

Téma: Efficient Loop Navigation for Symbolic Execution

Autoři: Jan Obdržálek, Marek Trtík

Přednáší: Marek Trtík

Abstrakt:

Symbolic execution is an important, successful and very popular technique used in software verification and testing, including dynamic test generation. A key limitation of symbolic execution is in dealing with code containing loops. The problem is that even a single loop can generate a huge number of different symbolic execution paths, corresponding to different number of loop iterations and various paths through the loop. The task of reaching a specific location anywhere in the code below the loop is then very difficult. As a possible solution to this problem we introduce a technique which, given a start location above some loops and a target location anywhere below these loops, returns a feasible path between these two locations, if exists. The technique infers a collection of constraint systems from the program and uses them to steer the symbolic execution towards the target. On reaching a loop it iteratively solves the appropriate constraint system to find out which path through this loop to take, or, alternatively, whether to continue below the loop. To construct the constraint system we express the values of variables modified in a loop as functions of the number of times a given path through the loop was executed. We have built a prototype implementation of our technique and compared it to state-of-the-art symbolic execution tools on several simple programs with loops. The results show significant improvements in the running time. We found instances where our algorithm finished in seconds, whereas the other tools did not finish within an hour. Our approach also shows very good results in the case when the target location is not reachable by any feasible path.