



Look up package or ID...

[About](#) [Advisories](#) [Report Vulnerabilities](#)



## RUSTSEC-2020-0080

[History](#) · [Edit](#)

**miow** invalidly assumes the memory layout of `std::net::SocketAddr`

**Reported** November 13, 2020

**Issued** December 2, 2020 (last modified: October 19, 2021)

**Package** [miow](#) ([crates.io](#))

**Type** INFO Unsound

**Keywords** [#memory](#) [#layout](#) [#cast](#)

**Aliases** [CVE-2020-35921](#)

**Details** <https://github.com/yoshuawuyts/miow/issues/38>

**CVSS Score** 5.5 MEDIUM

### CVSS Details

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

**CVSS Vector** [CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H](#)

**Patched** `<0.2.2`  
`>=0.3.6`

### Description

The `miow` crate has assumed `std::net::SocketAddrV4` and `std::net::SocketAddrV6` have the same memory layout as the system C representation `sockaddr`. It has simply casted the pointers to convert the socket addresses to the system representation. The standard library does not say anything about the memory layout, and this will cause invalid memory access if the standard library changes the implementation. No warnings or errors will be emitted once the change happens.