Watchtowers and BOLT#13

Sergi Delgado

Talaia Labs

ONE STEP BACK

What is the general paradigm behind third party watching systems (**AKA Watchtowers**)?

User:

- Sends **data** to the server alongside a **trigger** condition

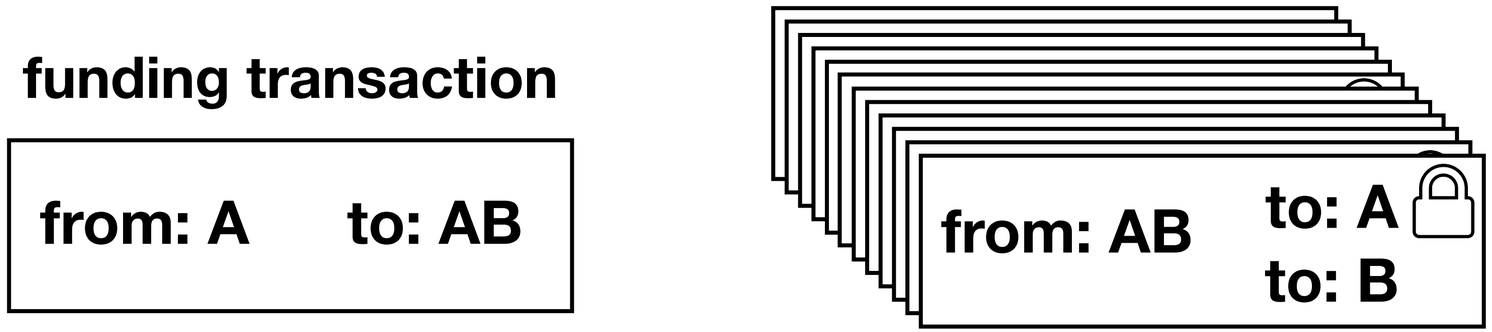
Server:

- **Looks for triggers** on a communication channel
- If the a trigger is seen, perform **an action** with the provided data

