A cross-crate opt-out trait was implemented on something which wasn't a struct or enum type.

Erroneous code example:

```
#![feature(auto_traits)]

struct Foo;

impl !Sync for Foo {}

unsafe impl Send for &'static Foo {}
// error: cross-crate traits with a default impl, like `core::marker::Send`,
//        can only be implemented for a struct/enum type, not
//        `&'static Foo`
```

Only structs and enums are permitted to impl Send, Sync, and other opt-out trait, and the struct or enum must be local to the current crate. So, for example, `unsafe impl Send for Rc<Foo>` is not allowed.