Capturing and Analyzing Kernel Crash Dumps with Kdump

When a Linux kernel panics, Kdump saves the memory of the crashed kernel to a file, `vmcore`, so that you can find out what caused the crash. Built on Kexec, Kdump keeps a second kernel ready in reserved memory and starts it the moment the main kernel crashes. This article explains how to set up Kdump on SUSE Linux Enterprise Server for NVIDIA. You install and enable Kdump, reserve enough memory for it and test that it captures dumps. Finally, you analyze a captured dump.

Prerequisites

- Enough free disk space at the dump target. By default, Kdump filters and compresses the dump, so it is smaller than the system memory.

Publication Date: 26 Nov 2026

Contents

1 What is Kdump and how does it use Kexec? 3

2 Installing and enabling Kdump 5

3 Sizing and reserving crash kernel memory 7

4 Saving crash dumps to a remote target 9

5 Verifying that Kdump captures a crash dump 10

6 Analyzing a captured `vmcore` file 12

7 Key options in `/etc/sysconfig/kdump` 15

8 Troubleshooting 18

9 For more information 19

10 Legal Notice 19

A GNU Free Documentation License 20

1 What is Kdump and how does it use Kexec?

Kdump is a crash-dumping mechanism used to capture a system memory image at the moment a kernel panics or crashes, allowing administrators to perform debugging after a crash. It does not save the dump from the crashed kernel itself. Instead, it starts a second kernel via Kexec and captures the dump from there.

1.1 The dual-kernel model

To save the memory of a crashed kernel, Kdump relies on two kernels:

- **The production kernel** is the kernel that normally runs your system and its workloads.
- **The capture kernel** (also called the crash kernel or Kdump kernel) is a second, minimal kernel image with its own `initramfs`. It runs in an area of memory that is reserved for it. Kdump builds the `initramfs` so that it contains the drivers and tools, such as **`makedumpfile`**, needed to save the dump.

When the production kernel panics, it can no longer be trusted to write its own crash information to disk: Its data structures may be corrupted, and a normal reboot would overwrite the memory you are trying to inspect. Because the capture kernel runs from its own reserved memory region, the production kernel's memory contents remain intact and readable while the capture kernel is running.

1.2 How Kdump uses Kexec

Kexec is the mechanism that starts the capture kernel: a system call and the **`kexec`** command, which load a kernel into memory and start it directly, without going through the firmware (BIOS/UEFI) and the boot loader. Kdump is the service built around this mechanism. It reserves the memory for the capture kernel, builds its `initramfs` and defines how and where the dump is saved.

When the system boots and later crashes, the following happens:

1. At boot, the kernel reserves an area of memory for the capture kernel. The size comes from the `crashkernel=` option on the kernel command line.
2. `kdump.service` loads the capture kernel and its `initramfs` into the reserved area with **`kexec -p`**. The capture kernel stays idle.

3. When the production kernel panics, it uses Kexec to start the capture kernel. The firmware and the boot loader do not run, so the memory of the crashed kernel stays intact.

4. The capture kernel saves that memory to a `vmcore` file, together with the kernel log, and reboots the system.

Enabling Kdump does not change normal reboots. Kdump loads the capture kernel only for a crash, so a normal reboot still goes through the firmware and the boot loader. You can also use Kexec on its own to reboot faster, as described in the separate article “Using Kexec for Faster Reboots”.

1.3 The Kdump services and configuration file

Kdump runs as four systemd services. Enabling `kdump.service` also enables the other three. The services read their settings from `/etc/sysconfig/kdump`, where comments describe each option. For the frequently used options and how to apply changes, see *Section 7, “Key options in `/etc/sysconfig/kdump`”*.

`kdump-early.service`
: Loads the capture kernel early during boot, before `kdump.service` runs.

`kdump.service`
: Rebuilds the capture `initramfs` if necessary and loads the capture kernel into the reserved memory. The service fails if no memory is reserved.

`kdump-commandline.service`
: Runs after `kdump.service` and checks whether the kernel command line contains the expected `crashkernel=` values. If they are missing or different, it updates the boot loader configuration, unless `KDUMP_UPDATE_BOOTLOADER` is set to `false`.

`kdump-notify.service`
: Sends an e-mail notification about new crash dumps at boot, using **`mailx`**. It only does this if `KDUMP_NOTIFICATION_TO` contains an e-mail address and the dumps are saved locally.

