NAME

core – core image file

DESCRIPTION

The operating system writes out a core image of a process when it is terminated due to the receipt of some signals. The core image is called **core** and is written in the process's working directory (provided it can be; normal access controls apply). A process with an effective user ID different from the real user ID will not produce a core image.

The core file contains all the process information pertinent to debugging: contents of hardware registers, process status and process data. The format of a core file is object file specific.

For ELF executable programs (see **a.out**(4)), the core file generated is also an ELF file, containing ELF program and file headers. The **e_type** field in the file header has type **ET_CORE**. The program header contains an entry for every loadable and writable segment that was part of the process address space, including shared library segments. The contents of the segments themselves are also part of the core image.

The program header of an ELF core file also contains a **NOTE** segment. This segment may contain the following entries. Each has entry name **"CORE"** and presents the contents of a system structure:

prstatus_t
The entry containing this structure has a **NOTE** type of 1. This structure contains things of interest to a debugger from the operating system's u-area, such as the general registers, signal dispositions, state, reason for stopping, process ID and so forth. The **prstatus_t** structure is defined in **<sys/procfs.h>**.

prfpregset_t
This entry is present only if the process used the floating-point hardware. It has a **NOTE** type of 2 and contains the floating-point registers. The **prfpregset_t** structure is defined in **<sys/procfs.h>**.

prpsinfo_t
The entry containing this structure has a **NOTE** type of 3. It contains information of interest to the **ps**(1) command, such as process status, cpu usage, "nice" value, controlling terminal, user ID, process ID, the name of the executable and so forth. The **prpsinfo_t** structure is defined in **<sys/procfs.h>**.

prxregset_t
This entry is present only if the process has extra register state associated with it. It has a **NOTE** type of 4 and contains the extra register state. The **prxregset_t** structure is defined in **<sys/procfs.h>**.

char *
The entry containing this structure has a **NOTE** type of 5 and contains a string describing the specific model of the hardware platform on which this core file was created. This information is the same as provided by **sysinfo**(2) when invoked with the command **SI_PLATFORM**.

auxv_t array
The entry containing this structure has a **NOTE** type of 6. It contains values of entries in the auxiliary vector that is passed by the operating system as startup

information to the dynamic linker. Auxiliary vector information is defined in **<sys/auxv.h>**.

The size of the core file created by a process may be controlled by the user (see **getrlimit**(2)).

SEE ALSO

adb(1), **gcore**(1), **crash**(1M), **getrlimit**(2), **setuid**(2), **sysinfo**(2), **elf**(3E), **a.out**(4), **proc**(4), **signal**(5)

ANSI C Programmer's Guide