LIGHTNING TRANSACTIONS IN 1 MIN

commitment transactions

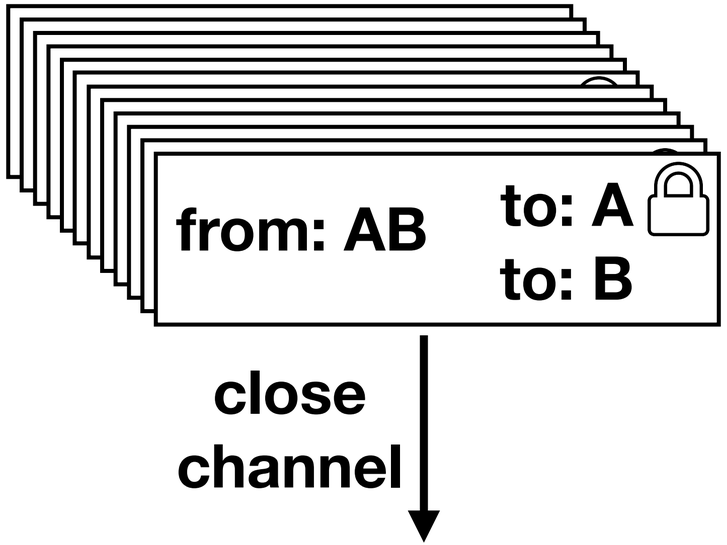
funding transaction



LIGHTNING TRANSACTIONS IN 1 MIN

commitment transactions

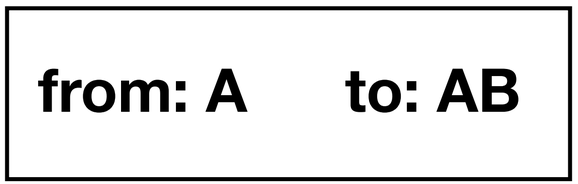
funding transaction

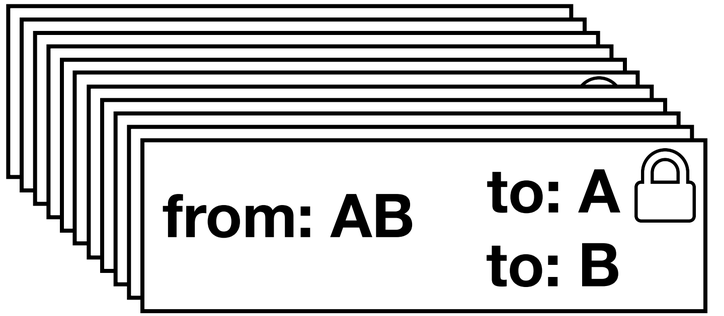


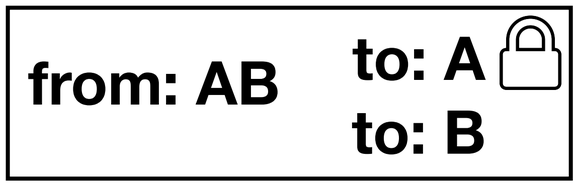
LIGHTNING TRANSACTIONS IN 1 MIN

commitment transactions

funding transaction





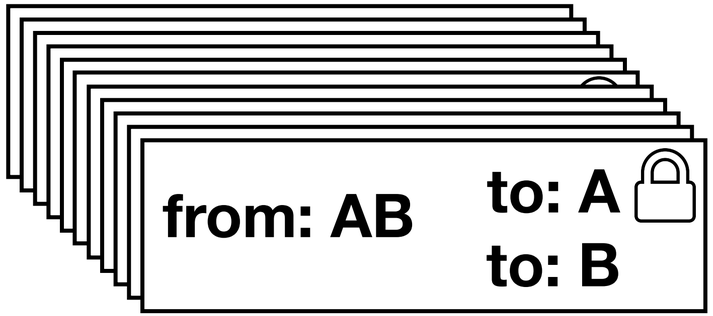


LIGHTNING TRANSACTIONS IN 1 MIN

commitment transactions

funding transaction

from: A	to: AB



close channel

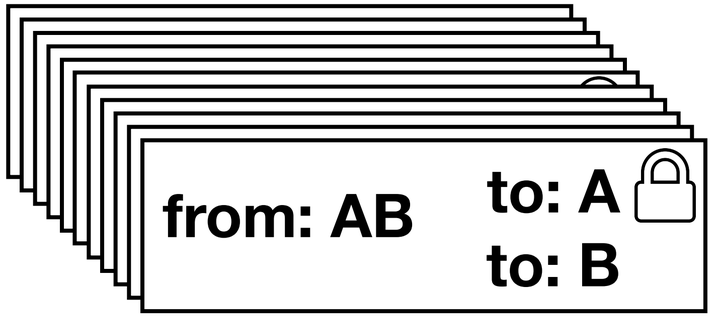
from: AB	to: A to: B

LIGHTNING TRANSACTIONS IN 1 MIN

commitment transactions

funding transaction

from: A	to: AB



close channel

from: AB	to: A to: B

closing transaction

LIGHTNING TRANSACTIONS IN 1 MIN

commitment transactions

funding transaction

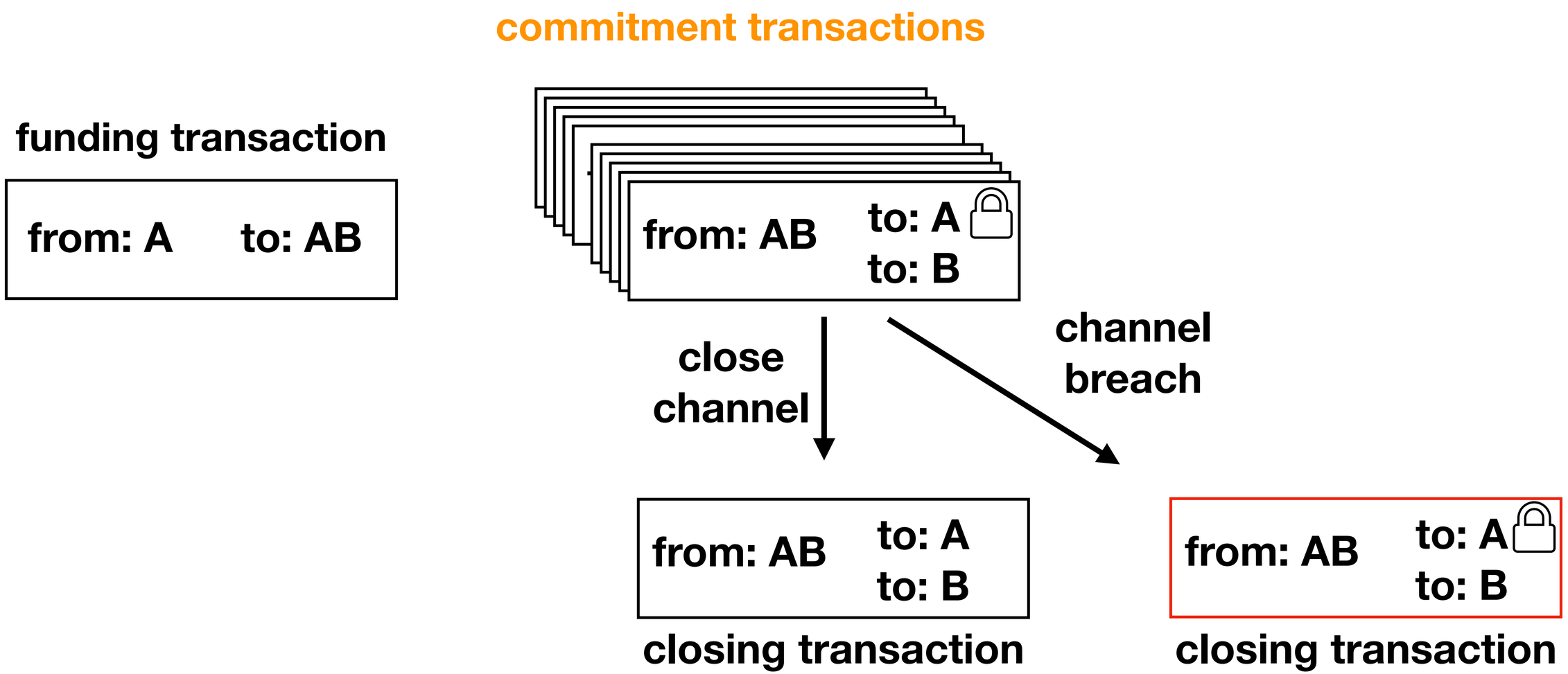
from: A	to: AB

close channel

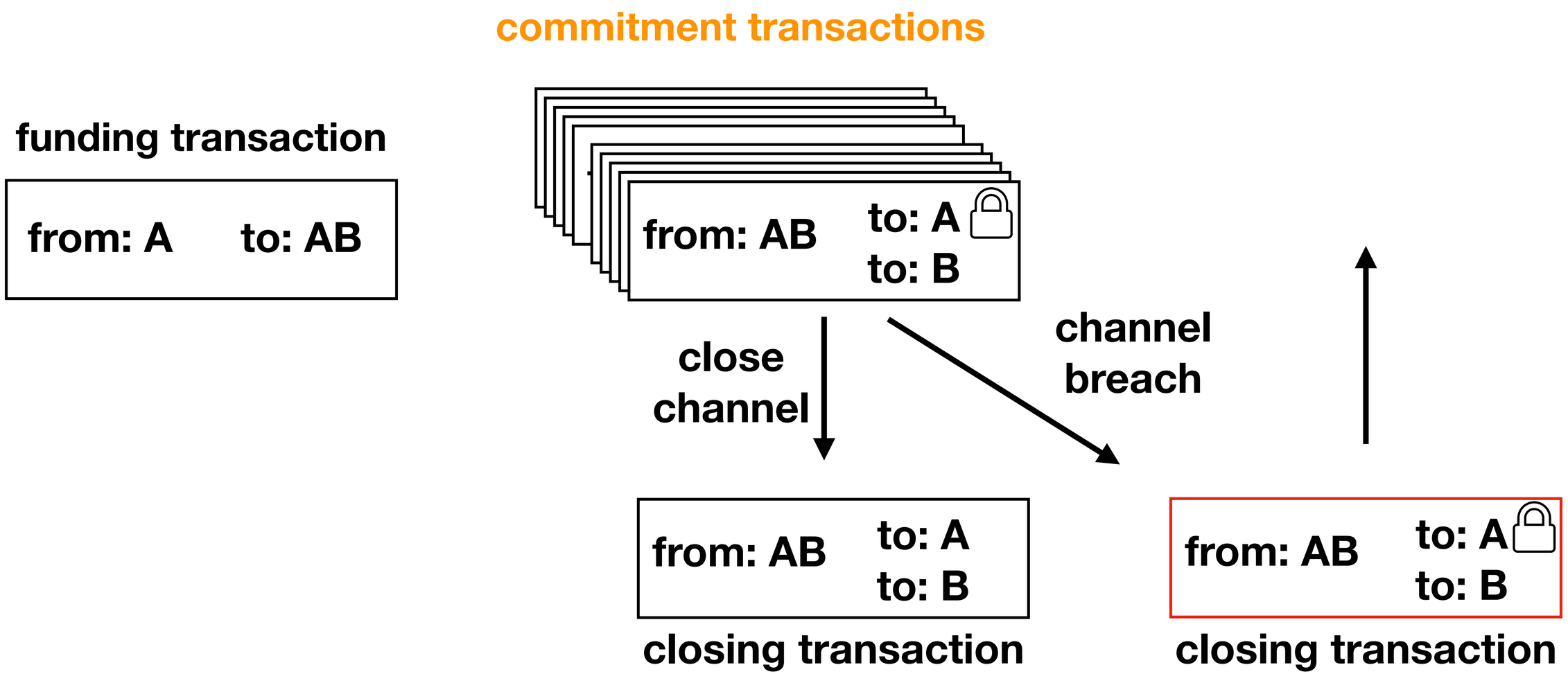
from: AB	to: A to: B

closing transaction

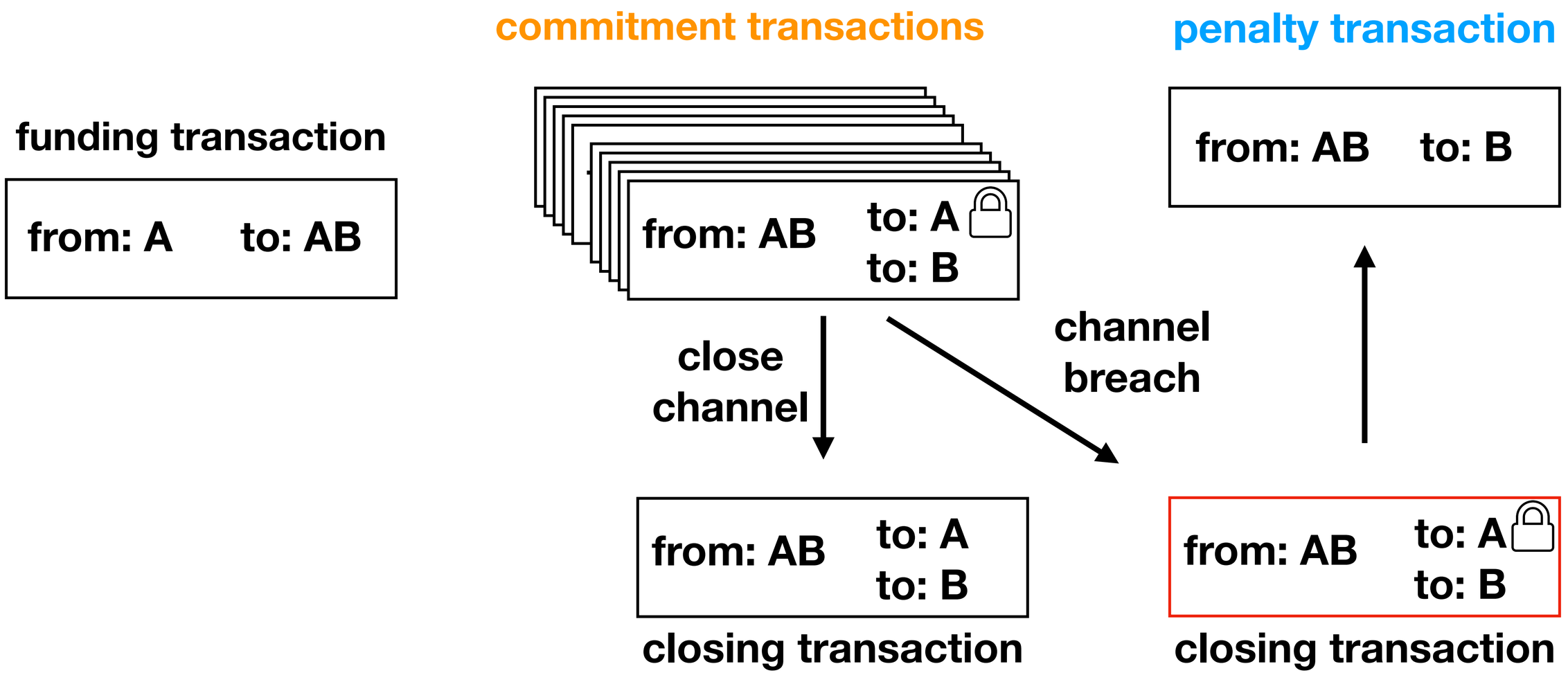
LIGHTNING TRANSACTIONS IN 1 MIN



LIGHTNING TRANSACTIONS IN 1 MIN



LIGHTNING TRANSACTIONS IN 1 MIN



BASIC WATCHTOWER PROTOCOL

BASIC WATCHTOWER PROTOCOL

BASIC WATCHTOWER PROTOCOL

BASIC WATCHTOWER PROTOCOL

[...]
commitment_txid,
penalty_tx,
[...]

BASIC WATCHTOWER PROTOCOL

[...]
commitment_txid,
penalty_tx,
[...]

BASIC WATCHTOWER PROTOCOL

[...]
commitment_txid,
penalty_tx,
[...]

