AUTOMATED REASONING IN SECURITY

A recent successful approach to security analysis reduces security questions about programs to constraint satisfaction problems in some formal logic. Automatic reasoners for that logic can then be used to solve those problems.

Example: A code for a password verification with a buffer overflow attack vulnerability is reduced to the set of constraints, solution to which provides an instance of a buffer overflow attack – i.e. proof of the vulnerability.

SOLVING STRING CONSTRAINTS

A major difficulty is that, in general, the satisfiability problem of any reasonably comprehensive theory of character strings is undecidable. However, one can identify several restricted, but still quite useful, fragments of the theory of strings that are decidable.

Theoretical Complexity Challenges

Recent research has focused on identifying decidable fragments suitable for program analysis and developing efficient solvers for them. Most of the string solvers are stand-alone tools that can reason only about small fragments of the theory of strings. These solvers are based on reductions to the satisfiability problems over other data types, such as bit-vectors, or to decision problems over automata. At the same time, they are either unsound, or lack expressiveness.

CVC4 STRING SOLVER

We developed a new algebraic approach for solving constraints over:

- a theory of unbounded strings,
- length constraints,
- extended regular expression membership,
- common string manipulating functions

directly without reducing to other problems. We implemented an automated string solver based on this approach, and incorporated it into the state-of-the-art SMT solver CVC4.

THEORETICAL RESULTS

We have a general proof of correctness for our string solver in terms of refutation soundness and solution completeness. We have also identified two expressive fragments for which it is a decision procedure: one with unbounded strings and length constraints and one with regular language membership and length constraints.