struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device) {

struct [**kobject**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kobject) [**kobj**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kobj);

struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device) \*[**parent**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/parent);

struct [**device\_private**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device_private) \*p;

const char \*[**init\_name**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/init_name); */\* initial name of the device \*/*

const struct [**device\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device_type) \*type;

struct [**bus\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bus_type) \*[**bus**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bus); */\* type of bus device is on \*/*

struct [**device\_driver**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device_driver) \*driver; */\* which driver has allocated this*

*device \*/*

void \*[**platform\_data**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/platform_data); */\* Platform specific data, device*

*core doesn't touch it \*/*

void \*[**driver\_data**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/driver_data); */\* Driver data, set and get with*

*dev\_set\_drvdata/dev\_get\_drvdata \*/*

#ifdef [**CONFIG\_PROVE\_LOCKING**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_PROVE_LOCKING)

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex) [**lockdep\_mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lockdep_mutex);

#endif

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex) [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex); */\* mutex to synchronize calls to*

*\* its driver.*

*\*/*

struct [**dev\_links\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_links_info) [**links**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/links);

struct [**dev\_pm\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_pm_info) [**power**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/power);

struct [**dev\_pm\_domain**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_pm_domain) \*[**pm\_domain**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pm_domain);

#ifdef [**CONFIG\_ENERGY\_MODEL**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_ENERGY_MODEL)

struct [**em\_perf\_domain**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/em_perf_domain) \*[**em\_pd**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/em_pd);

#endif

#ifdef [**CONFIG\_PINCTRL**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_PINCTRL)

struct [**dev\_pin\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_pin_info) \*[**pins**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pins);

#endif

struct [**dev\_msi\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_msi_info) [**msi**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/msi);

#ifdef [**CONFIG\_DMA\_OPS**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_DMA_OPS)

const struct [**dma\_map\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_map_ops) \*[**dma\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_ops);

#endif

[**u64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u64) \*[**dma\_mask**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_mask); */\* dma mask (if dma'able device) \*/*

[**u64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u64) [**coherent\_dma\_mask**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/coherent_dma_mask);*/\* Like dma\_mask, but for*

*alloc\_coherent mappings as*

*not all hardware supports*

*64 bit addresses for consistent*

*allocations such descriptors. \*/*

[**u64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u64) [**bus\_dma\_limit**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bus_dma_limit); */\* upstream dma constraint \*/*

const struct [**bus\_dma\_region**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bus_dma_region) \*[**dma\_range\_map**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_range_map);

struct [**device\_dma\_parameters**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device_dma_parameters) \*[**dma\_parms**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_parms);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**dma\_pools**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_pools); */\* dma pools (if dma'ble) \*/*

#ifdef [**CONFIG\_DMA\_DECLARE\_COHERENT**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_DMA_DECLARE_COHERENT)

struct [**dma\_coherent\_mem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_coherent_mem) \*[**dma\_mem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_mem); */\* internal for coherent mem*

*override \*/*

#endif

#ifdef [**CONFIG\_DMA\_CMA**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_DMA_CMA)

struct [**cma**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/cma) \*[**cma\_area**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/cma_area); */\* contiguous memory area for dma*

*allocations \*/*

#endif

#ifdef [**CONFIG\_SWIOTLB**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_SWIOTLB)

struct [**io\_tlb\_mem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/io_tlb_mem) \*[**dma\_io\_tlb\_mem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_io_tlb_mem);

#endif

*/\* arch specific additions \*/*

struct [**dev\_archdata**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_archdata) [**archdata**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/archdata);

struct [**device\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device_node) \*[**of\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/of_node); */\* associated device tree node \*/*

struct [**fwnode\_handle**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fwnode_handle) \*[**fwnode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fwnode); */\* firmware device node \*/*

#ifdef [**CONFIG\_NUMA**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_NUMA)

int [**numa\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/numa_node); */\* NUMA node this device is close to \*/*

#endif

[**dev\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_t) [**devt**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/devt); */\* dev\_t, creates the sysfs "dev" \*/*

[**u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u32) id; */\* device instance \*/*

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) [**devres\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/devres_lock);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**devres\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/devres_head);

struct [**class**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/class) \*[**class**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/class);

const struct [**attribute\_group**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/attribute_group) \*\*[**groups**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/groups); */\* optional groups \*/*

void (\*[**release**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/release))(struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device) \*dev);

struct [**iommu\_group**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iommu_group) \*[**iommu\_group**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iommu_group);

struct [**dev\_iommu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_iommu) \*[**iommu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iommu);

enum [**device\_removable**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device_removable) [**removable**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/removable);

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**offline\_disabled**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/offline_disabled):1;

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**offline**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/offline):1;

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**of\_node\_reused**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/of_node_reused):1;

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**state\_synced**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/state_synced):1;

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**can\_match**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/can_match):1;

#if [**defined**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/defined)([**CONFIG\_ARCH\_HAS\_SYNC\_DMA\_FOR\_DEVICE**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_ARCH_HAS_SYNC_DMA_FOR_DEVICE)) || \

[**defined**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/defined)([**CONFIG\_ARCH\_HAS\_SYNC\_DMA\_FOR\_CPU**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_ARCH_HAS_SYNC_DMA_FOR_CPU)) || \

[**defined**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/defined)([**CONFIG\_ARCH\_HAS\_SYNC\_DMA\_FOR\_CPU\_ALL**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_ARCH_HAS_SYNC_DMA_FOR_CPU_ALL))

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**dma\_coherent**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_coherent):1;

#endif

#ifdef [**CONFIG\_DMA\_OPS\_BYPASS**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_DMA_OPS_BYPASS)

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**dma\_ops\_bypass**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dma_ops_bypass) : 1;

#endif

};

struct [**device\_driver**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device_driver) {

const char \*name;

struct [**bus\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/bus_type) \*[**bus**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/bus);

struct [**module**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/module) \*[**owner**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/owner);

const char \*[**mod\_name**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/mod_name); */\* used for built-in modules \*/*

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/bool) [**suppress\_bind\_attrs**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/suppress_bind_attrs); */\* disables bind/unbind via sysfs \*/*

enum [**probe\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/probe_type) [**probe\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/probe_type);

const struct [**of\_device\_id**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/of_device_id) \*[**of\_match\_table**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/of_match_table);

const struct [**acpi\_device\_id**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/acpi_device_id) \*[**acpi\_match\_table**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/acpi_match_table);

int (\*[**probe**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/probe)) (struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev);

void (\*[**sync\_state**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/sync_state))(struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev);

int (\*[**remove**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/remove)) (struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev);

void (\*[**shutdown**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/shutdown)) (struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev);

int (\*[**suspend**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/suspend)) (struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev, [**pm\_message\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/pm_message_t) state);

int (\*[**resume**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/resume)) (struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev);

const struct [**attribute\_group**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/attribute_group) \*\*[**groups**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/groups);

const struct [**attribute\_group**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/attribute_group) \*\*[**dev\_groups**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/dev_groups);

const struct [**dev\_pm\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/dev_pm_ops) \*[**pm**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/pm);

void (\*[**coredump**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/coredump)) (struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/device) \*dev);

struct [**driver\_private**](https://elixir.bootlin.com/linux/latest/source/include/linux/device/latest/C/ident/driver_private) \*p;

};

#define [**MAJOR**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/MAJOR)(dev) ((unsigned int) ((dev) >> [**MINORBITS**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/MINORBITS)))

#define [**MINOR**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/MINOR)(dev) ((unsigned int) ((dev) & [**MINORMASK**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/MINORMASK)))

#define [**MKDEV**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/MKDEV)([**ma**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ma),[**mi**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mi)) ((([**ma**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ma)) << [**MINORBITS**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/MINORBITS)) | ([**mi**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mi)))

*\* @from: the first in the desired range of device numbers; must include* *the major number.*