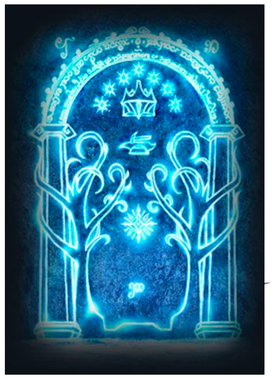
appointment

BASIC WATCHTOWER PROTOCOL

[...]
commitment_txid,
penalty_tx,
[...]

appointment

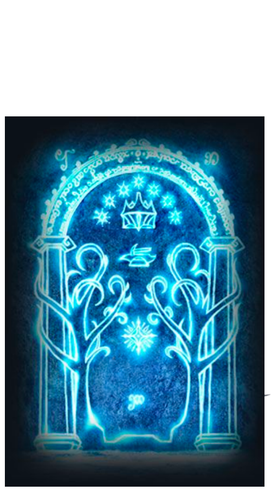
BASIC WATCHTOWER PROTOCOL



[...]
commitment_txid,
penalty_tx,
[...]

appointment

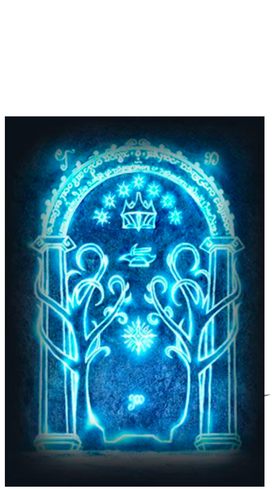
BASIC WATCHTOWER PROTOCOL



[...]
commitment_txid,
penalty_tx,
[...]

appointment

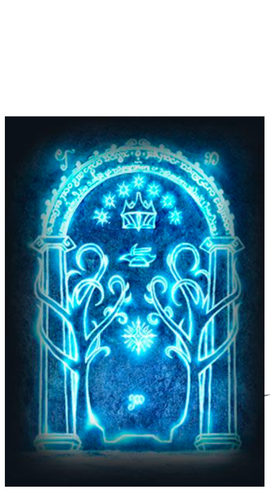
BASIC WATCHTOWER PROTOCOL



[...]
commitment_txid,
penalty_tx,
[...]

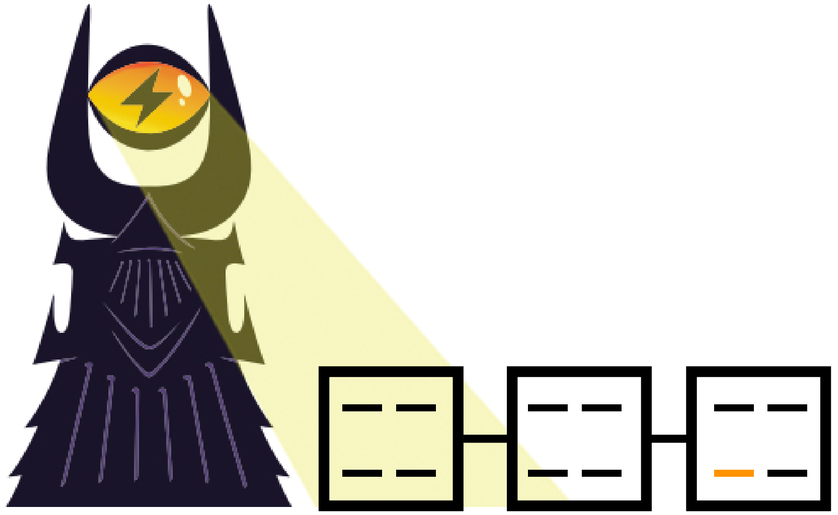
appointment

BASIC WATCHTOWER PROTOCOL

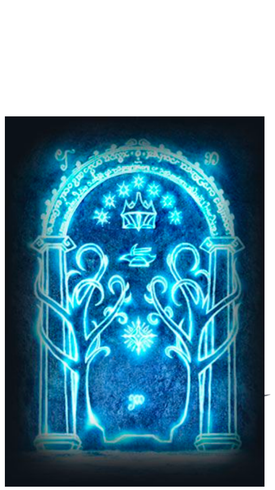


[...]
commitment_txid,
penalty_tx,
[...]

appointment

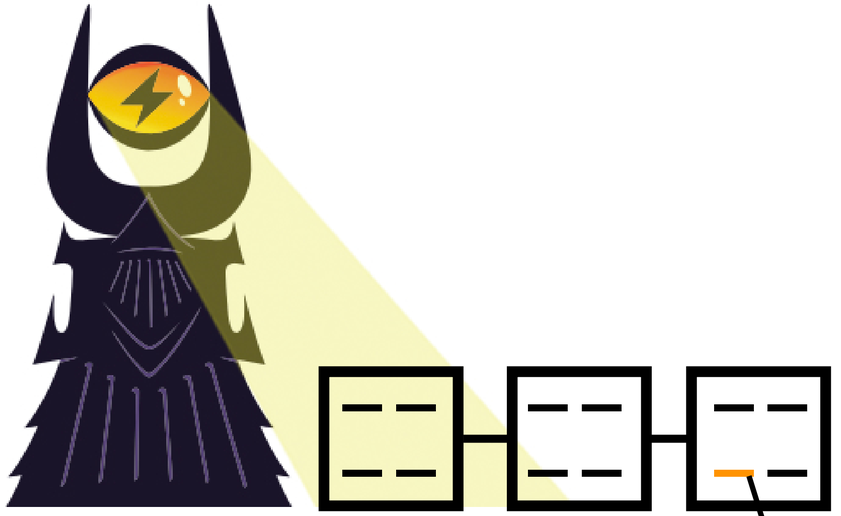


BASIC WATCHTOWER PROTOCOL



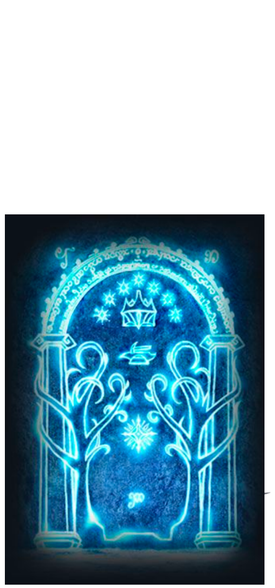
[...]
commitment_txid,
penalty_tx,
[...]

appointment



commitment_txid

BASIC WATCHTOWER PROTOCOL

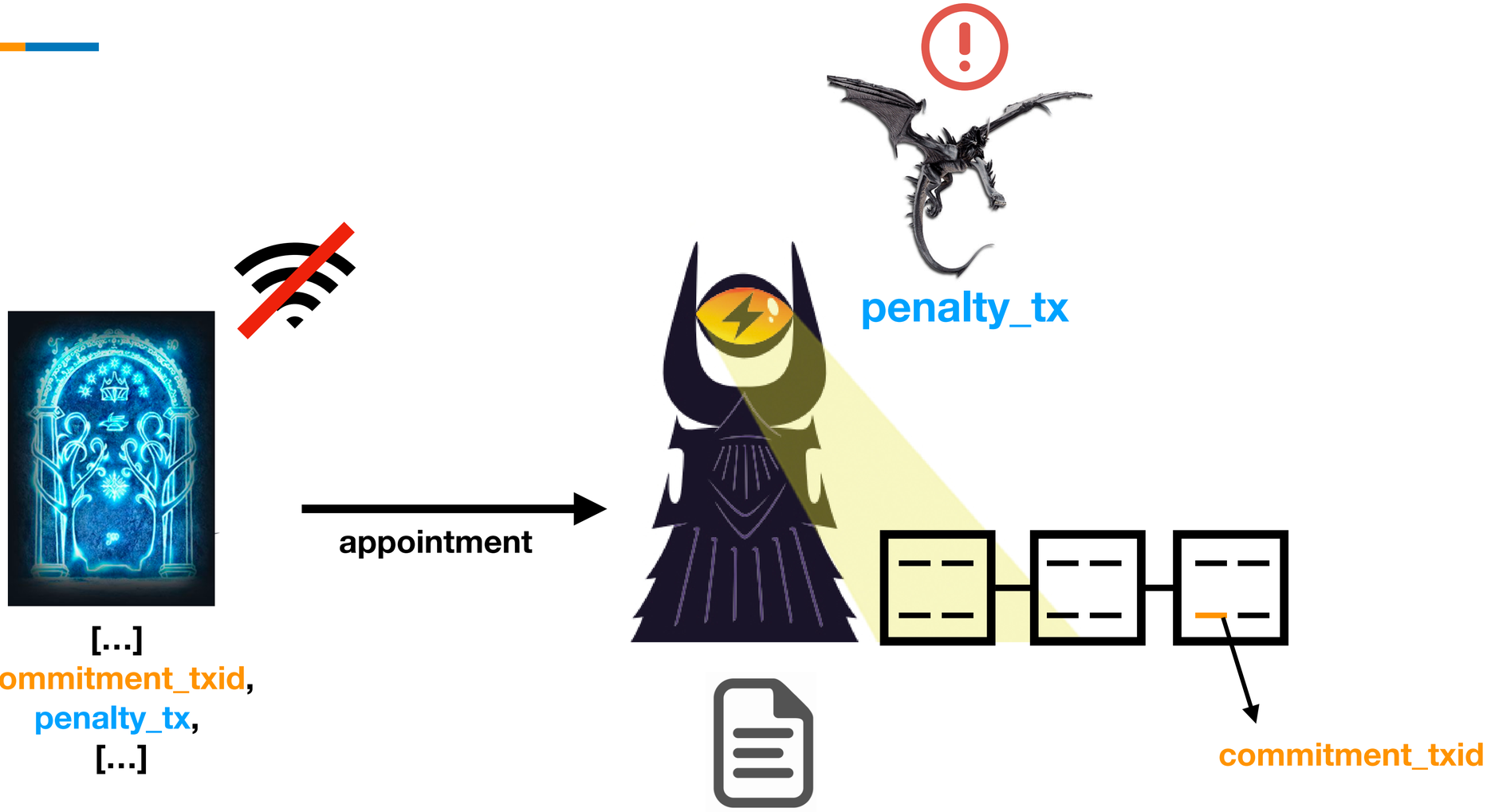


[...]
commitment_txid,
penalty_tx,
[...]

