Paper Reading -- Eraser --

topic Eraser
date: 2018/10/25
ID: 516413990003
Name: Huichuan Hui

1. According to the lockset algorithm, when does eraser signal a data race? Why is this condition chosen?

Eraser checks whether the program respects this discipline, that every shared variable is protected by some lock, in the sense that the lock is held by any thread whenever it accesses the variable, by monitoring all reads and writes as the program executes. Since Eraser has no way of knowing which locks are intended to protect which variables, it must infer the protection relation from the execution history.

2. Under what conditions does Eraser report a false positive? What conditions does it produce false negatives?

For example, if a thread t1 reads v while holding lock m 1 , and a thread t 2 writes v while holding lock m 2 , the violation of the locking discipline will be reported only if the write precedes the read.

3. Typically, instrumenting a program changes the intra-thread timing (the paper calls it interleaving). This can cause bugs to disappear when you start trying to find them. What aspect of the Eraser design mitigates this problem?

Eraser in which the lock covers technique of Dinning and Schonberg, is an improvement to the happens-before approach for programs that make heavy use of locks. Eraser engineers extended Dinning and Schonberg’s improvement and discard the underlying happens-before apparatus that is highly dependent on the interleaving produced by the scheduler. While Eraser is a testing tool and therefore cannot guarantee that a program is free from races, it can detect more races than tools based on happens-before.