*\* @count: the number of consecutive device numbers required*

*\* @name: the name of the device or driver.*

*\* Return value is zero on success, a negative error code on failure.*

int [**register\_chrdev\_region**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/register_chrdev_region)([**dev\_t**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/dev_t) [**from**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/from), unsigned count, const char \*name)

*\* @dev: output parameter for first assigned number*

*\* @baseminor: first of the requested range of minor numbers*

*\* @count: the number of minor numbers required*

*\* @name: the name of the associated device or driver*

int [**alloc\_chrdev\_region**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/alloc_chrdev_region)([**dev\_t**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/dev_t) \*dev, unsigned [**baseminor**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/baseminor), unsigned count,

const char \*name)

*\* @from: the first in the range of numbers to unregister*

*\* @count: the number of device numbers to unregister*

*\**

*\* This function will unregister a range of @count device numbers,*

*\* starting with @from. The caller should normally be the one who*

*\* allocated those numbers in the first place...*

void [**unregister\_chrdev\_region**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/unregister_chrdev_region)([**dev\_t**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/dev_t) [**from**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/from), unsigned count)

[**request\_irq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/request_irq)(unsigned int irq, [**irq\_handler\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/irq_handler_t) [**handler**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/handler), unsigned long flags,

const char \*name, void \*dev)

const void \*[**free\_irq**](https://elixir.bootlin.com/linux/latest/source/kernel/irq/latest/C/ident/free_irq)(unsigned int irq, void \*[**dev\_id**](https://elixir.bootlin.com/linux/latest/source/kernel/irq/latest/C/ident/dev_id))

*# IRQF\_SHARED - allow sharing the irq among several devices*

struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct)

{

struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) \*next;

unsigned long state;

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) count;

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**use\_callback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/use_callback);

union {

void (\*[**func**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/func))(unsigned long data);

void (\*[**callback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/callback))(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) \*t);

};

unsigned long data;}

#define [**DECLARE\_TASKLET**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/DECLARE_TASKLET)(name, [**\_callback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/_callback)) \

struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) name = { \

.count = [**ATOMIC\_INIT**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ATOMIC_INIT)(0), \

.[**callback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/callback) = [**\_callback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/_callback), \

.[**use\_callback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/use_callback) = [**true**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/true), \

}

enum

{

[**TASKLET\_STATE\_SCHED**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/TASKLET_STATE_SCHED), */\* Tasklet is scheduled for execution \*/*

[**TASKLET\_STATE\_RUN**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/TASKLET_STATE_RUN) */\* Tasklet is running (SMP only) \*/*

};

static inline void [**tasklet\_schedule**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_schedule)(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) \*t);

static inline void [**tasklet\_hi\_schedule**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_hi_schedule)(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) \*t);

static inline void [**tasklet\_disable**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_disable)(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) \*t);

static inline void [**tasklet\_enable**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_enable)(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasklet_struct) \*t);

void [**tasklet\_init**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/tasklet_init)(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/tasklet_struct) \*t, void (\*[**func**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/func))(unsigned long), unsigned long data);

void [**tasklet\_kill**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/tasklet_kill)(struct [**tasklet\_struct**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/tasklet_struct) \*t);

struct [**block\_device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/block_device) {

[**sector\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sector_t) [**bd\_start\_sect**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_start_sect);

[**sector\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sector_t) [**bd\_nr\_sectors**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_nr_sectors);

struct [**disk\_stats**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/disk_stats) [**\_\_percpu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__percpu) \*[**bd\_stats**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_stats);

unsigned long [**bd\_stamp**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_stamp);

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**bd\_read\_only**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_read_only); */\* read-only policy \*/*

[**dev\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_t) [**bd\_dev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_dev);

int [**bd\_openers**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_openers);

struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \* [**bd\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_inode); */\* will die \*/*

struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \* [**bd\_super**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_super);

void \* [**bd\_claiming**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_claiming);

struct [**device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/device) [**bd\_device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_device);

void \* [**bd\_holder**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_holder);

int [**bd\_holders**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_holders);

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**bd\_write\_holder**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_write_holder);

struct [**kobject**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kobject) \*[**bd\_holder\_dir**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_holder_dir);

[**u8**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u8) [**bd\_partno**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_partno);

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) [**bd\_size\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_size_lock); */\* for bd\_inode->i\_size updates \*/*

struct [**gendisk**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/gendisk) \* [**bd\_disk**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_disk);

struct [**request\_queue**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/request_queue) \* [**bd\_queue**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_queue);

*/\* The counter of freeze processes \*/*

int [**bd\_fsfreeze\_count**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_fsfreeze_count);

*/\* Mutex for freeze \*/*

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex) [**bd\_fsfreeze\_mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_fsfreeze_mutex);

struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*[**bd\_fsfreeze\_sb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_fsfreeze_sb);

struct [**partition\_meta\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/partition_meta_info) \*[**bd\_meta\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_meta_info);

#ifdef [**CONFIG\_FAIL\_MAKE\_REQUEST**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FAIL_MAKE_REQUEST)

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**bd\_make\_it\_fail**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bd_make_it_fail);

#endif

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

struct [**cdev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/cdev) {

struct [**kobject**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kobject) [**kobj**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kobj);

struct [**module**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/module) \*[**owner**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/owner);

const struct [**file\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_operations) \*ops;

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) list;

[**dev\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_t) dev;

unsigned int count;

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) {

[**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t) [**i\_mode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_mode); // mode

unsigned [**short**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/short) [**i\_opflags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_opflags);

[**kuid\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kuid_t) [**i\_uid**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_uid); // uid

[**kgid\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kgid_t) [**i\_gid**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_gid); // gid

unsigned int [**i\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_flags);

#ifdef [**CONFIG\_FS\_POSIX\_ACL**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FS_POSIX_ACL)

struct [**posix\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/posix_acl) \*[**i\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_acl);

struct [**posix\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/posix_acl) \*[**i\_default\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_default_acl);

#endif

const struct [**inode\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode_operations) \*[**i\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_op);

struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*[**i\_sb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_sb); // sb

struct [**address\_space**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/address_space) \*[**i\_mapping**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_mapping);

#ifdef [**CONFIG\_SECURITY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_SECURITY)

void \*[**i\_security**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_security);

#endif

*/\* Stat data, not accessed from path walking \*/*

unsigned long [**i\_ino**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_ino);

*/\**

*\* Filesystems may only read i\_nlink directly. They shall use the*

*\* following functions for modification:*

*\**

*\* (set|clear|inc|drop)\_nlink*

*\* inode\_(inc|dec)\_link\_count*

*\*/*

union {

const unsigned int [**i\_nlink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_nlink);

unsigned int [**\_\_i\_nlink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__i_nlink);

};

[**dev\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_t) [**i\_rdev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_rdev);

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) [**i\_size**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_size);

struct [**timespec64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/timespec64) [**i\_atime**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_atime);

struct [**timespec64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/timespec64) [**i\_mtime**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_mtime);

struct [**timespec64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/timespec64) [**i\_ctime**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_ctime);

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) [**i\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_lock); */\* i\_blocks, i\_bytes, maybe i\_size \*/*

unsigned [**short**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/short) [**i\_bytes**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_bytes);

[**u8**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u8) [**i\_blkbits**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_blkbits);

[**u8**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u8) [**i\_write\_hint**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_write_hint);

[**blkcnt\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/blkcnt_t) [**i\_blocks**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_blocks);

#ifdef [**\_\_NEED\_I\_SIZE\_ORDERED**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__NEED_I_SIZE_ORDERED)

[**seqcount\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seqcount_t) [**i\_size\_seqcount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_size_seqcount);

#endif

*/\* Misc \*/*

unsigned long [**i\_state**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_state);

struct [**rw\_semaphore**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rw_semaphore) [**i\_rwsem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_rwsem);

unsigned long [**dirtied\_when**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dirtied_when); */\* jiffies of first dirtying \*/*

unsigned long [**dirtied\_time\_when**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dirtied_time_when);

struct [**hlist\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_node) [**i\_hash**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_hash);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**i\_io\_list**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_io_list); */\* backing dev IO list \*/*

