

Note: this error code is no longer emitted by the compiler. Sub-bindings, e.g. `ref x @ Some(ref y)` are now allowed under `#![feature(bindings_after_at)]` and checked to make sure that memory safety is upheld.

In certain cases it is possible for sub-bindings to violate memory safety. Updates to the borrow checker in a future version of Rust may remove this restriction, but for now patterns must be rewritten without sub-bindings.

Before:

```
match Some("hi".to_string()) {
    ref op_string_ref @ Some(s) => {},
    None => {},
}
```

After:

```
match Some("hi".to_string()) {
    Some(ref s) => {
        let op_string_ref = &Some(s);
        // ...
    },
    None => {},
}
```

The `op_string_ref` binding has type `&Option<&String>` in both cases.

See also Issue 14587.