1.4 What data does a captured `vmcore` file contain?

Once the capture kernel is running, it writes the production kernel's memory contents to a file called `vmcore`, together with a copy of the kernel's ring buffer log (`dmesg`) from the moment of the crash. Depending on the dump level set in `KDUMP_DUMPLEVEL`, a `vmcore` file can contain:

- **Kernel state:** active kernel data structures, global variables and the call stack that was executing at the moment of the crash
- **Process information:** the list of processes that were running, along with their individual stacks and registers
- **Memory pages:** the actual data held in RAM by the kernel and by user-space applications
- **VMCOREINFO:** a section that records how the kernel's memory was laid out, such as the kernel release, the page size and the addresses of kernel symbols, so that analysis tools can interpret the raw memory

You analyze this file after the system has rebooted, using tools such as **`crash`** or GDB. See *Section 6, "Analyzing a captured `vmcore` file"* for details.

2 Installing and enabling Kdump

Depending on how the system was installed, Kdump may not be installed or enabled. Make sure it is both, so that a crash produces a dump you can analyze afterward.

2.1 Installing Kdump

1. Install the Kdump package. The package pulls in `kexec-tools` and `makedumpfile` as dependencies:

```
> sudo  zypper install kdump
```

2. Enable and start the Kdump service:

```
> sudo  systemctl enable --now kdump
```

If no crash kernel memory is reserved yet, `kdump.service` fails to start. The journal then shows `No crashkernel memory reserved, Kdump cannot start.` `kdump-commandline.service` still adds the `crashkernel=` values to the boot loader configuration. Kdump starts after the reboot in the next step.

3. Reboot so the crash kernel memory reservation takes effect:

```
> sudo reboot
```

The default `KDUMP_CRASHKERNEL="auto"` reserves an automatically calculated amount of memory, which works for most systems. To check the value or set your own, see *Section 3, "Sizing and reserving crash kernel memory"*.

2.2 What happens when Kdump starts

When `kdump.service` starts, it rebuilds the capture `initramfs` if necessary and loads the capture kernel and the `initramfs` into the reserved memory area. Afterward, `kdump-commandline.service` checks whether the running kernel's command line (`/proc/cmdline`) contains the `crashkernel=` values that `KDUMP_CRASHKERNEL` specifies. With the default `KDUMP_CRASHKERNEL="auto"`, these are the values that **`kdumptool calibrate`** calculates, and small differences in size are ignored. If the values are missing or different and `KDUMP_UPDATE_BOOTLOADER` is `true` (the default), `kdump-commandline.service` updates the boot loader configuration through **`pbl`**(8).

To check whether the service found and applied any change:

```
> sudo journalctl -u kdump-commandline.service
```

If `kdump-commandline.service` updated the boot loader configuration, a reboot is required before the new `crashkernel=` values take effect and Kdump can start capturing dumps.

Important: Keep manual `crashkernel=` values out of `GRUB_CMDLINE_LINUX`

On a system where `KDUMP_UPDATE_BOOTLOADER` is `true`, `kdump-commandline.service` replaces the `crashkernel=` values in `GRUB_CMDLINE_LINUX_DEFAULT` in `/etc/default/grub` and never changes `GRUB_CMDLINE_LINUX`. A `crashkernel=` value in `GRUB_CMDLINE_LINUX` applies to every boot entry, including the recovery entries, which otherwise carry no reservation. On the normal boot entry, it then appears alongside the value the service sets, so the command line carries two conflicting reservations.

To set `crashkernel=` yourself, set `KDUMP_UPDATE_BOOTLOADER="false"` in `/etc/sysconfig/kdump` so the service no longer rewrites it, and put the value only in `GRUB_CMDLINE_LINUX_DEFAULT`.

3 Sizing and reserving crash kernel memory

Before Kdump can capture a crash dump, a dedicated area of memory must be reserved for the capture kernel. Use **`kdumptool calibrate`** to determine a suitable size, or set it manually if the automatic value is not sufficient.

