

Major breakthrough improves software reliability and security

November 2 2011

Anyone who uses multithreaded computer programs -- and that's all of us, as these are the programs that power nearly all software applications including Office, Windows, MacOS, and Google Chrome Browser, and web services like Google Search, Microsoft Bing, and iCloud, -- knows well the frustration of computer crashes, bugs, and other aggravating problems. The most widely used method to harness the power we require from multicore processors, multithreaded programs can be difficult for programmers to get right and they often contain elusive bugs called races. Data races can cause very serious problems, like the software bug that set off the 2003 power blackout in the Northeast. Now there is a new system that will combat this problem.

Peregrine, a new software system developed by a team of researchers at Columbia Engineering School, led by Assistant Professor of Computer Science Junfeng Yang, will improve the reliability and security of multithreaded programs, benefiting virtually every computer user across the globe. Peregrine can be used by software vendors like Microsoft and Apple and web service providers like [Google](#) and [Facebook](#), to provide reliable services to computer users. This new research was published in the 23rd ACM Symposium on Operating Systems Principles, considered to be the most prestigious systems conference held each year, and presented by Yang's graduate student Heming Cui at Cascais, Portugal, on Oct. 26. The paper can be found [online](#).

"Multithreaded programs are becoming more and more critical and pervasive," says Professor Yang. "But these programs are

nondeterministic, so running them is like tossing a coin or rolling dice -- sometimes we get correct results, and sometimes we get wrong results or the program crashes. Our main finding in developing Peregrine is that we can make threads deterministic in an efficient and stable way: Peregrine can compute a plan for allowing when and where a thread can "change lanes" and can then place barriers between the lanes, allowing threads to change lanes only at fixed locations, following a fixed order. This prevents the random collisions that can occur in a nondeterministic system.

"Once Peregrine computes a good plan without collisions for one group of threads," adds Yang, "it can reuse the plan on subsequent groups to avoid the cost of computing a new plan for each new group. This approach matches our natural tendency to follow familiar routes so we can avoid both potential hazards in unknown routes and efforts to find a new route."

Yang notes that in contrast to many earlier systems that address only resultant problems but not the root cause, Peregrine addresses nondeterminism -- a system that is unpredictable as each input has multiple potential outcomes -- and thus simultaneously addresses all the problems that are caused by nondeterminism.

Peregrine also deals with data races or bugs, unlike most previous efforts that do not provide such fine-grained control over the execution of a program. And it's very fast -- many earlier systems may slow down the execution of a program by up to ten times. Peregrine is also a practical system that works with current hardware and programming languages -- it does not require new hardware or new languages, all of which can take years to develop. It reuses execution plans, whereas some previous work makes a different plan for each group of threads: as Yang points out, "The more plans one makes, the more likely some plans have errors and will lead to collisions."

"Today's software systems are large, complex, and plagued with errors, some of which have caused critical system failures and exploits," adds Yang. "My research is focused on creating effective tools to improve the reliability and security of real software systems. I'm excited about this area because it has the potential to make the cyberspace a better place and benefit every government, business, and individual who uses computers."

Provided by Columbia University

Citation: Major breakthrough improves software reliability and security (2011, November 2)
retrieved 18 April 2024 from

<https://phys.org/news/2011-11-major-breakthrough-software-reliability.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.