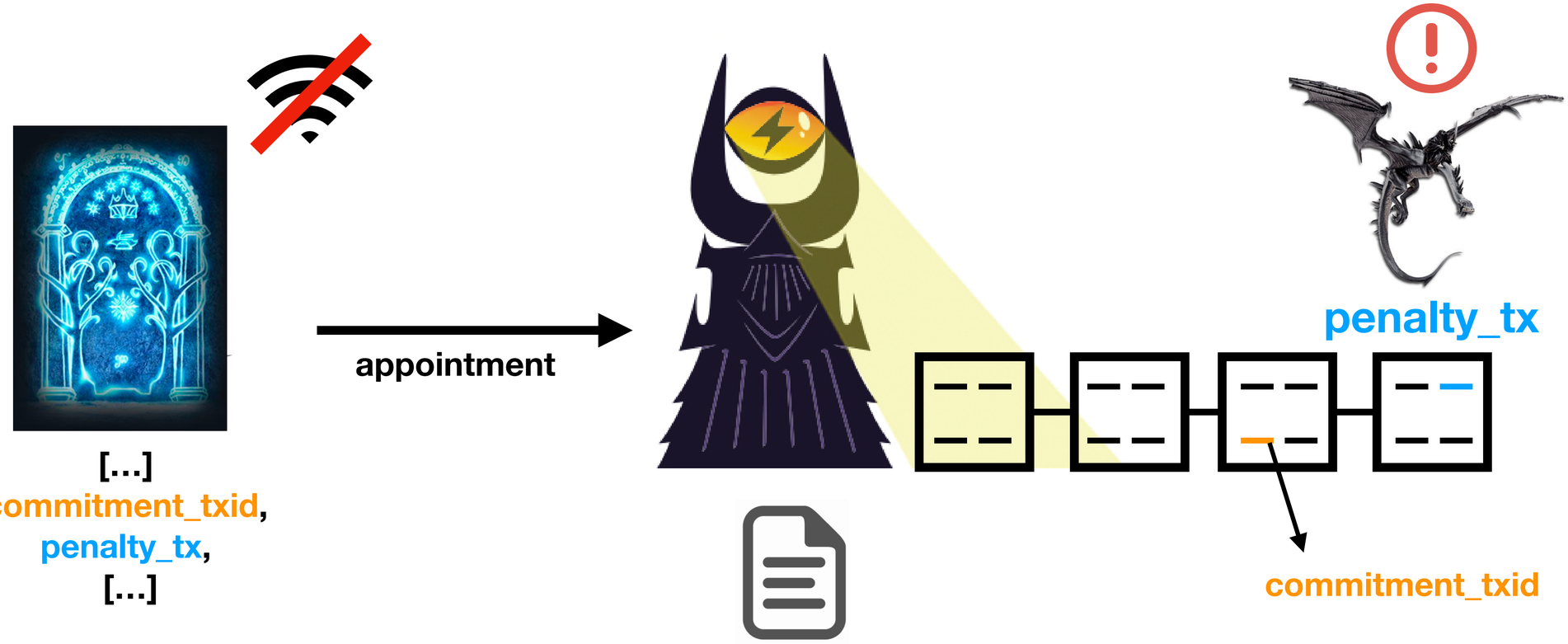
appointment

commitment_txid

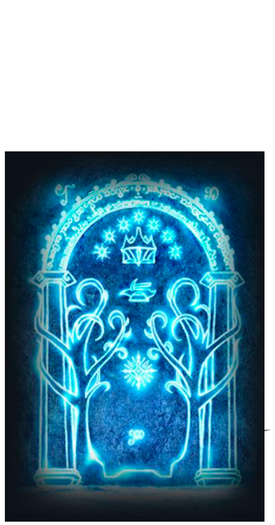
BASIC WATCHTOWER PROTOCOL



BASIC WATCHTOWER PROTOCOL



BASIC WATCHTOWER PROTOCOL



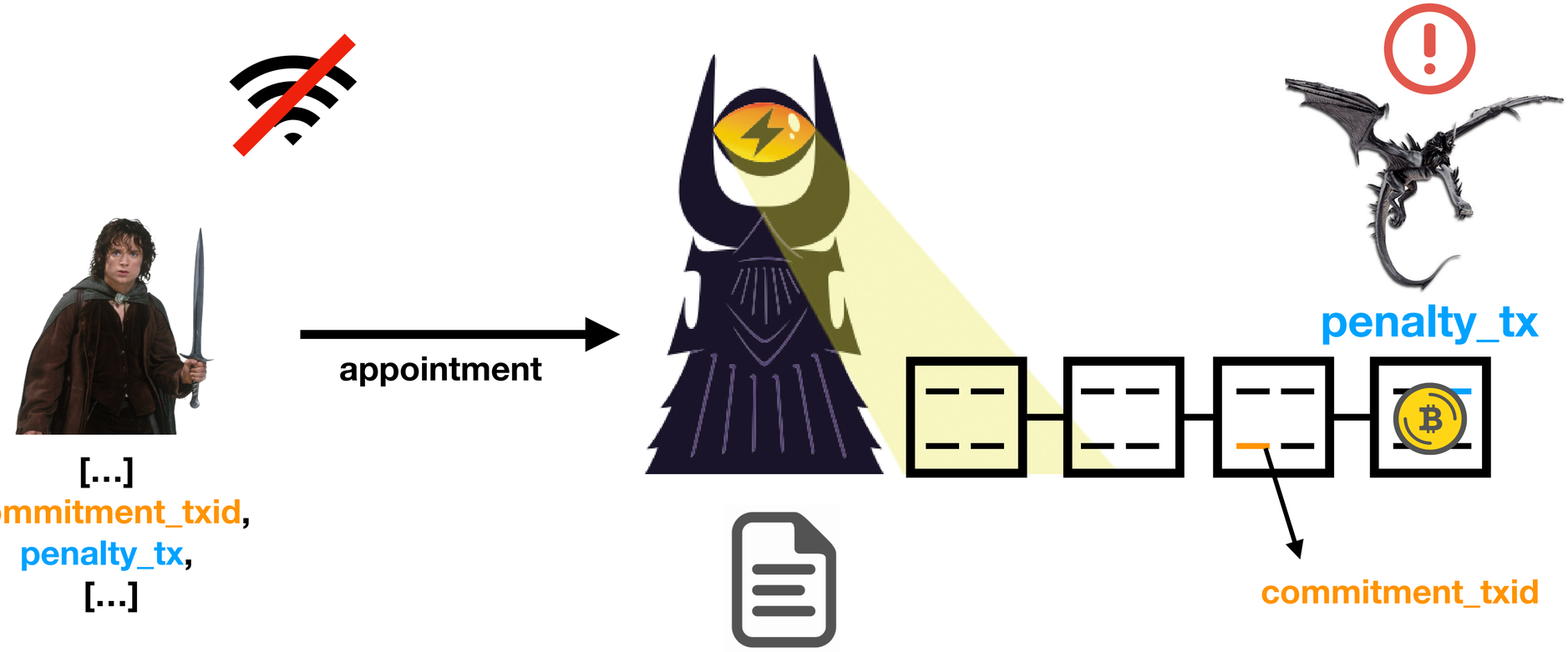
[...]
commitment_txid,
penalty_tx,
[...]