3.1 Using the automatically calculated size

By default, `KDUMP_CRASHKERNEL` in `/etc/sysconfig/kdump` is set to `auto`. With this setting, `kdumptool commandline` derives the crash kernel memory size from **`kdumptool calibrate`** automatically. You do not need to calculate or set anything manually to get started. Run the calibration tool directly to see the values it would use:

EXAMPLE 1: DETERMINING THE RECOMMENDED CRASH KERNEL MEMORY SIZE

```
> sudo  kdumptool calibrate
```

All values are given in megabytes. On AArch64, the whole reservation is reported as `Low`, and `High` is 0. `MinLow`/`MaxLow` and `MinHigh`/`MaxHigh` give the valid range around those recommendations. `NonCMA` and `CMA` are the suggested sizes for the two parts of the reservation when `KDUMP_USE_CMA` is enabled. See *Section 7, "Key options in `/etc/sysconfig/kdump`"*.



Note: Calibrated values depend on the system

The exact numbers returned by **`kdumptool calibrate`** depend on the amount of RAM and the hardware configuration of the machine it runs on. Always run it on the target system itself rather than reusing values from another machine.

3.2 Crash kernel memory formats

When you set `KDUMP_CRASHKERNEL` manually, the value is either `auto` or one or more `crashkernel=` options in one of these formats:

Fixed size

`crashkernel=`*`SIZE`*`[@`*`OFFSET`*`]` reserves a fixed amount of memory; for example, `crashkernel=256M`. The optional offset is the physical address where the reserved area starts. A suitable size is the sum of the `Low` and `High` values from **`kdumptool calibrate`**. On AArch64, `auto` produces this form.

Low and high split

`crashkernel=`*`SIZE_LOW`*`M,low crashkernel=`*`SIZE_HIGH`*`M,high` reserves one area below the 4 GiB mark and one from the remaining memory; for example, `crashkernel=72M,low crashkernel=384M,high`.

Size depending on installed RAM

`crashkernel=`*`RANGE1`*`:`*`SIZE1`*`[,`*`RANGE2`*`:`*`SIZE2`*`,...][@`*`OFFSET`*`]` reserves a size that depends on the amount of installed RAM. Each range has the form *`START`*-[*`END`*], where the start is inclusive and the end is exclusive. For example, `crashkernel=512M-2G:64M,2G-:128M` reserves:

- Nothing on systems with less than 512 MiB of RAM
- 64 MiB on systems with 512 MiB up to but not including 2 GiB
- 128 MiB on systems with 2 GiB or more

3.3 Setting a manual value

Override the automatically calculated reservation manually if it is not sufficient, for example, if Kdump still runs out of memory while capturing a dump. Take the value that **`kdumptool calibrate`** recommends and, if the system has multiple LUN kernel paths (paths to storage devices) attached, add roughly half the number of LUNs to it:

```
SIZE = RECOMMENDATION + (LUNS / 2)
```

Where *`LUNS`* is the maximum number of LUN kernel paths you expect the system to ever have (excluding multipath devices, which are ignored). To count the LUNs currently visible:

```
> grep -c Lun /proc/scsi/scsi
```

On AArch64, only the `Low` value applies. Use it as *`RECOMMENDATION`* and set the result in `/etc/sysconfig/kdump`:

```
KDUMP_CRASHKERNEL="crashkernel=SIZEM"
```

3.4 Applying a changed reservation

After changing `KDUMP_CRASHKERNEL`, a full system reboot is required, since the crash kernel memory reservation is applied at boot time. See *Section 2, "Installing and enabling Kdump"* for how the new value is picked up.

4 Saving crash dumps to a remote target

By default, Kdump saves crash dumps locally under `/var/crash`. On systems with limited local storage, or where dumps must be collected centrally, configure `KDUMP_SAVEDIR` to save dumps to a remote target instead.

4.1 Supported remote storage protocols

`KDUMP_SAVEDIR` in `/etc/sysconfig/kdump` accepts a URL in the form *`PROTOCOL`*://*`SPECIFICATION`*. Install the client package matching the protocol you plan to use:

- `nfs-client`, for `nfs://` targets
- `cifs-utils`, for `cifs://` targets (Samba/Windows shares)
- `lftp`, for `ftp://` targets
- `openssh-clients`, for `ssh://` and `sftp://` targets

For example, to save dumps to an FTP server:

```
KDUMP_SAVEDIR="ftp://USERNAME:PASSWORD@FTP_HOST:PORT/var/crash"
```

4.2 Saving dumps over SSH

A crash dump can contain sensitive data from system memory. SSH keeps that data encrypted in transit. SFTP is the preferred transport, since it does not require the target host to run any special server-side commands beyond the SFTP subsystem, which is enabled by default on SLE for NVIDIA.

