Support for Non-Lexical Lifetimes (NLL) has been included in the Rust compiler since 1.31, and has been enabled on the 2015 edition since 1.36. The new borrow checker for NLL uncovered some bugs in the old borrow checker, which in some cases allowed unsound code to compile, resulting in memory safety issues.

What do I do?

Change your code so the warning does no longer trigger. For backwards compatibility, this unsound code may still compile (with a warning) right now. However, at some point in the future, the compiler will no longer accept this code and will throw a hard error.

Shouldn't you fix the old borrow checker?

The old borrow checker has known soundness issues that are basically impossible to fix. The new NLL-based borrow checker is the fix.

Can I turn these warnings into errors by denying a lint?

No.

When are these warnings going to turn into errors?

No formal timeline for turning the warnings into errors has been set. See GitHub issue 58781 for more information.

Why do I get this message with code that doesn't involve borrowing?

There are some known bugs that trigger this message.