#ifdef [**CONFIG\_CGROUP\_WRITEBACK**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_CGROUP_WRITEBACK)

struct [**bdi\_writeback**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bdi_writeback) \*[**i\_wb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_wb); */\* the associated cgroup wb \*/*

*/\* foreign inode detection, see wbc\_detach\_inode() \*/*

int [**i\_wb\_frn\_winner**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_wb_frn_winner);

[**u16**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u16) [**i\_wb\_frn\_avg\_time**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_wb_frn_avg_time);

[**u16**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u16) [**i\_wb\_frn\_history**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_wb_frn_history);

#endif

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**i\_lru**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_lru); */\* inode LRU list \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**i\_sb\_list**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_sb_list);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**i\_wb\_list**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_wb_list); */\* backing dev writeback list \*/*

union {

struct [**hlist\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_head) [**i\_dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_dentry);

struct [**rcu\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rcu_head) [**i\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_rcu);

};

[**atomic64\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic64_t) [**i\_version**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_version);

[**atomic64\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic64_t) [**i\_sequence**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_sequence); */\* see futex \*/*

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) [**i\_count**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_count);

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) [**i\_dio\_count**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_dio_count);

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) [**i\_writecount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_writecount);

#if [**defined**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/defined)([**CONFIG\_IMA**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_IMA)) || [**defined**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/defined)([**CONFIG\_FILE\_LOCKING**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FILE_LOCKING))

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) [**i\_readcount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_readcount); */\* struct files open RO \*/*

#endif

union {

const struct [**file\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_operations) \*[**i\_fop**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_fop); */\* former ->i\_op->default\_file\_ops \*/*

void (\*[**free\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/free_inode))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

};

struct [**file\_lock\_context**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_lock_context) \*[**i\_flctx**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_flctx);

struct [**address\_space**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/address_space) [**i\_data**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_data);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**i\_devices**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_devices);

union {

struct [**pipe\_inode\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pipe_inode_info) \*[**i\_pipe**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_pipe);

struct [**cdev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/cdev) \*[**i\_cdev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_cdev);

char \*[**i\_link**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_link);

unsigned [**i\_dir\_seq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_dir_seq);

};

[**\_\_u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__u32) [**i\_generation**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_generation);

#ifdef [**CONFIG\_FSNOTIFY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FSNOTIFY)

[**\_\_u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__u32) [**i\_fsnotify\_mask**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_fsnotify_mask); */\* all events this inode cares about \*/*

struct [**fsnotify\_mark\_connector**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fsnotify_mark_connector) [**\_\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__rcu) \*[**i\_fsnotify\_marks**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_fsnotify_marks);

#endif

#ifdef [**CONFIG\_FS\_ENCRYPTION**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FS_ENCRYPTION)

struct [**fscrypt\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fscrypt_info) \*[**i\_crypt\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_crypt_info);

#endif

#ifdef [**CONFIG\_FS\_VERITY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FS_VERITY)

struct [**fsverity\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fsverity_info) \*[**i\_verity\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_verity_info);

#endif

void \*[**i\_private**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_private); */\* fs or device private pointer \*/*

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

struct [**workqueue\_struct**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/workqueue_struct) {

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/list_head) [**pwqs**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/pwqs); */\* WR: all pwqs of this wq \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/list_head) list; */\* PR: list of all workqueues \*/*

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/mutex) [**mutex**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/mutex); */\* protects this wq \*/*

int [**work\_color**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/work_color); */\* WQ: current work color \*/*

int [**flush\_color**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/flush_color); */\* WQ: current flush color \*/*

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/atomic_t) [**nr\_pwqs\_to\_flush**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/nr_pwqs_to_flush); */\* flush in progress \*/*

struct [**wq\_flusher**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/wq_flusher) \*[**first\_flusher**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/first_flusher); */\* WQ: first flusher \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/list_head) [**flusher\_queue**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/flusher_queue); */\* WQ: flush waiters \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/list_head) [**flusher\_overflow**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/flusher_overflow); */\* WQ: flush overflow list \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/list_head) [**maydays**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/maydays); */\* MD: pwqs requesting rescue \*/*

struct [**worker**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/worker) \*[**rescuer**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/rescuer); */\* MD: rescue worker \*/*

int [**nr\_drainers**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/nr_drainers); */\* WQ: drain in progress \*/*

int [**saved\_max\_active**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/saved_max_active); */\* WQ: saved pwq max\_active \*/*

struct [**workqueue\_attrs**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/workqueue_attrs) \*[**unbound\_attrs**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/unbound_attrs); */\* PW: only for unbound wqs \*/*

struct [**pool\_workqueue**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/pool_workqueue) \*[**dfl\_pwq**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/dfl_pwq); */\* PW: only for unbound wqs \*/*

#ifdef [**CONFIG\_SYSFS**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/K/ident/CONFIG_SYSFS)

struct [**wq\_device**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/wq_device) \*[**wq\_dev**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/wq_dev); */\* I: for sysfs interface \*/*

#endif

#ifdef [**CONFIG\_LOCKDEP**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/K/ident/CONFIG_LOCKDEP)

char \*[**lock\_name**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/lock_name);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/lock_class_key) [**key**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/key);

struct [**lockdep\_map**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/lockdep_map) [**lockdep\_map**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/lockdep_map);

#endif

char name[[**WQ\_NAME\_LEN**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/WQ_NAME_LEN)]; */\* I: workqueue name \*/*

*/\**

*\* Destruction of workqueue\_struct is RCU protected to allow walking*

*\* the workqueues list without grabbing wq\_pool\_mutex.*

*\* This is used to dump all workqueues from sysrq.*

*\*/*

struct [**rcu\_head**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/rcu_head) [**rcu**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/rcu);

*/\* hot fields used during command issue, aligned to cacheline \*/*

unsigned int flags [**\_\_\_\_cacheline\_aligned**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/____cacheline_aligned); */\* WQ: WQ\_\* flags \*/*

struct [**pool\_workqueue**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/pool_workqueue) [**\_\_percpu**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/__percpu) \*[**cpu\_pwqs**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/cpu_pwqs); */\* I: per-cpu pwqs \*/*

struct [**pool\_workqueue**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/pool_workqueue) [**\_\_rcu**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/__rcu) \*[**numa\_pwq\_tbl**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/numa_pwq_tbl)[]; */\* PWR: unbound pwqs indexed by node \*/*

};

struct [**work\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/work_struct) {

[**atomic\_long\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_long_t) data;

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) entry;

[**work\_func\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/work_func_t) [**func**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/func);

#ifdef [**CONFIG\_LOCKDEP**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_LOCKDEP)

struct [**lockdep\_map**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lockdep_map) [**lockdep\_map**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lockdep_map);

#endif

};

#define [**DECLARE\_WORK**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/DECLARE_WORK)(name, void (\*f)(void \*))

#define [**INIT\_WORK**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/INIT_WORK)(struct work\_struct \*work, void (\*f)(void), void \*)

static inline [**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**queue\_work**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/queue_work)(struct [**workqueue\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/workqueue_struct) \*[**wq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wq), struct [**work\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/work_struct) \*[**work**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/work));

void [**flush\_workqueue**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/flush_workqueue)(struct [**workqueue\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/workqueue_struct) \*[**wq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wq));

extern void [**destroy\_workqueue**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/destroy_workqueue)(struct [**workqueue\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/workqueue_struct) \*[**wq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wq));

void [**open\_softirq**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/open_softirq)(int [**nr**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/nr), void (\*[**action**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/action))(struct [**softirq\_action**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/softirq_action) \*))

{

[**softirq\_vec**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/softirq_vec)[[**nr**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/nr)].[**action**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/action) = [**action**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/action);

}

void [**raise\_softirq**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/raise_softirq)(unsigned int [**nr**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/nr))

