NATIONAL UNIVERSITY OF SINGAPORE

School of Computing

PH.D DEFENCE - PUBLIC SEMINAR

Title: **Systematic Methods for Memory Error Detection and Exploitation**

Speaker: Mr Hu Hong

Date/Time: 3 May 2016, Tuesday, 04:00 PM to 05:30 PM

Venue: MR1, COM1-03-19

Supervisor : Dr Liang Zhenkai, Associate Professor, School of Computing

Abstract:

Memory errors are persistent threats to computer systems. Attacks exploiting memory errors have resulted in severe damage in real-world incidents. At the same time, exploit mechanisms are rapidly evolving, enabling memory errors (even old ones) to bypass known protections. To defend against memory errors, it is crucial for us to detect them in advance and understand the consequence of new exploit mechanisms. In this thesis, we focus on new possible exploits resulted from changes of program fragment integration, as existing analysis methods take the program as an entirety and thus fall short on such new exploits.

We propose three novel solutions to detect memory errors and identify new exploit vectors. Our insight is that once the fragments that comprise the program change to integrate in an unexpected way of the design, the change will introduce new memory errors and exploits. In particular, we identify dereference-under-the-influence (DUI) vulnerability, a problem raised during privilege-based program transformation. We build a systematic way to detect DUI code in program fragments. Then we look into new exploit mechanisms resulted from changes in program fragments integration. Specifically, we study the consequence of data flow re-construction, where original data flows in a program are split into small fragments and reconstructed by memory errors. We propose a novel method, called data-flow stitching, to connect data flow fragments to build data-oriented attacks. Data-flow stitching significantly enlarges the space of data-oriented attacks. Finally, we explore the expressiveness of data-oriented attacks. We propose a new exploit construction technique, called data-oriented programming (DOP), to selectively stitch fundamental data-flow fragments for a desired purpose. With DOP, we build Turing-complete data-oriented attacks, starting from common memory errors, showing the counterintuitively strong expressiveness of data-oriented exploits.

With our work, we demonstrate that benign program fragments can lead to significant damage to real-world programs, when their integration method is changed due to various reasons, like attack or program transformation.