1. Make the target host's identity known to Kdump ahead of time, so a dump is never sent to an impersonating host. Connect to the target once as `root` from the Kdump host so its key is recorded:

   ```
   > sudo  ssh TARGET_HOST
   ```

2. Create a dedicated key for Kdump rather than reusing `root`'s own key:

   ```
   > sudo  ssh-keygen -f /root/.ssh/kdump_key -N ""
   ```

   Leave the passphrase empty, since the capture kernel environment cannot prompt for one.

3. Authorize the new key on the target host:

   ```
   > sudo  ssh-copy-id -i /root/.ssh/kdump_key TARGET_HOST
   ```

4. In `/etc/sysconfig/kdump`, point Kdump at the new key and set the save directory:

   ```
   KDUMP_SSH_IDENTITY="/root/.ssh/kdump_key"
   KDUMP_SAVEDIR="sftp://TARGET_HOST/path/to/dumps"
   ```

5. Restart Kdump to apply the new configuration:

   ```
   > sudo  systemctl restart kdump
   ```

5 Verifying that Kdump captures a crash dump

After configuring Kdump, validate the setup by triggering a test crash. Verifying dump capture in advance ensures that crash dumps (`vmcore`) are generated as expected during an unscheduled system failure.



Warning: This test crashes the system

Triggering a kernel panic immediately restarts the system, the same as a real crash would. Make sure no important workloads are running and no other users are logged in before you continue.

1. Force an emergency sync of cached data to disk, to reduce the chance of data loss from the deliberate crash:

```
>  echo s | sudo tee /proc/sysrq-trigger
```

2. Force an emergency remount of file systems to read-only, to reduce the risk of corruption:

```
>  echo u | sudo tee /proc/sysrq-trigger
```

3. Trigger the crash:

```
>  echo c | sudo tee /proc/sysrq-trigger
```

   The system panics immediately. Kdump captures a dump, then the system reboots on its own. Saving the dump can take a few minutes, depending on how much memory needs to be written.

4. Once the system is back, confirm a dump was captured:

EXAMPLE 2: CONFIRMING A CAPTURED CRASH DUMP

```
> sudo  ls -la /var/crash/
```

Sample output:

```
drwx------  1 root root  42 Oct 27 10:13 2026-10-27-10-13
```

Kdump creates one directory per captured dump, named after the date and time of the crash (`YYYY-MM-DD-HH-MM`). Inside it:

EXAMPLE 3: CONTENTS OF A CAPTURED DUMP DIRECTORY

```
> sudo  ls -la "/var/crash/DUMP_DIRECTORY/"
```

Sample output:

```
-rw-------  1 root root    81554 Oct 27 10:13 dmesg
-rw-------  1 root root      308 Oct 27 10:13 README.txt
-rw-------  1 root root 68839116 Oct 27 10:13 vmcore
```

- `dmesg` is the kernel's own ring buffer log at the moment of the crash.
- `vmcore` is the actual memory snapshot. See *Section 6, "Analyzing a captured `vmcore` file"* to read it.
- `README.txt` summarizes the capture itself: kernel version, crash and dump time stamps, dump level and dump format. Check it first if a capture looks incomplete or was truncated.

6 Analyzing a captured `vmcore` file

Once Kdump has captured a `vmcore` file, use the **`crash`** utility to inspect it and find out what the kernel was doing at the moment it panicked.

6.1 Installing the analysis tools

1. Enable the product's `Debug` repository. It is disabled by default. List your repositories to find its exact alias, then enable it:

```
> sudo  zypper lr | grep -i debug
> sudo  zypper modifyrepo --enable DEBUG_REPO_ALIAS
```

2. Install **`crash`** and the debuginfo package of the running kernel. The package name contains the kernel flavor, the last part of the **`uname -r`** output. For the `default` flavor, this is `kernel-default-debuginfo`. The following command inserts the flavor automatically:

```
> sudo  zypper install crash kernel-$(uname -r | sed 's/.*-//')-debuginfo
```



Note: The debuginfo package version must match the running kernel exactly

If the debuginfo package installs a version newer than the kernel you crashed (for example, because an update arrived after the crash but before you reboot), **`crash`** refuses to load it with a version mismatch error. Reboot onto the newest installed kernel first, and confirm **`uname -r`** matches the `Kernel version` line in the crash directory's `README.txt` before proceeding.