{

unsigned long flags;

[**local\_irq\_save**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/local_irq_save)(flags);

[**raise\_softirq\_irqoff**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/raise_softirq_irqoff)([**nr**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/nr));

[**local\_irq\_restore**](https://elixir.bootlin.com/linux/latest/source/kernel/latest/C/ident/local_irq_restore)(flags);

}

struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_dir_entry) {

*/\**

*\* number of callers into module in progress;*

*\* negative -> it's going away RSN*

*\*/*

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/atomic_t) [**in\_use**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/in_use);

[**refcount\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/refcount_t) [**refcnt**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/refcnt);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/list_head) [**pde\_openers**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/pde_openers); */\* who did ->open, but not ->release \*/*

*/\* protects ->pde\_openers and all struct pde\_opener instances \*/*

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/spinlock_t) [**pde\_unload\_lock**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/pde_unload_lock);

struct [**completion**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/completion) \*[**pde\_unload\_completion**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/pde_unload_completion);

const struct [**inode\_operations**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/inode_operations) \*[**proc\_iops**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_iops);

union {

const struct [**proc\_ops**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_ops) \*[**proc\_ops**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_ops);

const struct [**file\_operations**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/file_operations) \*[**proc\_dir\_ops**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_dir_ops);

};

const struct [**dentry\_operations**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/dentry_operations) \*[**proc\_dops**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_dops);

union {

const struct [**seq\_operations**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/seq_operations) \*[**seq\_ops**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/seq_ops);

int (\*[**single\_show**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/single_show))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/seq_file) \*, void \*);

};

[**proc\_write\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_write_t) [**write**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/write);

void \*data;

unsigned int [**state\_size**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/state_size);

unsigned int [**low\_ino**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/low_ino);

[**nlink\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/nlink_t) [**nlink**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/nlink);

[**kuid\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/kuid_t) [**uid**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/uid);

[**kgid\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/kgid_t) [**gid**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/gid);

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/loff_t) size;

struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/proc_dir_entry) \*[**parent**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/parent);

struct [**rb\_root**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/rb_root) [**subdir**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/subdir);

struct [**rb\_node**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/rb_node) [**subdir\_node**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/subdir_node);

char \*name;

[**umode\_t**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/umode_t) mode;

[**u8**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/u8) flags;

[**u8**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/u8) [**namelen**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/namelen);

char [**inline\_name**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/inline_name)[];

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/fs/proc/latest/C/ident/__randomize_layout);

struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*[**proc\_create**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_create)(const char \*name, [**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t) mode, struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*[**parent**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/parent), const struct [**proc\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_ops) \*[**proc\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_ops));

extern struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*[**proc\_mkdir**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_mkdir)(const char \*, struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*);

extern struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*[**proc\_symlink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_symlink)(const char \*,

struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*, const char \*);

extern void [**remove\_proc\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/remove_proc_entry)(const char \*, struct [**proc\_dir\_entry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proc_dir_entry) \*);

struct [**file\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_operations) {

struct [**module**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/module) \*[**owner**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/owner);

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) (\*[**llseek**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/llseek)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), int);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**read**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/read)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, char [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) \*);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**write**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/write)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, const char [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) \*);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**read\_iter**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/read_iter)) (struct [**kiocb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kiocb) \*, struct [**iov\_iter**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iov_iter) \*);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**write\_iter**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/write_iter)) (struct [**kiocb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kiocb) \*, struct [**iov\_iter**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iov_iter) \*);

int (\*[**iopoll**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iopoll))(struct [**kiocb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kiocb) \*[**kiocb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kiocb), struct [**io\_comp\_batch**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/io_comp_batch) \*,

unsigned int flags);

int (\*[**iterate**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iterate)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, struct [**dir\_context**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dir_context) \*);

int (\*[**iterate\_shared**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iterate_shared)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, struct [**dir\_context**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dir_context) \*);

[**\_\_poll\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__poll_t) (\*[**poll**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/poll)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, struct [**poll\_table\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/poll_table_struct) \*);

long (\*[**unlocked\_ioctl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/unlocked_ioctl)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, unsigned int, unsigned long);

long (\*[**compat\_ioctl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/compat_ioctl)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, unsigned int, unsigned long);

int (\*[**mmap**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mmap)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, struct [**vm\_area\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/vm_area_struct) \*);

unsigned long [**mmap\_supported\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mmap_supported_flags);

int (\*[**open**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/open)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*);

int (\*[**flush**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/flush)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**fl\_owner\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fl_owner_t) id);

int (\*[**release**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/release)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*);

int (\*[**fsync**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fsync)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), int [**datasync**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/datasync));

int (\*[**fasync**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fasync)) (int, struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, int);

int (\*lock) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, int, struct [**file\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_lock) \*);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**sendpage**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sendpage)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, struct [**page**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/page) \*, int, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) \*, int);

unsigned long (\*[**get\_unmapped\_area**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/get_unmapped_area))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, unsigned long, unsigned long, unsigned long, unsigned long);

int (\*[**check\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/check_flags))(int);

int (\*[**flock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/flock)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, int, struct [**file\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_lock) \*);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**splice\_write**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/splice_write))(struct [**pipe\_inode\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pipe_inode_info) \*, struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), unsigned int);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**splice\_read**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/splice_read))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) \*, struct [**pipe\_inode\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pipe_inode_info) \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), unsigned int);

int (\*[**setlease**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/setlease))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, long, struct [**file\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_lock) \*\*, void \*\*);

long (\*[**fallocate**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fallocate))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*[**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file), int mode, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) offset,

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) len);

void (\*[**show\_fdinfo**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/show_fdinfo))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq_file) \*m, struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*f);

#ifndef [**CONFIG\_MMU**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_MMU)

unsigned (\*[**mmap\_capabilities**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mmap_capabilities))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*);

#endif

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**copy\_file\_range**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/copy_file_range))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*,

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), unsigned int);

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) (\*[**remap\_file\_range**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/remap_file_range))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*[**file\_in**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_in), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) [**pos\_in**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pos_in),

struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*[**file\_out**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_out), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) [**pos\_out**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pos_out),

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) len, unsigned int remap\_flags);

int (\*[**fadvise**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fadvise))(struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t), int);

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

unsigned long [**copy\_to\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/copy_to_user)(void [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*[**to**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/to), const void \*[**from**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/from), unsigned long n);

struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) {

const char \*name; */\*= name \*/*

int [**fs\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_flags);

#define [**FS\_REQUIRES\_DEV**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_REQUIRES_DEV) 1

#define [**FS\_BINARY\_MOUNTDATA**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_BINARY_MOUNTDATA) 2

#define [**FS\_HAS\_SUBTYPE**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_HAS_SUBTYPE) 4

#define [**FS\_USERNS\_MOUNT**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_USERNS_MOUNT) 8 */\* Can be mounted by userns root \*/*

#define [**FS\_DISALLOW\_NOTIFY\_PERM**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_DISALLOW_NOTIFY_PERM) 16 */\* Disable fanotify permission events \*/*

#define [**FS\_ALLOW\_IDMAP**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_ALLOW_IDMAP) 32 */\* FS has been updated to handle vfs idmappings. \*/*

#define [**FS\_RENAME\_DOES\_D\_MOVE**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/FS_RENAME_DOES_D_MOVE) 32768 */\* FS will handle d\_move() during rename() internally. \*/*

int (\*[**init\_fs\_context**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/init_fs_context))(struct [**fs\_context**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_context) \*);

const struct [**fs\_parameter\_spec**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_parameter_spec) \*[**parameters**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/parameters);

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*(\*[**mount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mount)) (struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \*, int,

const char \*, void \*); */\* = mount\*/*

void (\*[**kill\_sb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kill_sb)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*); */\* = killsb \*/*

struct [**module**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/module) \*[**owner**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/owner); */\* = this module \*/*

struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \* next;

struct [**hlist\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_head) [**fs\_supers**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_supers);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**s\_lock\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_lock_key);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**s\_umount\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_umount_key);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**s\_vfs\_rename\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_vfs_rename_key);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**s\_writers\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_writers_key)[[**SB\_FREEZE\_LEVELS**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/SB_FREEZE_LEVELS)];

