VPP full core has to be transferred at any time for debugging.

Issue: When the process crashes in the 1st non-VPP -> 2nd non-VPP -> 3rd VPP sequence, the VPP core is discarded, transfer is based on first come first serve basis, and maximum of two cores are transferred.

- Core Filename:

When the process crashes, the core file is transferred and stored with the filename compiled by its card number, CPU number, and hextime to make it unique and identifiable.

Format: **crash-<cardno>-<cpuno>-<hextime>-core**

Issue: When two or more processes crash at the same hextime along with the same card and CPU number, the generated cores are written to the same core filename causing corruption.

New Behavior:

- Maximum number of core transfers:

The maximum number of core transfers are restricted to two in case of [1] and one in case of [2] with one non-VPP and one VPP always at any given point of time.

1. Maximum core transfer to two: For cores generated in the below sequences, both 1st and 2nd cores are transferred.

1st non-VPP -> 2nd VPP core

1st VPP + 2nd non-VPP core

2. Maximum core transfer to one: For cores generated in the below sequences, the 1st core is transferred and the 2nd core is discarded.

1st VPP -> 2nd VPP core

1st non-VPP -> 2nd non-VPP core

- Core File Name:

For all cores, the filename is extended by adding the PID of the process to make it unique, even when two or more processes crash at the same hextime along with the same card and CPU number.

Format: **crash-<cardno>-<cpuno>-<pid>-<hextime>-core**

Customer Impact:

The scripts must be updated to the new format, if the coded core filename is in the old format.

Old format for core file: **crash-<cardno>-<cpuno>-<hextime>-core**

New format for core file: **crash-<cardno>-<cpuno>-<pid>-<hextime>-core**

Command Changes

Configuring VPP Core Transfer

Use the following configuration to enable or disable mandating VPP core transfer along with non-VPP.

```
configure
  [ no ] crash enable vpp-core-transfer
exit
```

NOTES:

- **crash enable vpp-core-transfer**: Enables mandating VPP core transfer.
- **no crash enable vpp-core-transfer**: Disables mandating VPP core transfer.
- Default: Enabled

Performance Indicator Changes

show crash config

The existing **show crash config** command is enhanced to display the VPP core transfer status. The **Mandatory VPP Core Transfer** field displays whether VPP core transfer is enabled or disabled.

Sample Output:

```
# show crash config
URL : /hd-raid/cores
Disk Space Limit : Not Configured
Rotate Core Files Limit : 15 (default)
Core File Max-Size : 4096 MB
Core File Compression : gzip
Core Transmit Timeout : 120 seconds
Core Obfuscation : disabled
Async Core Transfer : enabled
Mandatory VPP Core Transfer : enabled
Critical Task : enabled
#
```

