

A Tool For Game-Based Proofs

Problem Statement

Many security proofs of cryptographic protocols are conceptually simple but technically complex. Using the game rewriting technique can be repetitive and error-prone. Errors can emerge from the misapplication of steps due to oversight or simple typos. The games can be quite lengthy, so rewriting is time-consuming and it can be difficult to locate the necessary parts of the game.

Most of the existing automatic prover tools convert the initial protocol into a low-level description and then perform a series of local substitutions until the final goal is reached. As a result, one loses interpretability and cannot easily guide the prover nor interpret the results in case of security flaws.

What Does ProveIt Do?

As a semi-automatic prover environment, **ProveIt** can greatly simplify the design and verification phase for new protocols as well as help with proving existing protocols.

ProveIt does not perform automatic proofs but assists the user in the proof process to make the tedious and tricky proof validation tractable using the protocol rewriting technique^{1,2}. This method is intuitive and assists the prover in finding the path between the initial and the final cryptographic game. This is done by applying certain transformations to the statements of the protocol. Each reduction step changes the game according to a previously specified pattern (reduction schema).

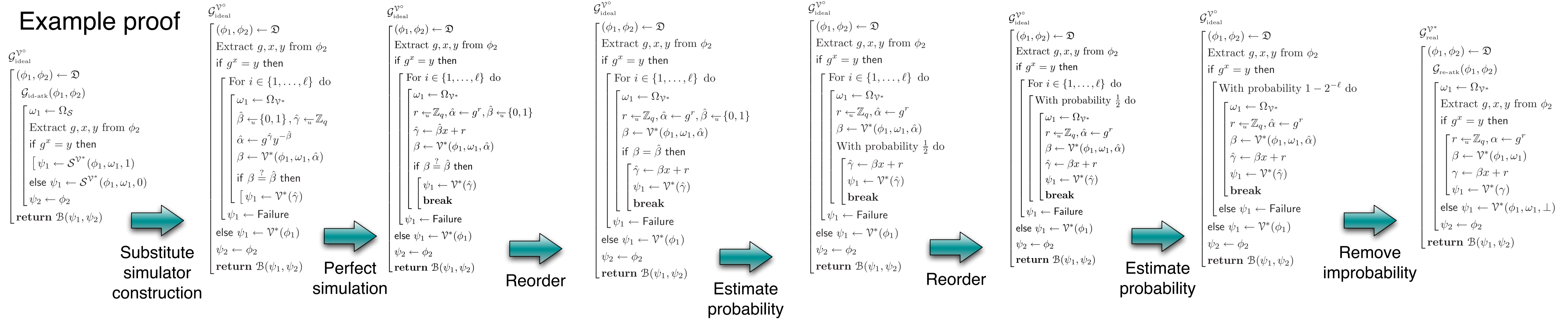
Who Is ProveIt For?

ProveIt makes proving cryptographic protocols easier and more intuitive, making it more usable to engineers and students.

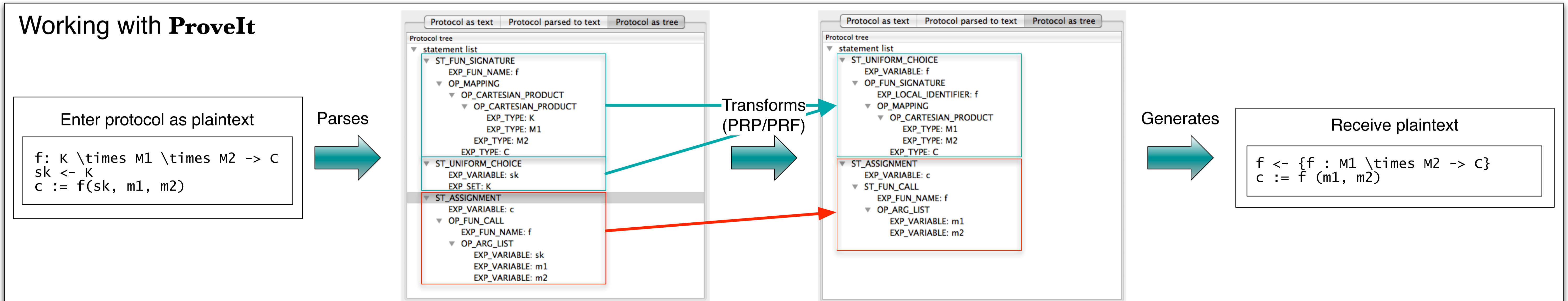
The user can check whether a reduction schema is applicable for the game. This prevents oversight and helps to ensure that the resulting proof is sound. One can also learn about the different reduction schemas and their applicability.

ProveIt helps the user with the most tedious and repetitive part of game-based proving, i.e. game rewriting. It helps concentrate on the process of proving the protocol instead of writing the games over and over and can save a lot of time, as trying different schemas becomes easier and less time-consuming.

Example proof



Working with ProveIt



¹Mihir Bellare and Phillip Rogaway. Code-based game-playing proofs and the security of triple encryption. Cryptology ePrint Archive, Report 2004/331, 2004.

²Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Report 2004/332, 2004.