appointment

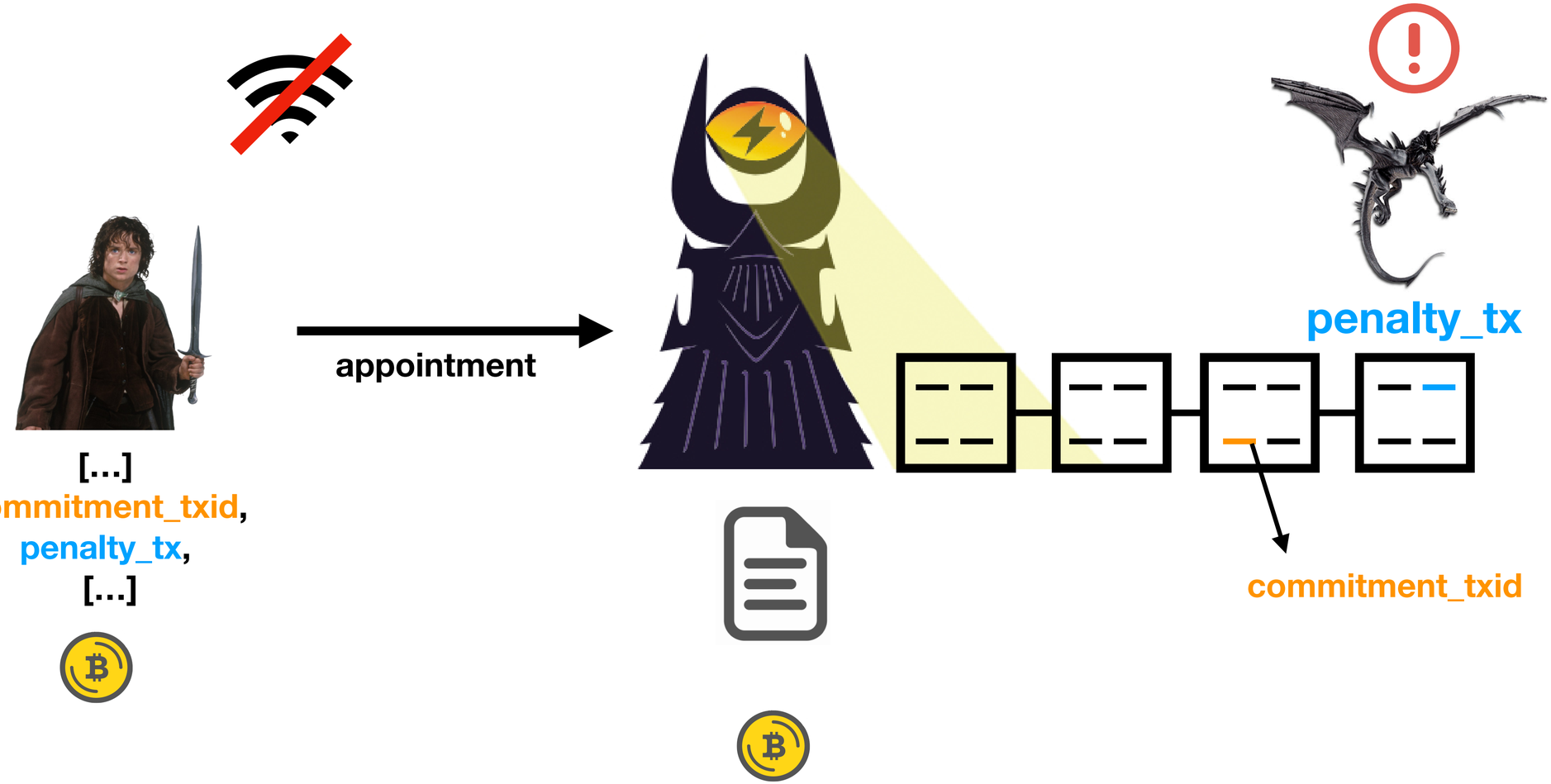
penalty_tx

commitment_txid

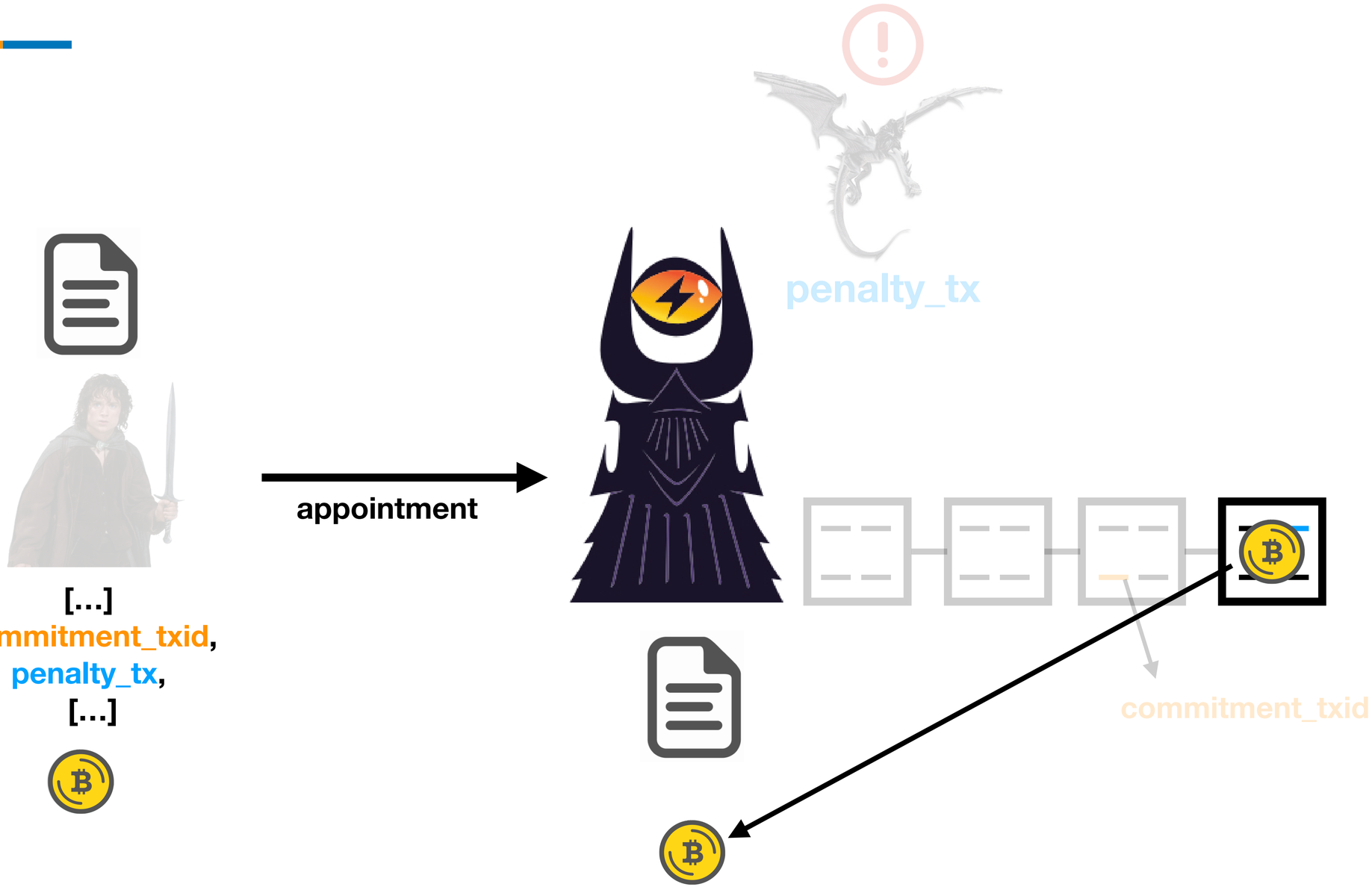
BASIC WATCHTOWER PROTOCOL



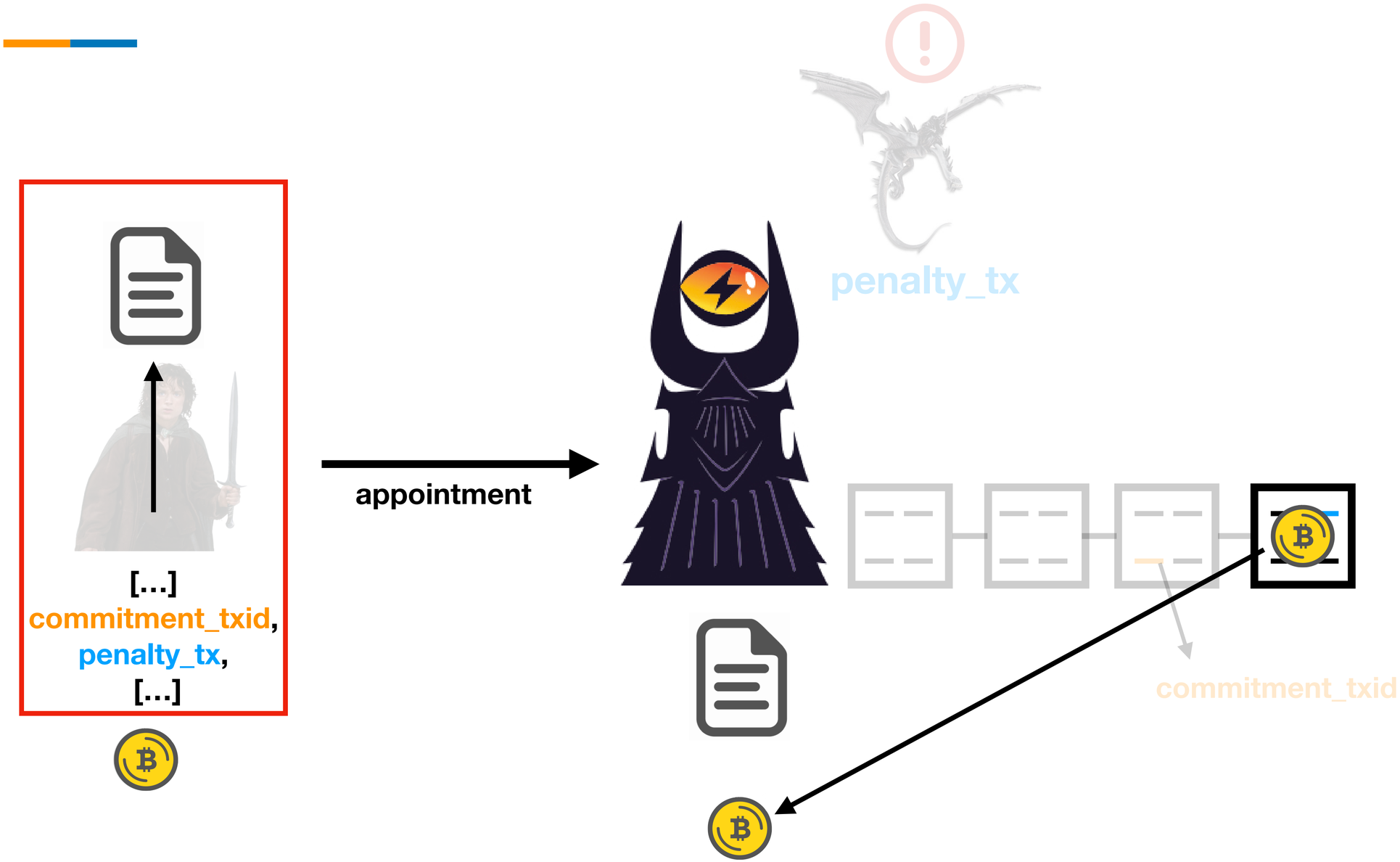
BASIC WATCHTOWER PROTOCOL



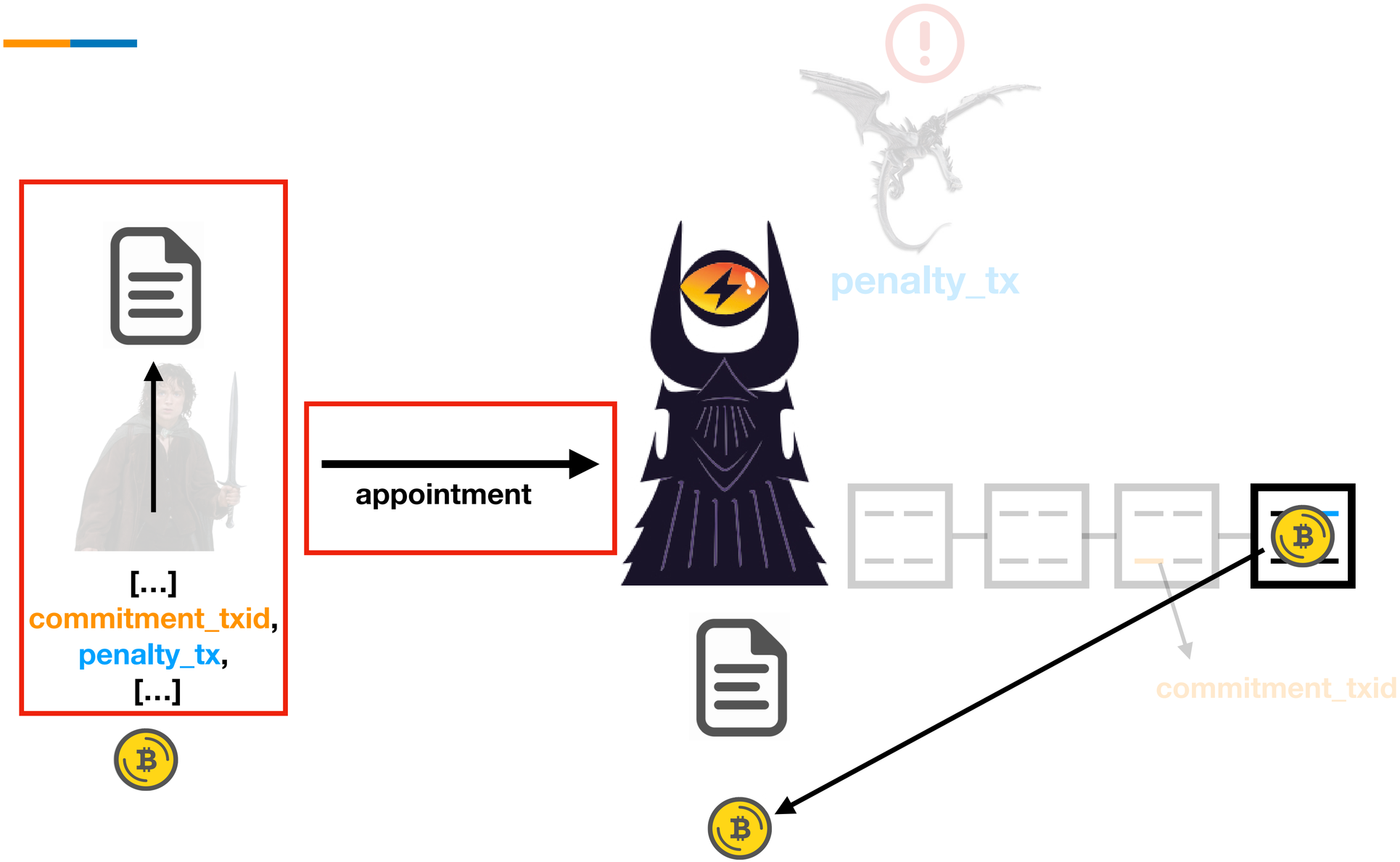
BASIC WATCHTOWER PROTOCOL



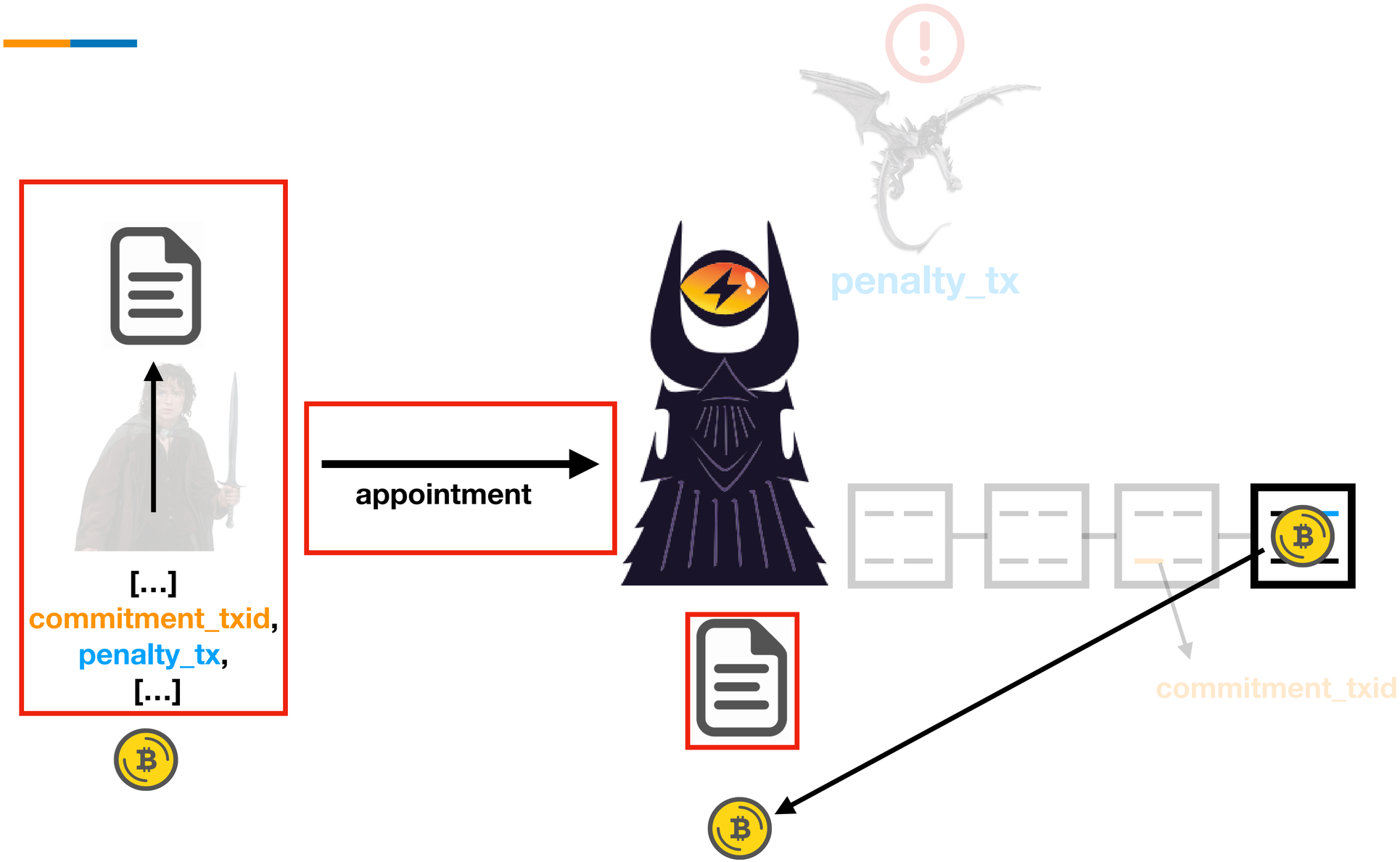
BASIC WATCHTOWER PROTOCOL



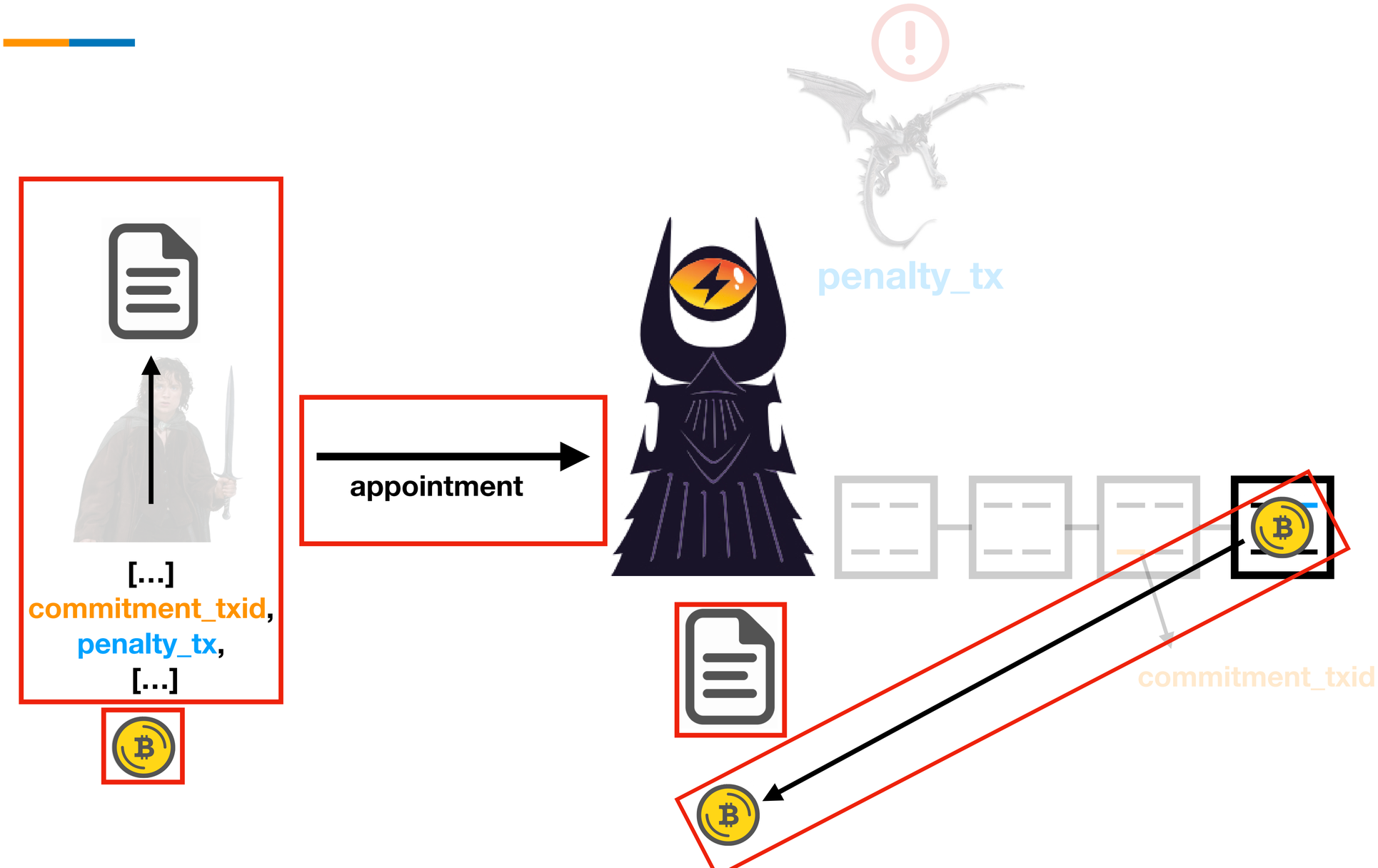
BASIC WATCHTOWER PROTOCOL



BASIC WATCHTOWER PROTOCOL



BASIC WATCHTOWER PROTOCOL



(NON-CUSTODIAL) WATCHTOWER DESIGN TRADEOFFS

PRIVACY

What does the Watchtower know about the node?

ACCESS

Who can use the Watchtower?

STORAGE

What does the Watchtower have to store?

COST

What is the user cost to use the Watchtower?

NO PRIVACY VS FULL PRIVACY

NO PRIVACY

The user sends the penalty transaction as clear text

- ✅ Can verify data is a transaction
- ❌ Cannot verify transaction is valid
- ❌ Payment information is leaked

FULL PRIVACY

The user sends an encrypted penalty transaction

- ✅ Data only leaked on breach (less)
- ❌ Cannot verify data is a transaction
- ❌ Heavier computation

NO PRIVACY VS FULL PRIVACY

