Covenants

Harsha Goli, sometimes known as arshbot
Lightning Labs

What is a “Covenant”?

What is a “Covenant”?

- main antagonist in the “Halo” series?

What is a "Covenant"?

- main antagonist in the "Halo" series?
- a witches coven?

What is a “Covenant”?

- main antagonist in the “Halo” series?
- a witches coven?
- a bible thing?

Covenants are agreements 🤝

Covenants are agreements

Covenants are agreements on how to spend the bitcoin

Covenants are agreements

Covenants are agreements on how to spend the bitcoin

The recipient agrees by accepting the bitcoin with special spending rules

An amusingly bad idea

— Greg Maxwell

Some good ideas

Some good ideas

Hodl Chicken

Bitcoin Will

🐔 Hodl chicken

	Alice withdraws first	Bob withdraws first	Neither withdraw
Alice	😩	😊	😩
Bob	😊	😩	😩

📝 Bitcoin Will

	I die	+spouse cheats	refund
Me	💀	💀	😊
Spouse	😊	😩	-
Kids	😊	😊	-
Attorney	😊	😊	-

A tidal wave of approaches

OP_TAPLEAF_UPDATE_VERIFY

OP_CHECKTEMPLATEVERIFY

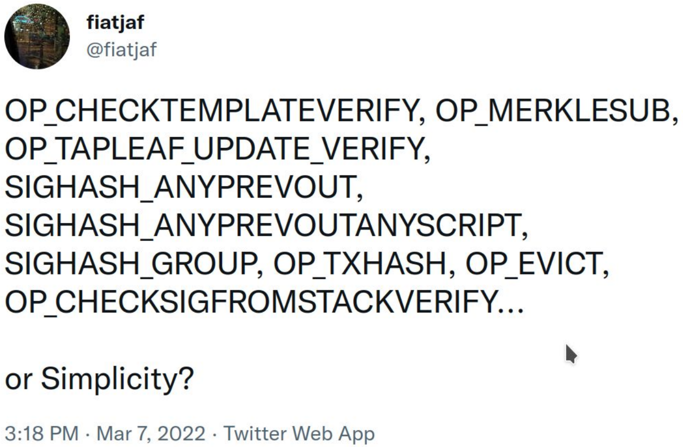
OP_CHECKSIGFROMSTACKVERIFY

OP_MERKLESUB

SIGHASH_ANYPREVOUT

OP_CAT

SIGHASH_GROUP



fiatjaf
@fiatjaf

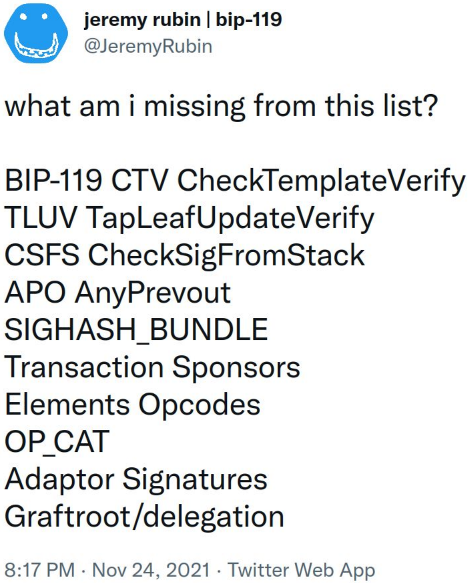
OP_CHECKTEMPLATEVERIFY, OP_MERKLESUB,
OP_TAPLEAF_UPDATE_VERIFY,
SIGHASH_ANYPREVOUT,
SIGHASH_ANYPREVOUTANYSCRIPT,
SIGHASH_GROUP, OP_TXHASH, OP_EVICT,
OP_CHECKSIGFROMSTACKVERIFY...

or Simplicity?

3:18 PM · Mar 7, 2022 · Twitter Web App

OP_EVICT

OP_TXHASH



jeremy rubin | bip-119
@JeremyRubin

what am i missing from this list?