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**i\_lock\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_lock_key);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**i\_mutex\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_mutex_key);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**invalidate\_lock\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/invalidate_lock_key);

struct [**lock\_class\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lock_class_key) [**i\_mutex\_dir\_key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/i_mutex_dir_key);

};

struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) {

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**s\_list**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_list); */\* Keep this first \*/*

[**dev\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_t) [**s\_dev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_dev); */\* search index; \_not\_ kdev\_t \*/*

unsigned char [**s\_blocksize\_bits**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_blocksize_bits); */\* = page\_shift \*/*

unsigned long [**s\_blocksize**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_blocksize); */\* = page\_size \*/*

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) [**s\_maxbytes**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_maxbytes); */\* Max file size \*/*

struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \*[**s\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_type); // type

const struct [**super\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_operations) \*[**s\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_op); */\* = ops \*/*

const struct [**dquot\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dquot_operations) \*[**dq\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dq_op);

const struct [**quotactl\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/quotactl_ops) \*[**s\_qcop**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_qcop);

const struct [**export\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/export_operations) \*[**s\_export\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_export_op);

unsigned long [**s\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_flags);

unsigned long [**s\_iflags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_iflags); */\* internal SB\_I\_\* flags \*/*

unsigned long [**s\_magic**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_magic); */\*= magic\*/*

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*[**s\_root**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_root); */\*= root \*/*

struct [**rw\_semaphore**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rw_semaphore) [**s\_umount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_umount);

int [**s\_count**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_count);

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) [**s\_active**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_active);

#ifdef [**CONFIG\_SECURITY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_SECURITY)

void \*[**s\_security**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_security);

#endif

const struct [**xattr\_handler**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/xattr_handler) \*\*[**s\_xattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_xattr);

#ifdef [**CONFIG\_FS\_ENCRYPTION**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FS_ENCRYPTION)

const struct [**fscrypt\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fscrypt_operations) \*[**s\_cop**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_cop);

struct [**key**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/key) \*[**s\_master\_keys**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_master_keys); */\* master crypto keys in use \*/*

#endif

#ifdef [**CONFIG\_FS\_VERITY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FS_VERITY)

const struct [**fsverity\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fsverity_operations) \*[**s\_vop**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_vop);

#endif

#if [**IS\_ENABLED**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/IS_ENABLED)([**CONFIG\_UNICODE**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_UNICODE))

struct [**unicode\_map**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/unicode_map) \*[**s\_encoding**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_encoding);

[**\_\_u16**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__u16) [**s\_encoding\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_encoding_flags);

#endif

struct [**hlist\_bl\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_bl_head) [**s\_roots**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_roots); */\* alternate root dentries for NFS \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**s\_mounts**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_mounts); */\* list of mounts; \_not\_ for fs use \*/*

struct [**block\_device**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/block_device) \*[**s\_bdev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_bdev);

struct [**backing\_dev\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/backing_dev_info) \*[**s\_bdi**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_bdi);

struct [**mtd\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mtd_info) \*[**s\_mtd**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_mtd);

struct [**hlist\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_node) [**s\_instances**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_instances);

unsigned int [**s\_quota\_types**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_quota_types); */\* Bitmask of supported quota types \*/*

struct [**quota\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/quota_info) [**s\_dquot**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_dquot); */\* Diskquota specific options \*/*

struct [**sb\_writers**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sb_writers) [**s\_writers**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_writers);

*/\**

*\* Keep s\_fs\_info, s\_time\_gran, s\_fsnotify\_mask, and*

*\* s\_fsnotify\_marks together for cache efficiency. They are frequently*

*\* accessed and rarely modified.*

*\*/*

void \*[**s\_fs\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_fs_info); */\* Filesystem private info \*/*

*/\* Granularity of c/m/atime in ns (cannot be worse than a second) \*/*

[**u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u32) [**s\_time\_gran**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_time_gran);

*/\* Time limits for c/m/atime in seconds \*/*

[**time64\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/time64_t) [**s\_time\_min**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_time_min);

[**time64\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/time64_t) [**s\_time\_max**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_time_max);

#ifdef [**CONFIG\_FSNOTIFY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_FSNOTIFY)

[**\_\_u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__u32) [**s\_fsnotify\_mask**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_fsnotify_mask);

struct [**fsnotify\_mark\_connector**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fsnotify_mark_connector) [**\_\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__rcu) \*[**s\_fsnotify\_marks**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_fsnotify_marks);

#endif

char [**s\_id**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_id)[32]; */\* Informational name \*/*

[**uuid\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/uuid_t) [**s\_uuid**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_uuid); */\* UUID \*/*

unsigned int [**s\_max\_links**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_max_links);

[**fmode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fmode_t) [**s\_mode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_mode);

*/\**

*\* The next field is for VFS \*only\*. No filesystems have any business*

*\* even looking at it. You had been warned.*

*\*/*

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex) [**s\_vfs\_rename\_mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_vfs_rename_mutex); */\* Kludge \*/*

*/\**

*\* Filesystem subtype. If non-empty the filesystem type field*

*\* in /proc/mounts will be "type.subtype"*

*\*/*

const char \*[**s\_subtype**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_subtype);

const struct [**dentry\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry_operations) \*[**s\_d\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_d_op); */\* default d\_op for dentries \*/*

struct [**shrinker**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/shrinker) [**s\_shrink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_shrink); */\* per-sb shrinker handle \*/*

*/\* Number of inodes with nlink == 0 but still referenced \*/*

[**atomic\_long\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_long_t) [**s\_remove\_count**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_remove_count);

*/\**

*\* Number of inode/mount/sb objects that are being watched, note that*

*\* inodes objects are currently double-accounted.*

*\*/*

[**atomic\_long\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_long_t) [**s\_fsnotify\_connectors**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_fsnotify_connectors);

*/\* Being remounted read-only \*/*

int [**s\_readonly\_remount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_readonly_remount);

*/\* per-sb errseq\_t for reporting writeback errors via syncfs \*/*

[**errseq\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/errseq_t) [**s\_wb\_err**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_wb_err);

*/\* AIO completions deferred from interrupt context \*/*

struct [**workqueue\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/workqueue_struct) \*[**s\_dio\_done\_wq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_dio_done_wq);

struct [**hlist\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_head) [**s\_pins**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_pins);

*/\**

*\* Owning user namespace and default context in which to*

*\* interpret filesystem uids, gids, quotas, device nodes,*

*\* xattrs and security labels.*

*\*/*

struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*[**s\_user\_ns**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_user_ns);

*/\**

*\* The list\_lru structure is essentially just a pointer to a table*

*\* of per-node lru lists, each of which has its own spinlock.*

*\* There is no need to put them into separate cachelines.*

*\*/*

struct [**list\_lru**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_lru) [**s\_dentry\_lru**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_dentry_lru);

struct [**list\_lru**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_lru) [**s\_inode\_lru**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_inode_lru);

struct [**rcu\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rcu_head) [**rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rcu);

struct [**work\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/work_struct) [**destroy\_work**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/destroy_work);

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex) [**s\_sync\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_sync_lock); */\* sync serialisation lock \*/*

*/\**

*\* Indicates how deep in a filesystem stack this SB is*

*\*/*

int [**s\_stack\_depth**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_stack_depth);

*/\* s\_inode\_list\_lock protects s\_inodes \*/*

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) s\_inode\_list\_lock [**\_\_\_\_cacheline\_aligned\_in\_smp**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/____cacheline_aligned_in_smp);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**s\_inodes**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_inodes); */\* all inodes \*/*

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) [**s\_inode\_wblist\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_inode_wblist_lock);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**s\_inodes\_wb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/s_inodes_wb); */\* writeback inodes \*/*

