



Linux Ubuntu 26.04 Manual Pages on command 'landlock_create_ruleset.2'

\$ man landlock_create_ruleset.2

landlock_create_ruleset(2) System Calls Manual landlock_create_ruleset(2)

NAME

landlock_create_ruleset - create a new Landlock ruleset

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <linux/landlock.h> /* Definition of LANDLOCK_* constants */
#include <sys/syscall.h>    /* Definition of SYS_* constants */
#include <unistd.h>

int syscall(SYS_landlock_create_ruleset,
           const struct landlock_ruleset_attr *attr,
           size_t size, uint32_t flags);
```

DESCRIPTION

A Landlock ruleset identifies a set of rules (i.e., actions on objects). This `landlock_create_ruleset()` system call creates a new file descriptor which identifies a ruleset. This file descriptor can then be used by `landlock_add_rule(2)` and `landlock_restrict_self(2)`. See `landlock(7)` for a global overview.

`attr` specifies the properties of the new ruleset. It points to the following structure:

```
struct landlock_ruleset_attr {
    __u64 handled_access_fs;
    __u64 handled_access_net;
};
```

`handled_access_fs` is a bitmask of handled filesystem actions (see `Filesystem actions`

in `landlock(7)`).

`handled_access_net` is a bitmask of handled network actions (see Network actions in `landlock(7)`).

This structure defines a set of handled access rights, a set of actions on different object types, which should be denied by default when the ruleset is enacted. Vice versa, access rights that are not specifically listed here are not going to be denied by this ruleset when it is enacted.

For historical reasons, the `LANDLOCK_ACCESS_FS_REFER` right is always denied by default, even when its bit is not set in `handled_access_fs`. In order to add new rules with this access right, the bit must still be set explicitly (see Filesystem actions in `landlock(7)`).

The explicit listing of handled access rights is required for backwards compatibility reasons. In most use cases, processes that use Landlock will handle a wide range or all access rights that they know about at build time (and that they have tested with a kernel that supported them all).

This structure can grow in future Landlock versions.

size must be specified as `sizeof(struct landlock_ruleset_attr)` for compatibility reasons.

flags must be 0 if attr is used. Otherwise, flags can be set to:

`LANDLOCK_CREATE_RULESET_VERSION`

If attr is NULL and size is 0, then the returned value is the highest supported Landlock ABI version (starting at 1). This version can be used for a best-effort security approach, which is encouraged when user space is not pinned to a specific kernel version. All features documented in these man pages are available with the version 1.

RETURN VALUE

On success, `landlock_create_ruleset()` returns a new Landlock ruleset file descriptor, or a Landlock ABI version, according to flags. On error, -1 is returned and `errno` is set to indicate the error.

ERRORS

`landlock_create_ruleset()` can fail for the following reasons:

`EOPNOTSUPP`

Landlock is supported by the kernel but disabled at boot time.

`EINVAL` Unknown flags, or unknown access, or too small size.

E2BIG size is too big.

EFAULT attr was not a valid address.

ENOMSG Empty accesses (i.e., attr did not specify any access rights to restrict).

STANDARDS

Linux.

HISTORY

Linux 5.13.

EXAMPLES

See `landlock(7)`.

SEE ALSO

`landlock_add_rule(2)`, `landlock_restrict_self(2)`, `landlock(7)`

Linux man-pages 6.17

2026-02-08

`landlock_create_ruleset(2)`