RUSTSEC-2020-0094

History · Edit

Unsound: can make `ARefss` contain a !Send, !Sync object.

Reported	December 1, 2020
Issued	January 6, 2021 (last modified: October 19, 2021)
Package	reffers (crates.io)
Type	INFO Unsound
Categories	memory-corruption thread-safety
Keywords	#concurrency
Aliases	CVE-2020-36203
Details	https://github.com/diwic/reffers-rs/issues/7
CVSS Score	4.7 MEDIUM
CVSS Details	

Attack vector	Local
Attack complexity	High
Privileges required	Low
User interaction	None
Scope	Unchanged
Confidentiality	None
Integrity	None
Availability	High

CVSS Vector	CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:H
Patched	`>=0.6.1`

Description

`ARefss<'a, V>` is a type that is assumed to contain objects that are `Send + Sync`.

In the affected versions of this crate, `Send` / `Sync` traits are unconditionally implemented for `ARefss<'a, V>`.

By using the `ARefss::map()` API, we can insert a `!Send` or `!Sync` object into `ARefss<'a, V>`. After that, it is possible to create a data race to the inner object of `ARefss<'a, V>`, which can lead to undefined behavior & memory corruption.

The flaw was corrected in commit 6dd7ca0 (https://github.com/diwic/reffers-rs/commit/6dd7ca0d50f2464df708975cdafcfaeeb6d41c66) by adding trait bound `V: Send + Sync` to `ARefss::map()` API.