BIP-119 CTV CheckTemplateVerify
TLUV TapLeafUpdateVerify
CSFS CheckSigFromStack
APO AnyPrevout
SIGHASH_BUNDLE
Transaction Sponsors
Elements Opcodes
OP_CAT
Adaptor Signatures
Graftroot/delegation

8:17 PM · Nov 24, 2021 · Twitter Web App

🪣 Two main buckets

🪣 Two main buckets

OPCode Based

OP_CTV

OP_COV

OP_PUSHTXDATA

OP_CD

🪣 Two main buckets

OPCode Based

OP_CTV OP_PUSHTXDATA

OP_COV OP_CD

Signature Based

OP_CSFS + OP_CAT + OP_TX

SIGHASH_ANYPREVOUT + NO_INPUT

OPCode based bois

OP_CHECKTEMPLATEVERIFY (OP_CTV)

Conceptually simple - checks if the spending transaction fits the specified template

OP_CHECKTEMPLATEVERIFY (OP_CTV)

Conceptually simple - checks if the spending transaction fits the specified template

The spending transaction/template transaction comparison is performed by hashing all of transaction’s relevant bits (defined by the template)

OP_CHECKTEMPLATEVERIFY (OP_CTV)

Important Bits!

LocktimeVersion

Sequence Hash

ScriptSig Hash

Number of Inputs

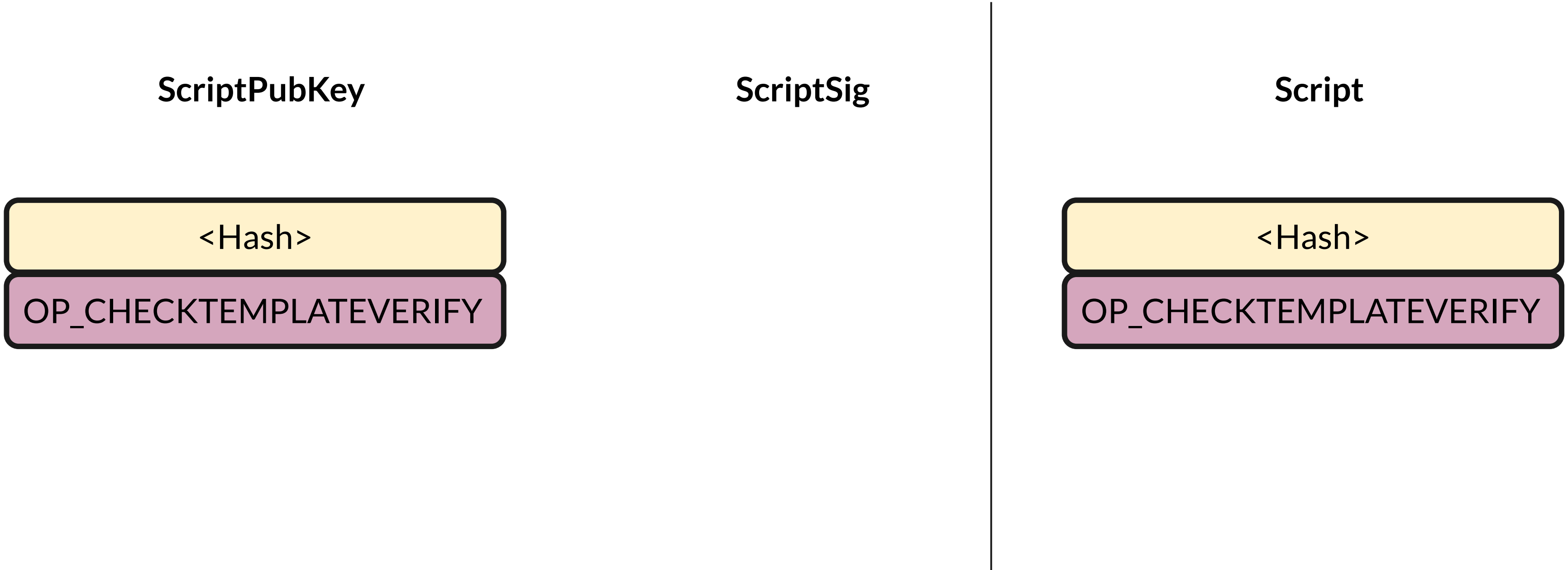
Number of Outputs

Outputs hash

Current input index

Values by Hash

Example



ScriptPubKey

<Hash>

OP_CHECKTEMPLATEVERIFY

ScriptSig

Script

<Hash>

OP_CHECKTEMPLATEVERIFY

OP_CHECKOUTPUTVERIFY

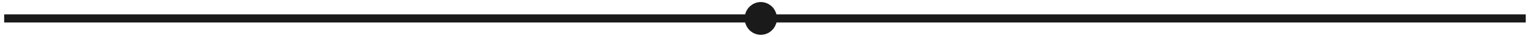
- OP_COV allows a user to specify a pattern and output index of the spending transaction
- Suffers from half spend problem (doesn’t restrict inputs to one)
- Allows recursive covenants (bad)