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) {

*/\* RCU lookup touched fields \*/*

unsigned int [**d\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_flags); */\* protected by d\_lock \*/*

seqcount\_spinlock\_t [**d\_seq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_seq); */\* per dentry seqlock \*/*

struct [**hlist\_bl\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_bl_node) [**d\_hash**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_hash); */\* lookup hash list \*/*

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*[**d\_parent**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_parent); */\* parent directory \*/*

struct [**qstr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/qstr) [**d\_name**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_name);

struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*[**d\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_inode); */\* Where the name belongs to - NULL is*

*\* negative \*/*

unsigned char [**d\_iname**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_iname)[[**DNAME\_INLINE\_LEN**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/DNAME_INLINE_LEN)]; */\* small names \*/*

*/\* Ref lookup also touches following \*/*

struct [**lockref**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lockref) [**d\_lockref**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_lockref); */\* per-dentry lock and refcount \*/*

const struct [**dentry\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry_operations) \*[**d\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_op);

struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*[**d\_sb**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_sb); */\* The root of the dentry tree \*/*

unsigned long [**d\_time**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_time); */\* used by d\_revalidate \*/*

void \*[**d\_fsdata**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_fsdata); */\* fs-specific data \*/*

union {

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**d\_lru**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_lru); */\* LRU list \*/*

[**wait\_queue\_head\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wait_queue_head_t) \*[**d\_wait**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_wait); */\* in-lookup ones only \*/*

};

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**d\_child**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_child); */\* child of parent list \*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**d\_subdirs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_subdirs); */\* our children \*/*

*/\**

*\* d\_alias and d\_rcu can share memory*

*\*/*

union {

struct [**hlist\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_node) [**d\_alias**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_alias); */\* inode alias list \*/*

struct [**hlist\_bl\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_bl_node) [**d\_in\_lookup\_hash**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_in_lookup_hash); */\* only for in-lookup ones \*/*

struct [**rcu\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rcu_head) [**d\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_rcu);

} [**d\_u**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_u);

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) {

union {

struct [**llist\_node**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/llist_node) [**fu\_llist**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fu_llist);

struct [**rcu\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rcu_head) [**fu\_rcuhead**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fu_rcuhead);

} [**f\_u**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_u);

struct [**path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/path) [**f\_path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_path); // path

struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*[**f\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_inode); */\* cached value: inode \*/*

const struct [**file\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_operations) \*[**f\_op**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_op);

*/\**

*\* Protects f\_ep, f\_flags.*

*\* Must not be taken from IRQ context.*

*\*/*

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) [**f\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_lock);

[**atomic\_long\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_long_t) [**f\_count**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_count);

unsigned int [**f\_flags**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_flags);

[**fmode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fmode_t) [**f\_mode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_mode); // mode

struct [**mutex**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mutex) [**f\_pos\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_pos_lock);

[**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) [**f\_pos**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_pos); // pos

struct [**fown\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fown_struct) [**f\_owner**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_owner);

const struct [**cred**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/cred) \*[**f\_cred**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_cred);

struct [**file\_ra\_state**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_ra_state) [**f\_ra**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_ra);

[**u64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u64) [**f\_version**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_version);

#ifdef [**CONFIG\_SECURITY**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_SECURITY)

void \*[**f\_security**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_security);

#endif

*/\* needed for tty driver, and maybe others \*/*

void \*[**private\_data**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/private_data);

#ifdef [**CONFIG\_EPOLL**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_EPOLL)

*/\* Used by fs/eventpoll.c to link all the hooks to this file \*/*

struct [**hlist\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/hlist_head) \*[**f\_ep**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_ep);

#endif */\* #ifdef CONFIG\_EPOLL \*/*

struct [**address\_space**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/address_space) \*[**f\_mapping**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_mapping);

[**errseq\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/errseq_t) [**f\_wb\_err**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_wb_err);

[**errseq\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/errseq_t) [**f\_sb\_err**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_sb_err); */\* for syncfs \*/*

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout)

[**\_\_attribute\_\_**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__attribute__)(([**aligned**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/aligned)(4))); */\* lest something weird decides that 2 is OK \*/*

struct [**file\_handle**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_handle) {

[**\_\_u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__u32) [**handle\_bytes**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/handle_bytes);

int [**handle\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/handle_type);

*/\* file identifier \*/*

unsigned char [**f\_handle**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/f_handle)[];

};

extern struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*[**mount\_bdev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mount_bdev)(struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \*[**fs\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_type),

int flags, const char \*[**dev\_name**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_name), void \*data,

int (\*[**fill\_super**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fill_super))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*, void \*, int));

extern struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*[**mount\_nodev**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mount_nodev)(struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \*[**fs\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_type),

int flags, void \*data,

int (\*[**fill\_super**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fill_super))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*, void \*, int));

struct [**kmem\_cache**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache) \*[**kmem\_cache\_create**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache_create)(const char \*name, unsigned int size,

unsigned int [**align**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/align), [**slab\_flags\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/slab_flags_t) flags,

void (\*[**ctor**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ctor))(void \*));

void [**kmem\_cache\_destroy**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache_destroy)(struct [**kmem\_cache**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache) \*s);

void \*[**kmem\_cache\_alloc**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache_alloc)(struct [**kmem\_cache**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache) \*s, [**gfp\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/gfp_t) flags) // GFP\_KERNEL

void [**kmem\_cache\_free**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache_free)(struct [**kmem\_cache**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kmem_cache) \*s, void \*objp);

extern int [**register\_filesystem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/register_filesystem)(struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \*);

extern int [**unregister\_filesystem**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/unregister_filesystem)(struct [**file\_system\_type**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_system_type) \*);

extern struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*[**new\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/new_inode)(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*sb);

struct [**super\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_operations) {

struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*(\*[**alloc\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/alloc_inode))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*sb);

void (\*[**destroy\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/destroy_inode))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

void (\*[**free\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/free_inode))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

void (\*[**dirty\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dirty_inode)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, int flags);

int (\*[**write\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/write_inode)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**writeback\_control**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/writeback_control) \*[**wbc**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wbc));

int (\*[**drop\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/drop_inode)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*); = generic\_delete\_inode

void (\*[**evict\_inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/evict_inode)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

void (\*[**put\_super**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/put_super)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*);

int (\*[**sync\_fs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sync_fs))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*sb, int [**wait**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wait));

int (\*[**freeze\_super**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/freeze_super)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*);

int (\*[**freeze\_fs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/freeze_fs)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*);

int (\*[**thaw\_super**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/thaw_super)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*);

int (\*[**unfreeze\_fs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/unfreeze_fs)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*);

int (\*[**statfs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/statfs)) (struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, struct [**kstatfs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kstatfs) \*); = simple\_stat\_fs

int (\*[**remount\_fs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/remount_fs)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*, int \*, char \*);

void (\*[**umount\_begin**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umount_begin)) (struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*);

int (\*[**show\_options**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/show_options))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq_file) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**show\_devname**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/show_devname))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq_file) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**show\_path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/show_path))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq_file) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**show\_stats**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/show_stats))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq_file) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

#ifdef [**CONFIG\_QUOTA**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_QUOTA)

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**quota\_read**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/quota_read))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*, int, char \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t));

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**quota\_write**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/quota_write))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*, int, const char \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t));

struct [**dquot**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dquot) \*\*(\*[**get\_dquots**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/get_dquots))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

#endif

long (\*[**nr\_cached\_objects**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/nr_cached_objects))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*,

struct [**shrink\_control**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/shrink_control) \*);

long (\*[**free\_cached\_objects**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/free_cached_objects))(struct [**super\_block**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/super_block) \*,

struct [**shrink\_control**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/shrink_control) \*);

};

int [**generic\_delete\_inode**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/generic_delete_inode)(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/inode) \*[**inode**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/inode))

