Domenic Cianfichi
Hugo Mailhot
Joseph McGee

Midterm project report

What we have done so far

We started our study of Script by studying its context of occurrence, the Bitcoin network. We identified the current use cases, and the standard templates that a Script program must satisfy to be accepted in the mainstream network. We also identified desirable and undesirable properties of a Script script.

We studied Forth, the language from which Script is derived. Doing so gave us an idea of the kind of notation and abstractions we can use to reason about a stack-based language.[1]

We agreed on a set of useful concepts and notations to describe Script and reason about it. We used this descriptive apparatus to express the syntax and the operational semantics of the language.

We described some of the commands in the language having to do with how bytes are read by the interpreter. We had to think about (and are still dealing with) the problem of how to balance abstractness and reliable description of the language. We find that describing the language in terms of its bytecode management can become inelegant, but completely abstracting away these features of the language would oversimplify our analysis.

What remains

Now that we know what standards were set by the community on Bitcoin scripts, we want to use our syntax and semantics to reason about the following:

1. Whether the standards rule out undesirable scripts
2. Whether a weaker set of constraints could still accomplish this
3. How past versions of the language were flawed

What the plan is

To continue meeting many times a week and generate enough analyses and insights to start writing and recording the final report by March 15th.

[1] Particularly useful was this document, which gave a short description of an abstract POSTFIX language: http://cs.wellesley.edu/~cs251/spring07/postfix.pdf.