OP_CONSTRAINDESTINATION

Collection of 4 op_codes

- OP_CD — X amount to Address
- OP_BBV — expires spend path after block
- OP_POS — pops data only if specific address is output
- OP_LFC — constrains fee

Flexibility Scale



less flexi

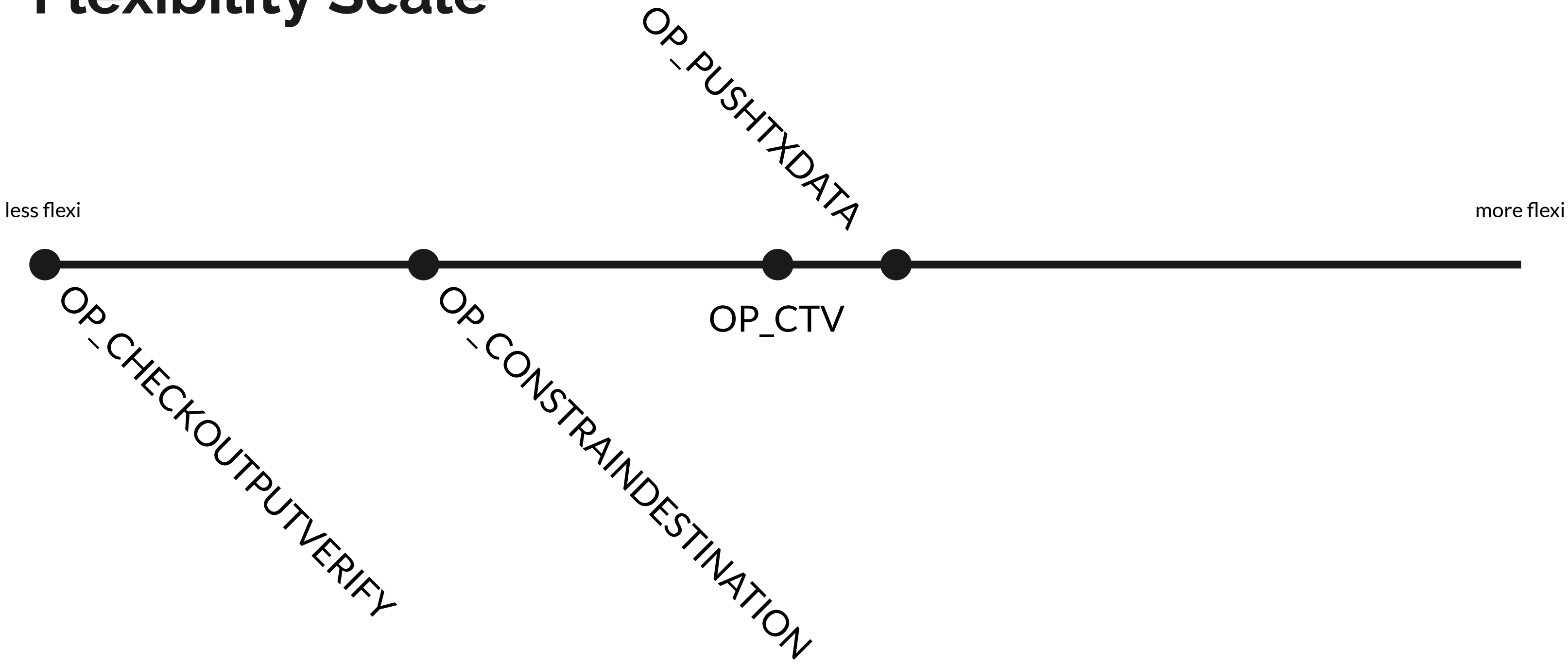
more flexi

OP_CTV

OP_PUSHTXDATA

- outputs must follow a pre-specified pattern (like op_ctv)
- Allows for fined tuned control of the spending transaction (unlike op_ctv)
- Allows control over:
  - Fees
  - transaction size with and without witness serialization
  - transaction weight

Flexibility Scale



less flexi

more flexi

OP_CHECKOUTPUTVERIFY

OP_CONSTRAINDESTINATION

OP_CTV

OP_PUSHTXDATA

Signature based bois

OP_CHECKSIGFROMSTACK + OP_CAT + OP_TXHASH

OP_CSFS

- Checks if a signature signs an arbitrary message
- arbitrary message can be a transaction
- Could be used as a more flexible OP_CTV