3. The debuginfo package alone is not enough: On SLE for NVIDIA, its `vmlinux.debug` file contains only debug symbols, not the actual kernel code and data **`crash`** needs alongside them. Place the real kernel image from the matching kernel package next to `vmlinux.debug`. The kernel package contains it compressed, so decompress it:

```
> xz -dc /usr/lib/modules/$(uname -r)/vmlinux.xz | \
    sudo tee /usr/lib/debug/usr/lib/modules/$(uname -r)/vmlinux > /dev/null
```

6.2 Opening the `vmcore` file in **`crash`**

Start **`crash`**, giving it the kernel image you just placed and the captured `vmcore` file:

```
> sudo  crash /usr/lib/debug/usr/lib/modules/$(uname -r)/vmlinux \
    "/var/crash/DUMP_DIRECTORY/vmcore"
```

Sample output, from a crash deliberately triggered as described in *Section 5, "Verifying that Kdump captures a crash dump"*:

```
      KERNEL: /usr/lib/debug/usr/lib/modules/6.12.0-160100.7-default/vmlinux
   DEBUGINFO: /usr/lib/debug/usr/lib/modules/6.12.0-160100.7-default//vmlinux.debug
    DUMPFILE: /var/crash/2026-10-27-10-13/vmcore  [PARTIAL DUMP]
        CPUS: 2
        DATE: Tue Oct 27 10:13:42 CET 2026
      UPTIME: 00:00:21
LOAD AVERAGE: 0.59, 0.14, 0.05
       TASKS: 180
    NODENAME: sles16
     RELEASE: 6.12.0-160100.7-default
     VERSION: #1 SMP PREEMPT_DYNAMIC Wed Sep 23 08:40:15 UTC 2026 (3a3c8e4)
     MACHINE: x86_64  (2304 Mhz)
      MEMORY: 4 GB
       PANIC: "Kernel panic - not syncing: sysrq triggered crash"
         PID: 1290
     COMMAND: "tee" (1)
        TASK: ffff8d67dd620000  [THREAD_INFO: ffff8d67dd620000]
         CPU: 1
       STATE: TASK_RUNNING (PANIC)
crash> bt
PID: 1290     TASK: ffff8d67dd620000  CPU: 1    COMMAND: "tee"
 #0 [fffffce83c0d8f8e0] machine_kexec at ffffffffb26916a2
 #1 [fffffce83c0d8f940] __crash_kexec at ffffffffb27d2cab
 #2 [fffffce83c0d8fa00] panic at ffffffffb3360b1d
 #3 [fffffce83c0d8fa80] sysrq_handle_crash at ffffffffb2ec2dba
 #4 [fffffce83c0d8fa88] __handle_sysrq.cold at ffffffffb339e7a8
```

```
 #5 [ffffce83c0d8fab8] write_sysrq_trigger at ffffffffb2ec3959
 #6 [ffffce83c0d8fae0] proc_reg_write at ffffffffb2b1be67
 #7 [ffffce83c0d8faf8] vfs_write at ffffffffb2a6a45b
 #8 [ffffce83c0d8fb90] ksys_write at ffffffffb2a6aac7
 #9 [ffffce83c0d8fbc8] do_syscall_64 at ffffffffb33bea1f
#10 [ffffce83c0d8ff40] entry_SYSCALL_64_after_hwframe at ffffffffb340012f
    RIP: 00007faad43c8f04  RSP: 00007ffeba327138  RFLAGS: 00000202
    RAX: ffffffffffffffda  RBX: 0000000000000002  RCX: 00007faad43c8f04
    RDX: 0000000000000002  RSI: 00007ffeba3272b0  RDI: 0000000000000003
    RBP: 00007ffeba3272b0   R8: 0000000000000004   R9: 0000000000000001
    R10: 00000000000001b6  R11: 0000000000000202  R12: 0000000000000002
    R13: 0000555607ac12c0  R14: 00007faad44aaea0  R15: 0000000000000001
    ORIG_RAX: 0000000000000001  CS: 0033  SS: 002b
crash> quit
```

1. The process that happened to be running when the crash was triggered, not necessarily a sign of a problem with that process. Here it is `tee` because the crash in this example was deliberately triggered with `echo c | sudo tee /proc/sysrq-trigger`. On a real, unplanned crash this would show whatever command was running at the time.