NO PRIVACY

FULL PRIVACY

✘Useless information can be sent to the tower

NO PRIVACY VS FULL PRIVACY

NO PRIVACY

FULL PRIVACY

✘Useless information can be sent to the tower

STORAGE

NO PRIVACY VS FULL PRIVACY

NO PRIVACY

FULL PRIVACY

✘Useless information can be sent to the tower

STORAGE

Therefore, privacy by design seems a better approach

PRIVATE VS PUBLIC ACCESS

PRIVATE ACCESS

A limited number of (trusted) users can use the tower

- ✅ No DoS risk
- ✅ Potentially free service
- ❌ Can’t accommodate the whole network

PUBLIC ACCESS

Anyone can use the tower

- ✅ Tower as a service
- ➖ Access control required
- ❌ Paid service (high DoS surface if not properly priced)

PRIVATE VS PUBLIC ACCESS

PRIVATE ACCESS

A limited number of (trusted) users can use the tower

LOW STORAGE

LOW / NO COST

PUBLIC ACCESS

Anyone can use the tower

- ✅ Tower as a service
- ➖ Access control required
- ❌ Paid service (High DoS surface if not properly priced)

PRIVATE VS PUBLIC ACCESS

PRIVATE ACCESS

A limited number of (trusted) users can use the tower

LOW STORAGE

LOW / NO COST

PUBLIC ACCESS

Anyone can use the tower

HIGH STORAGE

