

# 1 Struttura dati del descrittore di processo di Linux (task\_struct)

```

struct task_struct {
/* these are hardcoded - don't touch */
volatile long      state;          /* -1 unrunnable, 0 runnable, >0 stopped */
long               counter;
long               priority;
unsigned           long signal;
unsigned           long blocked;   /* bitmap of masked signals */
unsigned           long flags;    /* per process flags, defined below */
int               errno;
long               debugreg[8];   /* Hardware debugging registers */
struct exec_domain *exec_domain;
/* various fields */
struct linux_binfmt *binfmt;
struct task_struct *next_task, *prev_task;
struct task_struct *next_run, *prev_run;
unsigned long      saved_kernel_stack;
unsigned long      kernel_stack_page;
int                exit_code, exit_signal;
/* ??? */
unsigned long      personality;
int                dumpable:1;
int                did_exec:1;
int                pid;
int                pgrp;
int                tty_old_pgrp;
int                session;
/* boolean value for session group leader */
int                leader;
int                groups[NGROUPS];
/*
 * pointers to (original) parent process, youngest child, younger sibling,
 * older sibling, respectively. (p->father can be replaced with
 * p->p_pptr->pid)
 */
struct task_struct *p_opptr, *p_pptr, *p_cptra,
                  *p_ysptr, *p_osptra;
struct wait_queue *wait_chldexit;
unsigned short    uid,euid,suid,fsuid;
unsigned short    gid,egid,sgid,fsgid;
unsigned long      timeout, policy, rt_priority;
unsigned long      it_real_value, it_prof_value, it_virt_value;
unsigned long      it_real_incr, it_prof_incr, it_virt_incr;
struct timer_list real_timer;
long               utime, stime, cutime, cstime, start_time;
/* mm fault and swap info: this can arguably be seen as either
   mm-specific or thread-specific */
unsigned long      min_flt, maj_flt, nswap, cmin_flt, cmaj_flt, cnswap;
int                swappable:1;
unsigned long      swap_address;
unsigned long      old_maj_flt;   /* old value of maj_flt */
unsigned long      dec_flt;       /* page fault count of the last time */
unsigned long      swap_cnt;      /* number of pages to swap on next pass */
/* limits */
struct rlimit      rlim[RLIM_NLIMITS];
unsigned short     used_math;
char               comm[16];
/* file system info */
int                link_count;
struct tty_struct *tty;          /* NULL if no tty */
/* ipc stuff */
struct sem_undo    *semundo;
struct sem_queue   *semsleeping;
/* ldt for this task - used by Wine. If NULL, default_ldt is used */
struct desc_struct *ldt;
/* tss for this task */
struct thread_struct tss;
/* filesystem information */
struct fs_struct   *fs;
/* open file information */

```

```
    struct files_struct *files;
/* memory management info */
    struct mm_struct *mm;
/* signal handlers */
    struct signal_struct *sig;
#ifdef __SMP__
    int processor;
    int last_processor;
    int lock_depth; /* Lock depth.
                    We can context switch in and out
                    of holding a syscall kernel lock... */
#endif
};
```