The banner shows the panic reason, the running process and basic system information. **`bt`** without arguments shows the call stack of the process active at the moment of the crash, here showing the path from the `sysrq-trigger` write down to `machine_kexec`, which is Kexec handing control to the capture kernel.

6.3 Useful follow-up commands

`ps`
: Lists the processes that were running at the moment of the crash.

`sym` *`SYMBOL_OR_ADDRESS`*
: Resolves a kernel symbol to its address, or an address back to a symbol.

`files`
: Lists the open file descriptors of the current process.

`kmem`
: Displays kernel memory usage details.

`crash` has extensive built-in help: Run **`help`** *`COMMAND`* inside a session for details on any of these.

7 Key options in /etc/sysconfig/kdump

The following table summarizes the /etc/sysconfig/kdump options you need most often.

TABLE 1: FREQUENTLY USED /etc/sysconfig/kdump OPTIONS

Option	Purpose
KDUMP_CRASHKERNEL	Crash kernel memory reservation. auto (default) uses the value from **kdumptool calibrate**. See *Section 3, "Sizing and reserving crash kernel memory"*.
KDUMP_UPDATE_BOOTLOADER	When true (default), kdump-command-line.service updates the boot loader configuration whenever the running kernel command line does not match the expected crashkernel= options. Set to false to only check and warn, without modifying the boot loader.
KDUMP_USE_CMA	Experimental (default: false). When true, the crash kernel reservation is split into a standard part and a part taken from the Contiguous Memory Allocator (CMA); for example, crashkernel=300M crashkernel=640M,cma. The CMA part stays available to user space on the running system. Kdump can become less reliable because DMA transfers set up by the production kernel can corrupt the memory of the capture kernel. KDUMP_DUMPLEVEL must filter out user-space pages, as the default value 31 does. The kernel does not support CMA crash kernel reservations on AArch64.
KDUMP_KERNELVER	Kernel version string to use for the capture kernel. Leave empty to auto-detect a suitable kernel.
KDUMP_CPUS	Number of CPUs available to the capture kernel environment (default: 4, or all available CPUs if set to 0). Lowering this reduces the memory the

Option	Purpose
	capture kernel needs. Updating the `kdump` package keeps a value that is already set in `/etc/sysconfig/kdump`, so an updated system can still use the previous default of 32.
`KDUMP_SAVEDIR`	Where captured dumps are stored. Defaults to `/var/crash`. See *Section 4, "Saving crash dumps to a remote target"* for remote targets.
`KDUMP_KEEP_OLD_DUMPS`	Number of old dumps to keep. Honored only for local save directories. The default, `0`, never deletes old dumps. `-1` deletes all old dumps, so that only the newly saved dump remains.
`KDUMP_FREE_DISK_SIZE`	Minimum free space, in megabytes, that must remain at the save location after writing a dump. If less space remains, Kdump deletes the dump again. Applies only to local save directories. Default: `64`.
`KDUMP_DUMPLEVEL`	How much of memory to include in the dump, from `0` (largest, most complete dump) to `31` (smallest, the default). See `man 5 kdump` for the table of levels.
`KDUMP_DUMPFORMAT`	Format of the saved dump. The default, `compressed`, produces small dumps that only **`crash`** can analyze. `ELF` produces larger dumps that GDB can also read. `none` saves only the kernel log. See `man 5 kdump` for the other formats.
`KDUMP_IMMEDIATE_REBOOT`	When `true` (default), the system reboots automatically once the dump is saved.
`KDUMP_CONTINUE_ON_ERROR`	When `true` (default), the capture kernel continues with the next action after an error while saving or deleting dumps. Set to `false` to open a

Option	Purpose
	shell at that point instead. See *Section 8, "Troubleshooting"*.
`KDUMP_VERBOSE`	Bit mask that sets the verbosity of the dump process (default: `0`). See *Section 8, "Troubleshooting"*.
`KDUMP_COMMANDLINE`	If set, Kdump uses this command line for the capture kernel instead of building one. The command line that Kdump builds by default contains options the capture kernel needs, such as `root=kdump`, `irqpoll`, `reset_devices` and `numa=off`. To add options, use `KDUMP_COMMANDLINE_APPEND` instead.
`KDUMP_COMMANDLINE_APPEND`	Extra options to append to the command line of the capture kernel. For example, an `ip=` option configures the network of the capture kernel manually. Kdump then skips its own network configuration.
`KDUMP_SSH_IDENTITY`	Path to the SSH private key used when `KDUMP_SAVEDIR` points at an `ssh://` or `sftp://` target. See *Section 4, "Saving crash dumps to a remote target"*.



