Writing Smart Contracts
06 Smart Signatures

Peter H. Gruber

Supported by the Algorand Foundation

Smart signatures

What is a smart signature?

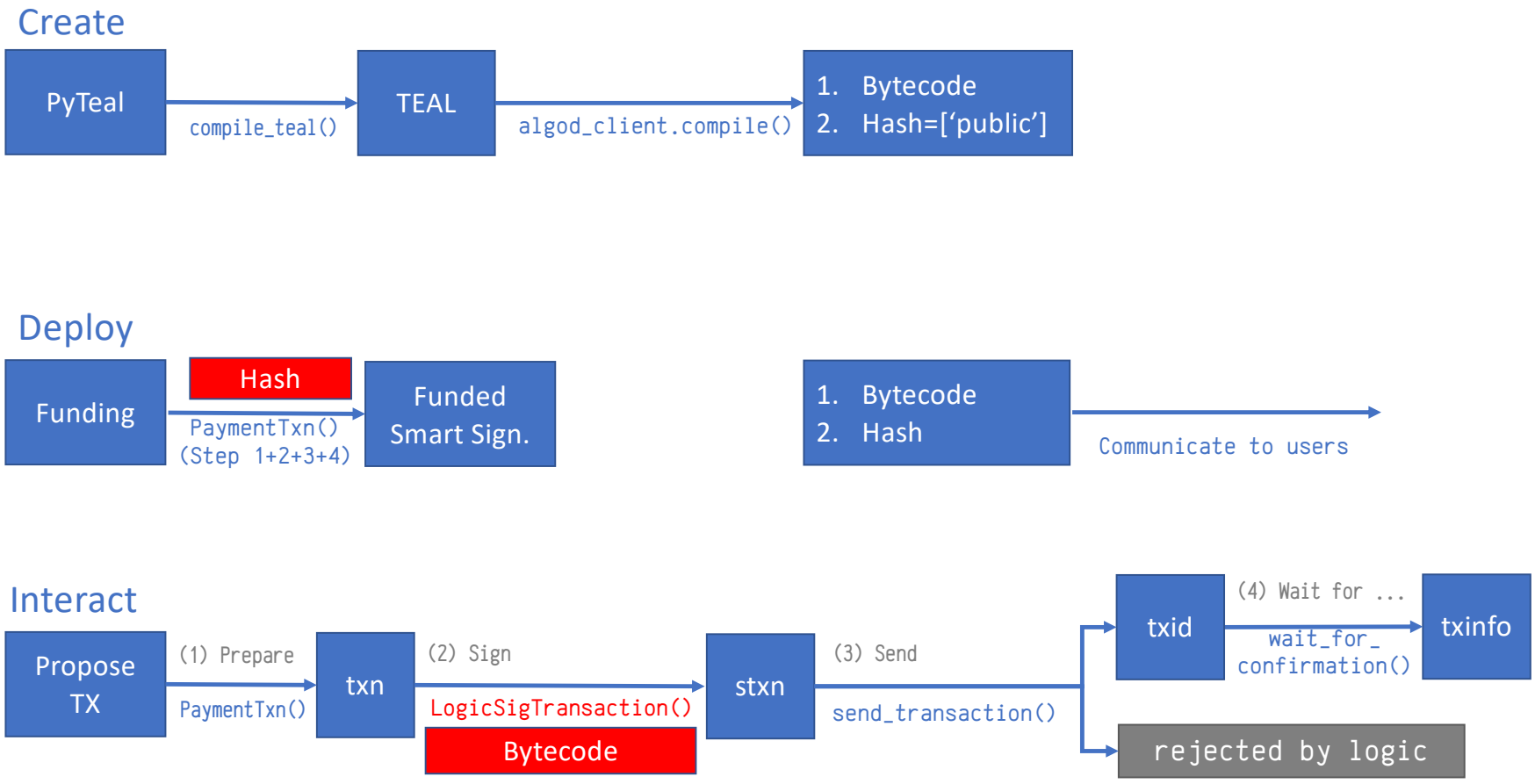
- A "yes" or "no" smart contract
- A function that produces only `True` or `False`
- Can accept or reject a **proposed transaction**

Arguments

- **Possible:** any property of the proposed TX
  - Receiver
  - Amount
  - First/last round
  - TX type (Payment or AssetTransfer)
  - TX note, TX arguments
  - Asset index
- **Not possible:** on-chain information
  - Asset amount of smart contract or receiver
  - Transaction history

`https://pyteal.readthedocs.io/en/stable/accessing_transaction_field.html`

Life of a smart signature



Life of a smart signature (2)

Deployment – funding

- Anyone can fund a smart signature
- Hash = public address calculated from program code
- Multiple funding operations possible

Deployment – communication

- Communicate to possible users of the smart signature:
  - Hash = public address
  - Result = program code
- Provide template (web) for acceptable transactions

Smart signature attacks

- Transaction fee attack
  - Propose transaction with excessive fee
  - Make smart sig loose money
- Rekey attack
  - Rekey smart sig to other address
  - Steal all money from smart sig
- Closeout attack
  - Add closeout to valid payment transaction
  - Steal all money from smart sig
- Cross-asset attack
  - Propose attack for a different ASA as intended
  - Obtain ASA much cheaper than intended
- Transaction group attack