{

return 1;

}

int [**simple\_statfs**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/simple_statfs)(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/dentry) \*[**dentry**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/dentry), struct [**kstatfs**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/kstatfs) \*buf)

{

buf->[**f\_type**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/f_type) = [**dentry**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/dentry)->[**d\_sb**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/d_sb)->[**s\_magic**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/s_magic);

buf->[**f\_bsize**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/f_bsize) = [**PAGE\_SIZE**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/PAGE_SIZE);

buf->[**f\_namelen**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/f_namelen) = [**NAME\_MAX**](https://elixir.bootlin.com/linux/latest/source/fs/latest/C/ident/NAME_MAX);

return 0;

}

struct [**inode\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode_operations) {

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \* (\*[**lookup**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/lookup)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, unsigned int);

const char \* (\*[**get\_link**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/get_link)) (struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**delayed\_call**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/delayed_call) \*);

int (\*[**permission**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/permission)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, int);

struct [**posix\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/posix_acl) \* (\*[**get\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/get_acl))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, int, [**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool));

int (\*[**readlink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/readlink)) (struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, char [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*,int);

int (\*[**create**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/create)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

[**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t), [**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool));

int (\*[**link**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/link)) (struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**unlink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/unlink)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**symlink**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/symlink)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

const char \*);

int (\*[**mkdir**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mkdir)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

[**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t));

int (\*[**rmdir**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rmdir)) (struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**mknod**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mknod)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

[**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t),[**dev\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dev_t));

int (\*[**rename**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/rename)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, unsigned int);

int (\*[**setattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/setattr)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

struct [**iattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/iattr) \*);

int (\*[**getattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/getattr)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, const struct [**path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/path) \*,

struct [**kstat**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kstat) \*, [**u32**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u32), unsigned int);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**listxattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/listxattr)) (struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, char \*, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t));

int (\*[**fiemap**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fiemap))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**fiemap\_extent\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fiemap_extent_info) \*, [**u64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u64) start,

[**u64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/u64) len);

int (\*[**update\_time**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/update_time))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**timespec64**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/timespec64) \*, int);

int (\*[**atomic\_open**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_open))(struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*, struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*, unsigned [**open\_flag**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/open_flag),

[**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t) [**create\_mode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/create_mode));

int (\*[**tmpfile**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tmpfile)) (struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, [**umode\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umode_t));

int (\*[**set\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/set_acl))(struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*,

struct [**posix\_acl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/posix_acl) \*, int);

int (\*[**fileattr\_set**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fileattr_set))(struct [**user\_namespace**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/user_namespace) \*[**mnt\_userns**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mnt_userns),

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*[**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry), struct [**fileattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fileattr) \*[**fa**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fa));

int (\*[**fileattr\_get**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fileattr_get))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*[**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry), struct [**fileattr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fileattr) \*[**fa**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fa));

} [**\_\_\_\_cacheline\_aligned**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/____cacheline_aligned);

struct [**dentry\_operations**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry_operations) {

int (\*[**d\_revalidate**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_revalidate))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, unsigned int);

int (\*[**d\_weak\_revalidate**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_weak_revalidate))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, unsigned int);

int (\*[**d\_hash**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_hash))(const struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, struct [**qstr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/qstr) \*);

int (\*[**d\_compare**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_compare))(const struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*,

unsigned int, const char \*, const struct [**qstr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/qstr) \*);

int (\*[**d\_delete**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_delete))(const struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

int (\*[**d\_init**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_init))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

void (\*[**d\_release**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_release))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

void (\*[**d\_prune**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_prune))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*);

void (\*[**d\_iput**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_iput))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

char \*(\*[**d\_dname**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_dname))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, char \*, int);

struct [**vfsmount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/vfsmount) \*(\*[**d\_automount**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_automount))(struct [**path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/path) \*);

int (\*[**d\_manage**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_manage))(const struct [**path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/path) \*, [**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool));

struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*(\*[**d\_real**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/d_real))(struct [**dentry**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/dentry) \*, const struct [**inode**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/inode) \*);

} [**\_\_\_\_cacheline\_aligned**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/____cacheline_aligned);

struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) {

[**socket\_state**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket_state) state;

[**short**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/short) type;

unsigned long flags;

struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*[**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file);

struct [**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock) \*sk;

const struct [**proto\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proto_ops) \*ops;

struct [**socket\_wq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket_wq) [**wq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wq);

};

typedef enum {

[**SS\_FREE**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/SS_FREE) = 0, */\* not allocated \*/*

[**SS\_UNCONNECTED**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/SS_UNCONNECTED), */\* unconnected to any socket \*/*

[**SS\_CONNECTING**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/SS_CONNECTING), */\* in process of connecting \*/*

[**SS\_CONNECTED**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/SS_CONNECTED), */\* connected to socket \*/*

[**SS\_DISCONNECTING**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/SS_DISCONNECTING) */\* in process of disconnecting \*/*

} [**socket\_state**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/socket_state);

struct [**sockaddr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockaddr) {

[**sa\_family\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sa_family_t) [**sa\_family**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sa_family); */\* address family, AF\_xxx \*/*

char [**sa\_data**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sa_data)[14]; */\* 14 bytes of protocol address \*/*

};

struct [**sockaddr\_in**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sockaddr_in) {

[**\_\_kernel\_sa\_family\_t**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/__kernel_sa_family_t) [**sin\_family**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sin_family); */\* Address family \*/*

[**\_\_be16**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/__be16) [**sin\_port**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sin_port); */\* Port number \*/*

struct [**in\_addr**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/in_addr) [**sin\_addr**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sin_addr); */\* Internet address \*/*

*/\* Pad to size of `struct sockaddr'. \*/*

unsigned char [**\_\_pad**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/__pad)[[**\_\_SOCK\_SIZE\_\_**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/__SOCK_SIZE__) - sizeof([**short**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/short) int) -

sizeof(unsigned [**short**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/short) int) - sizeof(struct [**in\_addr**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/in_addr))];

};

Sys\_socket -> sock\_create

#define [**UNIX\_PATH\_MAX**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/UNIX_PATH_MAX) 108

struct [**sockaddr\_un**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sockaddr_un) {

[**\_\_kernel\_sa\_family\_t**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/__kernel_sa_family_t) [**sun\_family**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sun_family); */\* AF\_UNIX \*/*

char [**sun\_path**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/sun_path)[[**UNIX\_PATH\_MAX**](https://elixir.bootlin.com/linux/latest/source/include/uapi/linux/latest/C/ident/UNIX_PATH_MAX)]; */\* pathname \*/*

};

#define [**\_\_\_htonl**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___htonl)(x) [**\_\_cpu\_to\_be32**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/__cpu_to_be32)(x)

#define [**\_\_\_htons**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___htons)(x) [**\_\_cpu\_to\_be16**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/__cpu_to_be16)(x)

#define [**\_\_\_ntohl**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___ntohl)(x) [**\_\_be32\_to\_cpu**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/__be32_to_cpu)(x)

#define [**\_\_\_ntohs**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___ntohs)(x) [**\_\_be16\_to\_cpu**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/__be16_to_cpu)(x)

#define [**htonl**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/htonl)(x) [**\_\_\_htonl**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___htonl)(x)

#define [**ntohl**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/ntohl)(x) [**\_\_\_ntohl**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___ntohl)(x)

#define [**htons**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/htons)(x) [**\_\_\_htons**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___htons)(x)