LOW COST

O(N) STORAGE

STORAGE 

The required storage is always going to be big (modulo the number of channel updates).

- Highly linked to price
- Strategies to align the incentives of the user and the tower are required
- ✘ One appointment per channel update

ALTRUISTIC VS NON-ALTRUISTIC TOWERS

NO COST

Using the tower is free

- ✅ OK for private towers
- ❌ Highly unviable for public towers (highest cost and DoS surface)

LOW COST

The tower charges a fee

- ✅ High traffic = profit **(if properly priced)**
- ✅ Data can be deleted **(if incentives are aligned)**

ALTRUISTIC VS NON-ALTRUISTIC TOWERS

NO COST

Using the tower is free

PRIVATE ACCESS AND LOW STORAGE

LOW COST

The tower charges a fee

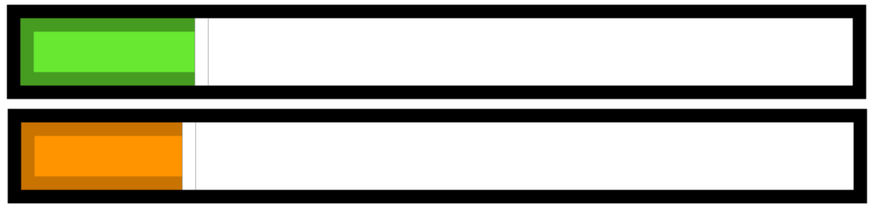
- ✅ High traffic = profit **(if properly priced)**
- ✅ Data can be deleted **(if incentives are aligned)**

ALTRUISTIC VS NON-ALTRUISTIC TOWERS

NO COST

Using the tower is free

PRIVATE ACCESS AND LOW STORAGE



OR
PUBLIC ACCESS AND &^%$ STORAGE

LOW COST

The tower charges a fee

- ✅ High traffic = profit
  (if properly priced)
- ✅ Data can be deleted
  (if incentives are aligned)