Important: Restart Kdump after editing its configuration

After changing `/etc/sysconfig/kdump`, run **`systemctl restart kdump`** for the change to take effect. An exception is `KDUMP_CRASHKERNEL`: Since it changes the boot-time memory reservation, it requires a full system reboot, not just a service restart. See *Section 2, "Installing and enabling Kdump"*.

8 Troubleshooting

Kdump usually fails at one of three points: The crash kernel memory reservation is too small, the capture kernel does not start or the dump is not saved. Use the checks below to narrow down which stage is failing.

8.1 Crash kernel memory reservation is insufficient

If a crash produces error messages about low memory, or the Out Of Memory (OOM) killer being invoked inside the capture kernel, the reserved crash kernel memory is too small for this system. The reservation can be too small even after following *Section 3, "Sizing and reserving crash kernel memory"*, since the right amount depends on the actual workload running at the time of the crash.

1. Check the currently reserved amount from the kernel command line:

   ```
   > cat /proc/cmdline
   ```

   Add up every `crashkernel=` value shown (a plain value, or separate `,low`/`,high` values) to get the current total reservation in MiB.

2. Edit `/etc/sysconfig/kdump` and roughly double `KDUMP_CRASHKERNEL` from the current total:

   ```
   KDUMP_CRASHKERNEL="crashkernel=DOUBLED_SIZEM"
   ```

3. Reboot to apply the new reservation, then try to reproduce the crash again. See *Section 5, "Verifying that Kdump captures a crash dump"*.

   ```
   > sudo reboot
   ```

4. Repeat with smaller increases until the dump completes reliably. A note taken between the last failing and the last working size narrows down a reasonable final value faster than repeatedly doubling.

8.2 Other common issues

The screen stays blank during the capture kernel boot

Configure a serial console so the capture kernel's boot messages are captured somewhere you can read them. A blank local console gives you nothing to debug from. Once you have output, increase logging by setting `KDUMP_VERBOSE="11"` in `/etc/sysconfig/kdump`. The value 11 removes the `quiet` option from the command line of the capture kernel. It also turns on debugging output for **`makedumpfile`** and for the **`kdump-save`** script, which saves the dump.

`makedumpfile` or network configuration errors during capture

Set `KDUMP_CONTINUE_ON_ERROR="false"` or `KDUMP_IMMEDIATE_REBOOT="false"`. Instead of rebooting automatically, this drops you into an interactive shell inside the capture environment when an error occurs, or once **`makedumpfile`** finishes, so you can inspect the problem directly.

Dracut errors during the capture kernel boot

Add dracut's own debugging options to `KDUMP_COMMANDLINE_APPEND` in `/etc/sysconfig/kdump`; for example, `KDUMP_COMMANDLINE_APPEND="rd.debug"`.

The capture `initramfs` does not generate correctly

Regenerate it with debugging enabled to see where it fails:

```
> sudo  mkdumprd -f -d
```

To inspect the contents of the generated image directly:

```
> sudo  lsinitrd --unpack /var/lib/kdump/initrd
```

9 For more information

- Man pages: `man 5 kdump`, `man 7 kdump`
- Linux kernel documentation on Kdump: https://docs.kernel.org/admin-guide/kdump/kdump.html

10 Legal Notice

Copyright© 2006– 2026 SUSE LLC and contributors. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or (at your option) version 1.3; with the Invariant Section being this copyright notice and license. A copy of the license version 1.2 is included in the section entitled “GNU Free Documentation License”.

For SUSE trademarks, see https://www.suse.com/company/legal/. All other third-party trademarks are the property of their respective owners. Trademark symbols (®, ™ etc.) denote trademarks of SUSE and its affiliates. Asterisks (*) denote third-party trademarks.

All information found in this book has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Neither SUSE LLC, its affiliates, the authors, nor the translators shall be held liable for possible errors or the consequences thereof.

A GNU Free Documentation License

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image for-

mats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally

prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See https://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled “GNU
Free Documentation License”.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.