OP_CAT

- Removed once because of a bug that created 184 Billion bitcoin in a transaction due to an overflow bug
- Concats two elements, useful for constructing messages
- Often included to proposals OP_CHECKSIGFROMSTACK proposals

OP_TXHASH

- Simple functionality that is kind of like OP_CTV
- Computes tagged txhash and pushes onto stack

or

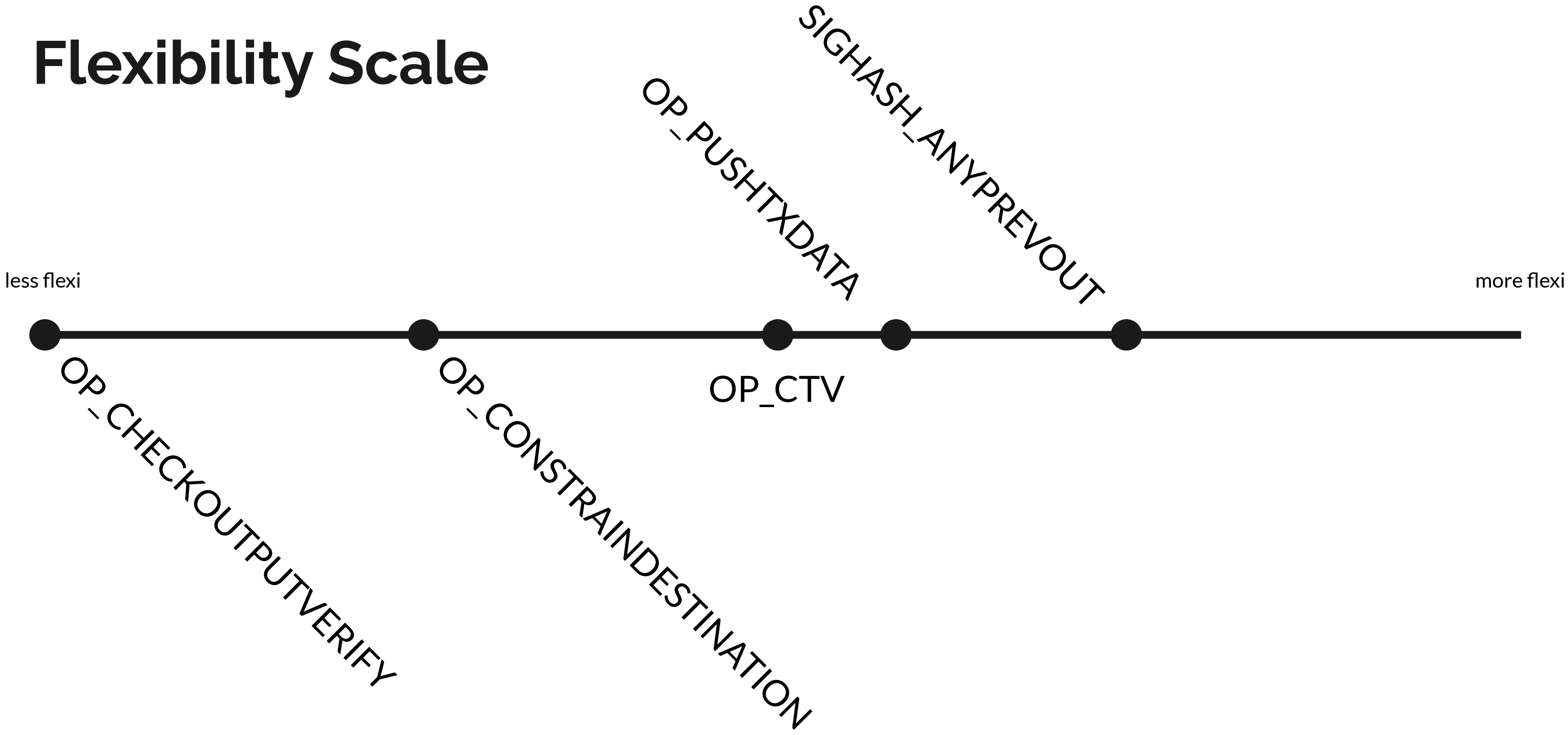
OP_TX

- Simplified alternative to OP_TXHASH
- breakdown of the steps in OP_TXHASH since something similar can be accomplished with OP_SHA256 and OP_TX

