RUSTSEC-2020-0060

futures_task::waker may cause a use-after-free if used on a type that isn't 'static

Reported	September 4, 2020
Issued	October 31, 2020 (last modified: October 19, 2021)
Package	futures-task (crates.io)
Type	Vulnerability
Categories	code-execution memory-corruption
Keywords	#use-after-free #arbitrary-code-execution #memory-corruption #memory-management
Aliases	CVE-2020-35906
Details	https://github.com/rust-lang/futures-rs/pull/2206
CVSS Score	7.8 HIGH

CVSS Details

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

CVSS Vector	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
Patched	`>=0.3.6`
Unaffected	`<=0.2.1`

Affected Functions	Version
`futures_task::waker`	`>=0.3.0`

Description

Affected versions of the crate did not properly implement a `'static` lifetime bound on the `waker` function. This resulted in a use-after-free if `Waker::wake()` is called after original data had been dropped.

The flaw was corrected by adding `'static` lifetime bound to the data `waker` takes.