Modelling and verifying Bitcoin contracts

(Invited Talk)

Massimo Bartoletti
University of Cagliari, Italy
bart@unica.it

Albeit the primary usage of Bitcoin is to exchange currency, its blockchain and consensus mechanism can also be exploited to securely execute some forms of smart contracts. These are agreements among mutually distrusting parties, which can be automatically enforced without resorting to a trusted intermediary. However, the existing informal, low-level descriptions, and the use of poorly documented Bitcoin features, pose obstacles to the research in this field. To overcome these issue we have recently proposed BitML, a high-level DSL for smart contracts with a computationally sound compiler to Bitcoin transactions. In this talk we start from BitML to investigate a landmark property of contracts, called liquidity: in a non-liquid contract, it may happen that some funds remain frozen. Liquidity is a relevant issue, as witnessed by a recent attack to the Ethereum Parity Wallet, which has frozen $\sim$160M USD within the contract, making this sum unredeemable by any user.

We develop a verification technique for liquidity of BitML contracts. To prove this, we first transform the infinite-state semantics of BitML into a finite-state one, which focusses on the behaviour of any given set of contracts, abstracting the moves of the context. With respect to the chosen contracts, this abstraction in sound and complete. Our decision procedure for liquidity is then based on model-checking the finite space of states of the abstraction.

Submitted to:
VPT 2019