Landlock LSM: kernel documentation

Author: MickaÃ«l SalaÃ¼n
Date: March 2021

Landlock's goal is to create scoped access-control (i.e. sandboxing). To harden a whole system, this feature should be available to any process, including unprivileged ones. Because such process may be compromised or backdoored (i.e. untrusted), Landlock's features must be safe to use from the kernel and other processes point of view. Landlock's interface must therefore expose a minimal attack surface.

Landlock is designed to be usable by unprivileged processes while following the system security policy enforced by other access control mechanisms (e.g. DAC, LSM). Indeed, a Landlock rule shall not interfere with other access-controls enforced on the system, only add more restrictions.

Any user can enforce Landlock rulesets on their processes. They are merged and evaluated according to the inherited ones in a way that ensures that only more constraints can be added.

User space documentation can be found here: Documentation/userspace-api/landlock.rst.

Guiding principles for safe access controls

- A Landlock rule shall be focused on access control on kernel objects instead of syscall filtering (i.e. syscall arguments), which is the purpose of seccomp-bpf.
- To avoid multiple kinds of side-channel attacks (e.g. leak of security policies, CPU-based attacks), Landlock rules shall not be able to programmatically communicate with user space.
- Kernel access check shall not slow down access request from unsandboxed processes.
- Computation related to Landlock operations (e.g. enforcing a ruleset) shall only impact the processes requesting them.

Tests

Userspace tests for backward compatibility, ptrace restrictions and filesystem support can be found here: tools/testing/selftests/landlock/.

Kernel structures

Object

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\security\[linux-master][Documentation][security]landlock.rst, line 57)

Unknown directive type "kernel-doc".

```
.. kernel-doc:: security/landlock/object.h
    :identifiers:
```

Filesystem

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\security\[linux-master][Documentation][security]landlock.rst, line 63)

Unknown directive type "kernel-doc".

```
.. kernel-doc:: security/landlock/fs.h
    :identifiers:
```

Ruleset and domain

A domain is a read-only ruleset tied to a set of subjects (i.e. tasks' credentials). Each time a ruleset is enforced on a task, the current domain is duplicated and the ruleset is imported as a new layer of rules in the new domain. Indeed, once in a domain, each rule is tied to a layer level. To grant access to an object, at least one rule of each layer must allow the requested action on the object. A task can then only transit to a new domain that is the intersection of the constraints from the current domain and those of a ruleset provided by the task.

The definition of a subject is implicit for a task sandboxing itself, which makes the reasoning much easier and helps avoid pitfalls.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\security\[linux-master][Documentation][security]landlock.rst, line 81)

Unknown directive type "kernel-doc".

```
.. kernel-doc:: security/landlock/ruleset.h
    :identifiers:
```