Code that you can rely upon

Formal Land

INQUIRIES?

contact@formal.land
https://formal.land/

Formal Land

What we are formally verifying on the protocol of Tezos

DATA-ENCODINGS

We verify that the data-encoding primitives, which are functions translating values from the OCaml runtime to binary format and conversely, are well defined. This ensures there will be no data corruption when registering the blockchain state on disk.

NEXT

Our next target is the verification of the smart rollups mechanism, and in particular, the refutation game protocol. Our goal is to show that an honest player cannot lose the refutation game.

BACKWARD COMPATIBILITY

We verify that for two successive versions of the protocol of Tezos, the interpreter of Michelson smart contracts is backward compatible. This is important as it is not always possible to upgrade existing smart contracts, and a change in the semantics of the Michelson language could open a security breach.

NO INTERNAL ERRORS

We classify the protocol errors into two categories, user-related and internal. We show that internal errors cannot happen for any execution path. This verification effort covers most of the protocol code. We also check that the invariants of the various data types used in Tezos are preserved by each functions.

Project page:

formal-land.gitlab.io/coq-tezos-of-ocaml

How can we help Tezos to grow further?