

# Memory Safety and Buffer Overflow Lecture

Host: Holly Rushmeier



Friday, March 26, 2021

10:30 a.m.

Speaker: Jay P. Lim

Zoom Presentation

**Abstract:** Memory safety is a concept of protecting the memory and data of programs. Because memory safety is closely related to the integrity of the data, it is paramount for developing safe and secure software. Memory safety errors can occur when an unallocated region of memory is accessed or when memory is accessed using methods that cause undefined behaviors. In this lecture, we will discuss about the various causes and consequences of memory safety error as well as techniques to prevent them. We will then focus on a particular class of memory safety known as buffer overflow error and its implication in data integrity.

This lecture is designed for students who have some experiences in programming in C language. In particular, knowledge in the interactions between pointers, arrays, and dynamic allocation of memory is paramount for the lecture. Understanding the basics of stack and heap memory region is recommended, and usually taught in previous lectures.

**Bio:** Jay P. Lim is a seventh year Ph.D. Student at Rutgers University advised by Professor Santosh Nagarakatte. His research focus is on developing techniques to verify and synthesizes safe and secure low level computer systems. Jay completed his undergraduate degree in Computer Science and Mathematics at the University of Wisconsin-Madison.

Jay is passionate about teaching and he has been a TA and an instructor for various courses during his time in Ph.D. program including Computer Architecture, Programming Languages, Compilers, Computer Security, and Discrete Math. He has been an instructor for both large undergraduate courses and small graduate courses. One of his passions is to make learning interesting and fun to engage the students. It is his happiness to see students enjoying his classes.

Harper Dissertation Fellowship. His personal website is <https://people.cs.uchicago.edu/~yongshan/>.