#define [**ntohs**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/ntohs)(x) [**\_\_\_ntohs**](https://elixir.bootlin.com/linux/latest/source/include/linux/byteorder/latest/C/ident/___ntohs)(x)

struct [**proto\_ops**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/proto_ops) {

int [**family**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/family);

struct [**module**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/module) \*[**owner**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/owner);

int (\*[**release**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/release)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock));

int (\*[**bind**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bind)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock),

struct [**sockaddr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockaddr) \*[**myaddr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/myaddr),

int [**sockaddr\_len**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockaddr_len));

int (\*[**connect**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/connect)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock),

struct [**sockaddr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockaddr) \*[**vaddr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/vaddr),

int [**sockaddr\_len**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockaddr_len), int flags);

int (\*[**socketpair**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socketpair))(struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*sock1,

struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*sock2);

int (\*[**accept**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/accept)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock),

struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*newsock, int flags, [**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**kern**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/kern));

int (\*[**getname**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/getname)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock),

struct [**sockaddr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockaddr) \*addr,

int [**peer**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/peer));

[**\_\_poll\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__poll_t) (\*[**poll**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/poll)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*[**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file), struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock),

struct [**poll\_table\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/poll_table_struct) \*[**wait**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wait));

int (\*[**ioctl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ioctl)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), unsigned int cmd,

unsigned long arg);

#ifdef [**CONFIG\_COMPAT**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/K/ident/CONFIG_COMPAT)

int (\*[**compat\_ioctl**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/compat_ioctl)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), unsigned int cmd,

unsigned long arg);

#endif

int (\*[**gettstamp**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/gettstamp)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), void [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*userstamp,

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**timeval**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/timeval), [**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) time32);

int (\*[**listen**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/listen)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), int len);

int (\*[**shutdown**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/shutdown)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), int flags);

int (\*[**setsockopt**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/setsockopt))(struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), int [**level**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/level),

int [**optname**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/optname), [**sockptr\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sockptr_t) [**optval**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/optval),

unsigned int [**optlen**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/optlen));

int (\*[**getsockopt**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/getsockopt))(struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), int [**level**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/level),

int [**optname**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/optname), char [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*[**optval**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/optval), int [**\_\_user**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__user) \*[**optlen**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/optlen));

void (\*[**show\_fdinfo**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/show_fdinfo))(struct [**seq\_file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq_file) \*m, struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock));

int (\*[**sendmsg**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sendmsg)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), struct [**msghdr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/msghdr) \*m,

[**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t) [**total\_len**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/total_len));

*/\* Notes for implementing recvmsg:*

*\* ===============================*

*\* msg->msg\_namelen should get updated by the recvmsg handlers*

*\* iff msg\_name != NULL. It is by default 0 to prevent*

*\* returning uninitialized memory to user space. The recvfrom*

*\* handlers can assume that msg.msg\_name is either NULL or has*

*\* a minimum size of sizeof(struct sockaddr\_storage).*

*\*/*

int (\*[**recvmsg**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/recvmsg)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), struct [**msghdr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/msghdr) \*m,

[**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t) [**total\_len**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/total_len), int flags);

int (\*[**mmap**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mmap)) (struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) \*[**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file), struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock),

struct [**vm\_area\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/vm_area_struct) \* [**vma**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/vma));

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**sendpage**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sendpage)) (struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), struct [**page**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/page) \*[**page**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/page),

int offset, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t) size, int flags);

[**ssize\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ssize_t) (\*[**splice\_read**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/splice_read))(struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock), [**loff\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/loff_t) \*[**ppos**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/ppos),

struct [**pipe\_inode\_info**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pipe_inode_info) \*[**pipe**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pipe), [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t) len, unsigned int flags);

int (\*[**set\_peek\_off**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/set_peek_off))(struct [**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock) \*sk, int val);

int (\*[**peek\_len**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/peek_len))(struct [**socket**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/socket) \*[**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock));

*/\* The following functions are called internally by kernel with*

*\* sock lock already held.*

*\*/*

int (\*[**read\_sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/read_sock))(struct [**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock) \*sk, [**read\_descriptor\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/read_descriptor_t) \*desc,

[**sk\_read\_actor\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sk_read_actor_t) [**recv\_actor**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/recv_actor));

int (\*[**sendpage\_locked**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sendpage_locked))(struct [**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock) \*sk, struct [**page**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/page) \*[**page**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/page),

int offset, [**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t) size, int flags);

int (\*[**sendmsg\_locked**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sendmsg_locked))(struct [**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock) \*sk, struct [**msghdr**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/msghdr) \*msg,

[**size\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/size_t) size);

int (\*[**set\_rcvlowat**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/set_rcvlowat))(struct [**sock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sock) \*sk, int val);

};

extern struct [**task\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/task_struct) [**init\_task**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/init_task);

#define [**next\_task**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/next_task)(p) \

[**list\_entry\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/list_entry_rcu)((p)->[**tasks**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/tasks).next, struct [**task\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/task_struct), [**tasks**](https://elixir.bootlin.com/linux/latest/source/include/linux/sched/latest/C/ident/tasks))

struct [**task\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/task_struct) {

*\* describes a process running in the system, created dynamically. \*/*

int [**prio**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/prio);

int [**static\_prio**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/static_prio);

char [comm](https://docs.huihoo.com/doxygen/linux/kernel/3.7/structtask__struct.html#a59d00a3d4123d24c34c7d6a461b26a48)[[TASK\_COMM\_LEN](https://docs.huihoo.com/doxygen/linux/kernel/3.7/include_2linux_2sched_8h.html#a389bdfc15bea83bdd54353e1758e22e3)]; /\* executable name excluding path

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**tasks**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tasks);

…

struct [**mm\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mm_struct) \*[**mm**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mm);

struct [**mm\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/mm_struct) \*[**active\_mm**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/active_mm);

…

[**pid\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pid_t) [**pid**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pid);

[**pid\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pid_t) [**tgid**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/tgid);

…

*/\**

*\* Children/sibling form the list of natural children:*

*\*/*

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**children**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/children);

struct [**list\_head**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/list_head) [**sibling**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/sibling);

…

*/\* Filesystem information: \*/*

struct [**fs\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_struct) \*[**fs**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs);

*/\* Open file information: \*/*

struct [**files\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/files_struct) \*[**files**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/files);

*/\* Namespaces: \*/*

struct [**nsproxy**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/nsproxy) \*[**nsproxy**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/nsproxy);

};

struct [**fs\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fs_struct) {

*/\* information about the file system to which the process belongs\*/*

int [**users**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/users);

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) lock;

seqcount\_spinlock\_t [**seq**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/seq);

int [**umask**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/umask);

int [**in\_exec**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/in_exec);

struct [**path**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/path) root, [**pwd**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/pwd); the mounting object of the root directory and working directory

} [**\_\_randomize\_layout**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__randomize_layout);

*/\**

*\* Open file table structure*

*\*/*

struct [**files\_struct**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/files_struct) {

*/\**

*\* read mostly part*

*\*/*

[**atomic\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/atomic_t) count;

[**bool**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/bool) [**resize\_in\_progress**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/resize_in_progress);

[**wait\_queue\_head\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/wait_queue_head_t) [**resize\_wait**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/resize_wait);

struct [**fdtable**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fdtable) [**\_\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__rcu) \*[**fdt**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fdt);

struct [**fdtable**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fdtable) [**fdtab**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fdtab);

*/\**

*\* written part on a separate cache line in SMP*

*\*/*

[**spinlock\_t**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/spinlock_t) [**file\_lock**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file_lock) [**\_\_\_\_cacheline\_aligned\_in\_smp**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/____cacheline_aligned_in_smp);

unsigned int [**next\_fd**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/next_fd);

unsigned long [**close\_on\_exec\_init**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/close_on_exec_init)[1]; // fd that should be closed when exec() is called

unsigned long [**open\_fds\_init**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/open_fds_init)[1]; // initial set of fd

unsigned long [**full\_fds\_bits\_init**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/full_fds_bits_init)[1];

struct [**file**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/file) [**\_\_rcu**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/__rcu) \* [**fd\_array**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/fd_array)[[**NR\_OPEN\_DEFAULT**](https://elixir.bootlin.com/linux/latest/source/include/linux/latest/C/ident/NR_OPEN_DEFAULT)];

};