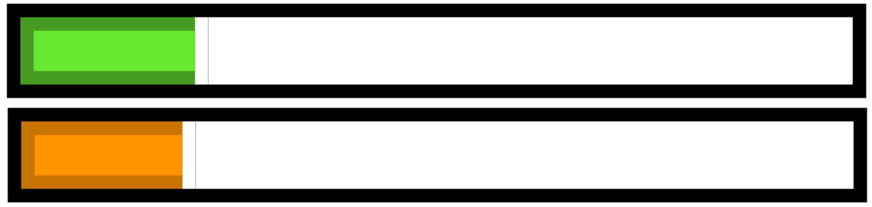
ALTRUISTIC VS NON-ALTRUISTIC TOWERS

NO COST

LOW COST

Using the tower is free

PRIVATE ACCESS AND LOW STORAGE



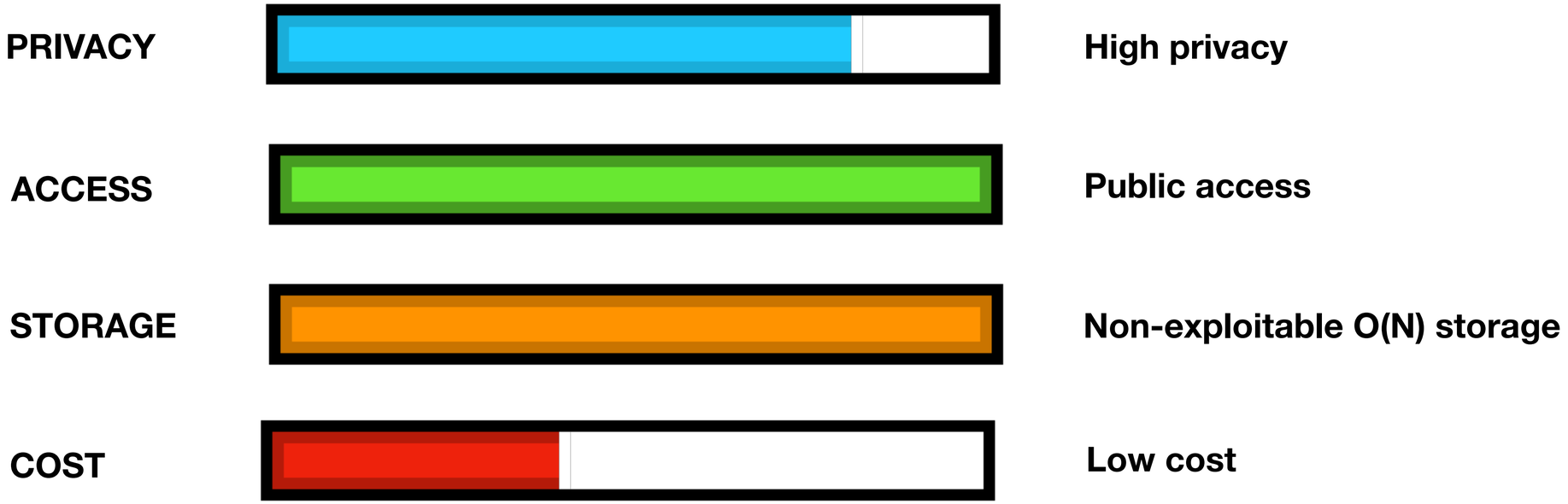
OR

PUBLIC ACCESS AND &^%$ STORAGE

The tower charges a fee

HIGH STORAGE

IDEAL WATCHTOWER (NO ELTOO)



IDEAL WATCHTOWER (NO ELTOO)

PRIVACY

High privacy

ACCESS

Public access

STORAGE

Non-exploitable O(N) storage

COST

Low cost

INTEROPERABLE!

BOLT#13

PRIVACY VIA MONITOR APPROACH (1/3)

For every channel update:

- The penalty transaction is **encrypted** under a key derived from the commitment transaction id **(sk and iv)**
- A **locator** is also derived from the commitment transaction id
- The tower receives the **encrypted blob and the locator**

Tadge Dryja - Unlinkable Outsourced Channel Monitoring - Scaling Bitcoin 2016

PRIVACY VIA MONITOR APPROACH (2/3)

User side

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB → **locator**

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB → **locator**

cipher = CHACHA20POLY1305

sk = SHA256(commitment_txid)

IV = 0

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB → **locator**

cipher = CHACHA20POLY1305

sk = SHA256(commitment_txid)

IV = 0

encrypt (penalty_tx, sk, IV) →

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB → **locator**

cipher = CHACHA20POLY1305

sk = SHA256(commitment_txid)

IV = 0

encrypt (penalty_tx, sk, IV) → **encrypted blob**

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB → **locator**

cipher = CHACHA20POLY1305

sk = SHA256(commitment_txid)

IV = 0

encrypt (penalty_tx, sk, IV) → **encrypted blob**

PRIVACY VIA MONITOR APPROACH (2/3)

User side

penalty_tx = 020000000001010d8b7512b1f530338ca886…1f9624914fb8a6800000000

commitment_txid = 4a5e1e4baab89f3a32518...cc77ab2127b7afdeda33

16 MSB → **locator**

SENT TO THE TOWER

cipher = CHACHA20POLY1305

sk = SHA256(commitment_txid)

IV = 0

encrypt (penalty_tx, sk, IV) → **encrypted blob**

PRIVACY VIA MONITOR APPROACH (3/3)

Tower side

PRIVACY VIA MONITOR APPROACH (3/3)

Tower side

for every transaction_id in every block

locator = 16 MSB transaction_id

PRIVACY VIA MONITOR APPROACH (3/3)

Tower side

for every transaction_id in every block

locator = 16 MSB transaction_id

if locator in appointments:

sk = SHA256(transaction_id)

IV = 0

PRIVACY VIA MONITOR APPROACH (3/3)

Tower side

for every transaction_id in every block

locator = 16 MSB transaction_id

if locator in appointments:

sk = SHA256(transaction_id)

IV = 0

decrypt (encrypted blob, sk, IV)

PRIVACY VIA MONITOR APPROACH (3/3)

Tower side

for every transaction_id in every block

locator = 16 MSB transaction_id

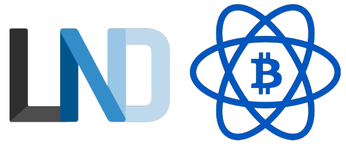
if locator in appointments:

sk = SHA256(transaction_id)

IV = 0

decrypt (encrypted blob, sk, IV) → **penalty_tx**

REVENUE MODELS

Bounty 

The penalty transaction includes an output for the tower.

Per-appointment

The tower is paid beforehand, appointment per appointment.

Subscription 

A subscription is paid to the tower that grants access to the user for a certain time/number of appointments.

BOUNTY - REVENUE MODELS

The tower is paid only if a breach happens and the penalty makes it to the chain	✓	✗
Multiple towers can be hired for the price of one	✓	✗
The tower **can** use CPFP to bump the fee of the penalty transaction	✓	✓
It’s easy to spam/DoS the tower with junk	–	✗

PER-APPOINTMENT - REVENUE MODELS

The tower is paid beforehand, even if the it does not responds to the breach	✗	✓
A rational user will only hire so many towers	–	✓
The tower **cannot** use CPFP to bump the fee of the penalty transaction	✗	✗
Spamming the tower has a cost	–	✓
A payment is required for every update	✗	–
Easily exploitable due lack of entry cost	–	✗

SUBSCRIPTION - REVENUE MODELS

The tower is paid beforehand, even if the it does not responds to the breach	✗	✓
A rational user will only hire so many towers	–	✓
The tower **cannot** use CPFP to bump the fee of the penalty transaction	✗	✗
Spamming the tower has a cost	–	✓
Minimises number payment to the tower	✓	–
Exploiting requires paying for a subscription	–	✓

SUBSCRIPTIONS VS BOUNTY

BOTH MODELS HAVE THEIR PROS AND CONS…

SUBSCRIPTIONS VS BOUNTY

BOTH MODELS HAVE THEIR PROS AND CONS…

SUBSCRIPTION & BOUNTY - REVENUE MODELS

The tower is paid **a fraction of the cost** beforehand, the rest is paid as a bounty	–	–
A rational user will only hire so many towers	–	✓
The tower **can** use CPFP to bump the fee of the penalty transaction	✓	✓
Spamming the tower has a cost	–	✓
Minimises number payment to the tower	✓	–
Exploiting requires paying for a subscription	–	✓

USER AUTHENTICATION

- Authenticating the user helps preventing resource abuse
- It is required for the subscription model
- **Message signing using the node’s secret key**
- **Message signing using an ephemeral key** (not linked node id)
- **Authentication via LSAT or similar approaches**

EXTENSIONS

The BOLT should have room for extensions so additional features can be added:

- Accountability
- Backups / arbitrary data storage
- Extend trigger logic to work with other protocols

CURRENT STATE OF THE CODE

- Standalone FOSS tower
- Plugin for c-lightning
- Subscriptions fee of charge
    - Paid subscriptions **Soon™**
- Communication via REST API
- Can be used for backups out of the box (even though it’s not meant for it at the moment)
- Live testing instances both for mainnet and testnet

RESOURCES (1/2)

The Eye of Satoshi

https://github.com/talaia-labs/python-teos

BOLT13

https://github.com/sr-gi/bolt13/blob/master/13-watchtowers.md

c-lightning plugin

https://github.com/talaia-labs/python-teos/tree/master/watchtower-plugin

RESOURCES (2/2)

Connecting to The Eye of Satoshi live instances

lightning-cli registertower towerid@host:port

mainnet

02f695cd372bcd949ff29465e72296eb959468e013a9b080742fb60fff27edc5f2@https://teos.pisa.watch:443

testnet

02f695cd372bcd949ff29465e72296eb959468e013a9b080742fb60fff27edc5f2@https://teos-testnet.pisa.watch:443

QUESTIONS

sr_gi

Talaia Labs