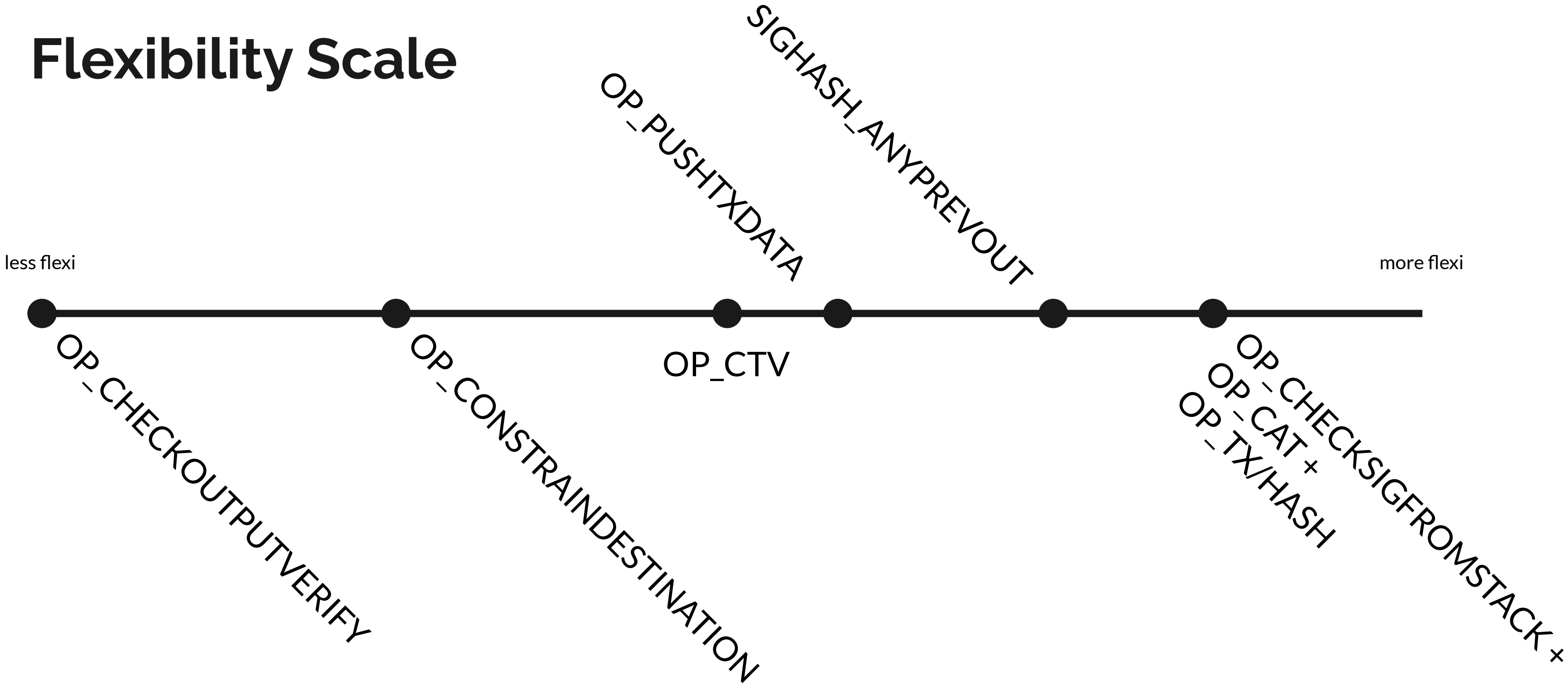
SIGHASH_ANYPREVOUT

- New kind of public key for tapscript transactions
- Allows you to create a signature for which only a specific spending transaction can fulfill
- sighash where the identifier for the UTXO being spent is not signed, allowing the signature to be used with any UTXO that's protected by a similar script
- Necessary for eltoo
- with OP_CAT main proposal can be compatible with schnorr

Flexibility Scale



Flexibility Scale



Wrapping up

Wrapping up

What we've learned

- Covenants are *just* agreements
- Future is rife with possibilities
- Loads of proposals, 2 main groups
- OP_CTV has the most support
- Signature based is more flexible but at a cost

Thank you!

Cheers,
Harsha Goli, sometimes known as arshbot
tw _arshbot