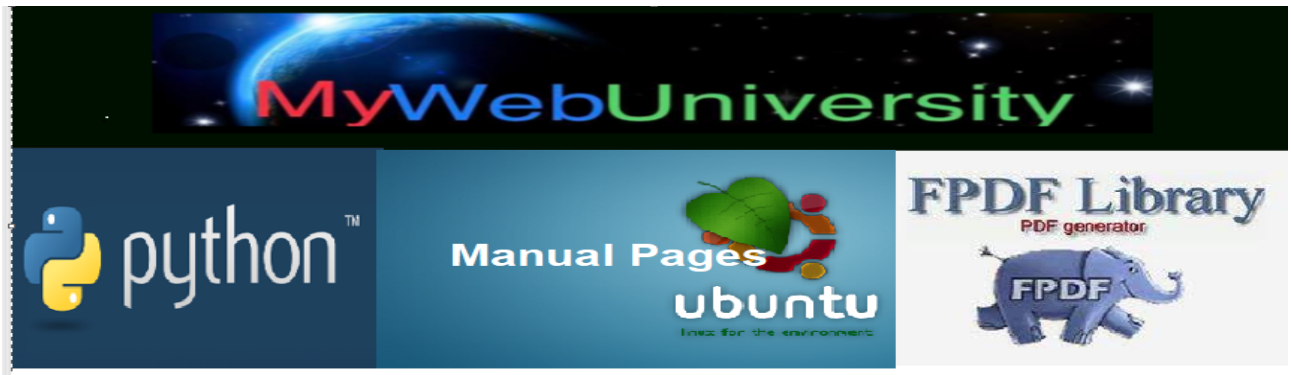

Full credit is given to the above companies including the OS that this PDF file was generated!

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

$ man landlock_add_rule.2

landlock_add_rule(2) System Calls Manual landlock_add_rule(2)

NAME

landlock_add_rule - add a new Landlock rule to a ruleset

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <linux/landlock.h>  /* Definition of LANDLOCK_* constants */
#include <sys/syscall.h>     /* Definition of SYS_* constants */
int syscall(SYS_landlock_add_rule, int ruleset_fd,
            enum landlock_rule_type rule_type,
            const void *rule_attr, uint32_t flags);
```

DESCRIPTION

A Landlock rule describes an action on an object which the process intends to perform. A set of rules is aggregated in a ruleset, which can then restrict the thread enforcing it, and its future children.

The landlock_add_rule() system call adds a new Landlock rule to an existing ruleset. See landlock(7) for a global overview.

ruleset_fd is a Landlock ruleset file descriptor obtained with landlock_create_ruleset(2).

rule_type identifies the structure type pointed to by rule_attr. Currently, Linux supports the following rule_type values:

LANDLOCK_RULE_PATH_BENEATH

For these rules, the object is a file hierarchy, and the related filesystem actions are defined with filesystem access rights.

In this case, rule_attr points to the following structure:

```
struct landlock_path_beneath_attr {
    __u64 allowed_access;
    __s32 parent_fd;
} __attribute__((packed));
```

allowed_access contains a bitmask of allowed filesystem actions, which can be applied on the given parent_fd (see Filesystem actions in landlock(7)).

parent_fd is an opened file descriptor, preferably with the O_PATH flag, which identifies the parent directory of the file hierarchy or just a file.

LANDLOCK_RULE_NET_PORT

For these rules, the object is a TCP port, and the related actions are defined with network access rights.

In this case, rule_attr points to the following structure:

```
struct landlock_net_port_attr {
    __u64 allowed_access;
    __u64 port;
};
```

allowed_access contains a bitmask of allowed network actions, which can be applied on the given port.

port is the network port in host endianness.

It should be noted that port 0 passed to bind(2) will bind to an available port from the ephemeral port range. This can be configured in the /proc/sys/net/ipv4/ip_local_port_range sysctl (also used for IPv6).

A Landlock rule with port 0 and the LANDLOCK_ACCESS_NET_BIND_TCP right means that requesting to bind on port 0 is allowed and it will automatically translate to binding on the related port range.

flags must be 0.

RETURN VALUE

On success, landlock_add_rule() returns 0. On error, -1 is returned and errno is set to indicate the error.

ERRORS

landlock_add_rule() can fail for the following reasons:

EAFNOSUPPORT

rule_type is LANDLOCK_RULE_NET_PORT, but TCP is not supported by the running kernel.

EOPNOTSUPP

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

EINVAL flags is not 0.

EINVAL The rule accesses are inconsistent (i.e., rule_attr->allowed_access is not a subset of the ruleset handled accesses).

EINVAL In struct landlock_path_beneath_attr, the rule accesses are not applicable to the file (i.e., some access rights in rule_attr->allowed_access are only applicable to directories, but rule_attr->parent_fd does not refer to a directory).

EINVAL In struct landlock_net_port_attr, the port number is greater than 65535.

ENOMSG Empty accesses (i.e., rule_attr->allowed_access is 0).

EBADF ruleset_fd is not a file descriptor for the current thread, or a member of rule_attr is not a file descriptor as expected.

EBADFD ruleset_fd is not a ruleset file descriptor, or a member of rule_attr is not the ex? pected file descriptor type.

EPERM ruleset_fd has no write access to the underlying ruleset.

EFAULT rule_attr was not a valid address.

STANDARDS

Linux.

HISTORY

Linux 5.13.

EXAMPLES

See landlock(7).

SEE ALSO

landlock_create_ruleset(2), landlock_restrict_self(2), landlock(7)