Bounded Model Checking – dealing with infinite state trajectories

Description Given a system that is modeled as a Finite State Machine (FSM), and specifications of what that system is supposed to do, model checking is the task of making sure that all specifications are satisfied by the model. This applies to hardware as well as software. The specifications state functional, security, and safety requirements (avoidance of bad states that can be exploited by an attacker or that result in a crash). The problem of model checking is solved using some form of logic. In most cases some form of temporal logic is best suited for this role because operators in temporal logic are designed to handle state queries about events indefinitely far into the future.

Over the years Boolean logic has risen as a great tool for solving hard but significant real-world problems including many instances of NP-complete problems: this is due to the efficiency of SAT and SMT solvers. But Boolean formulas cannot be allowed to grow arbitrarily to match what temporal logic can do. However, a FSM can be "unrolled" for a finite number of states and properties checked in that limited space. That is known as Bounded Model Checking (BMC). BMC, with the success of SAT and SMT solver progression over the years, has become and important and effective part of model checking; increasing confidence in system robustness.

Purpose Provides illumination of and practice in using Cryptol to solve problems with Bounded Model Checking.

Audience This module is intended for:

1 The general public
2 K-12 and college classes on cyber defense
3 preparation for proficiency in the use of tools and a computing environment suitable for the study of cyber defense

Objectives After completing the module:

1 know the type of problem that can be solved with BMC
2 know how to set up a BMC solution in Cryptol
3 Understand the benefits and limitations of BMC

Keywords Bounded Model Checking (BMC), Cryptol, Temporal Logic, Hardware Verification, Set/Reset Latch, Simple Counter, Properties to Prove

Category cybersecurity > education

Delivery java applets and written documentation in pdf format

Team John Franco and Ethan Link

Assessment The applets provide the means for experimentation. Questions are asked in the documentation that help with the set up of experiments. The ideas that learners come up with is evidence that the module was successful.

Workflow No particular schedule was established

Environment All materials are contained in a single jar file. The jar file can be run on any computer where java version 14 or higher and some pdf reader such as acroread or evince are available. The jar file may be executed in the cyber range or learners may download the jar file (which is considered to be an executable file) and run it on their personal computers.