State of the Ark:
so do we need CTV?

Who am I?

- Steven Roose
- building Ark @ Second
- covenants.info

Ark?

- new layer-two protocol
- non-custodial
  - → users maintain unilateral exit

Ark?

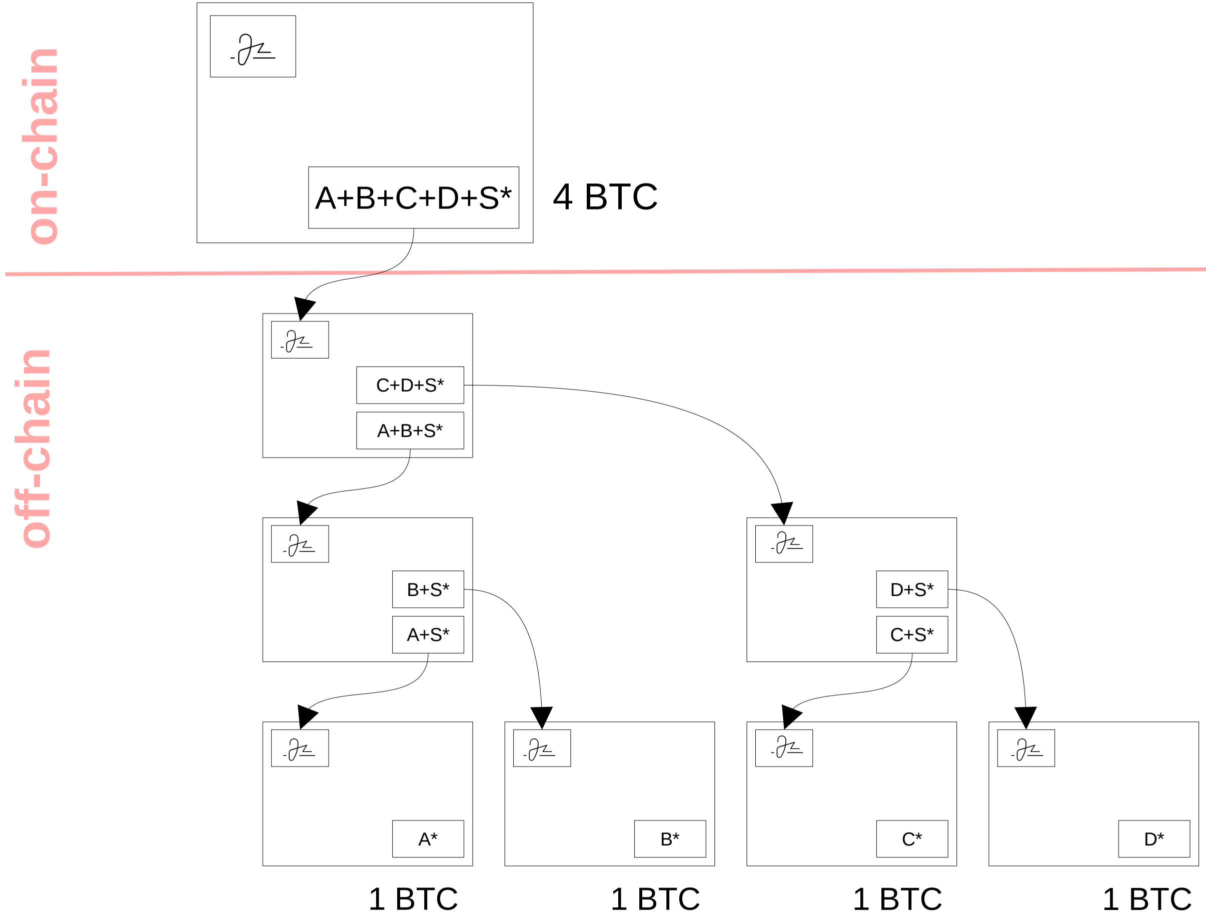
- new layer-two protocol
- non-custodial
- based on shared utxos
  - → virtual utxos or vtxos

Ark?

- new layer-two protocol
- non-custodial
- based on shared utxos
- good Lightning interoperability

Ark?

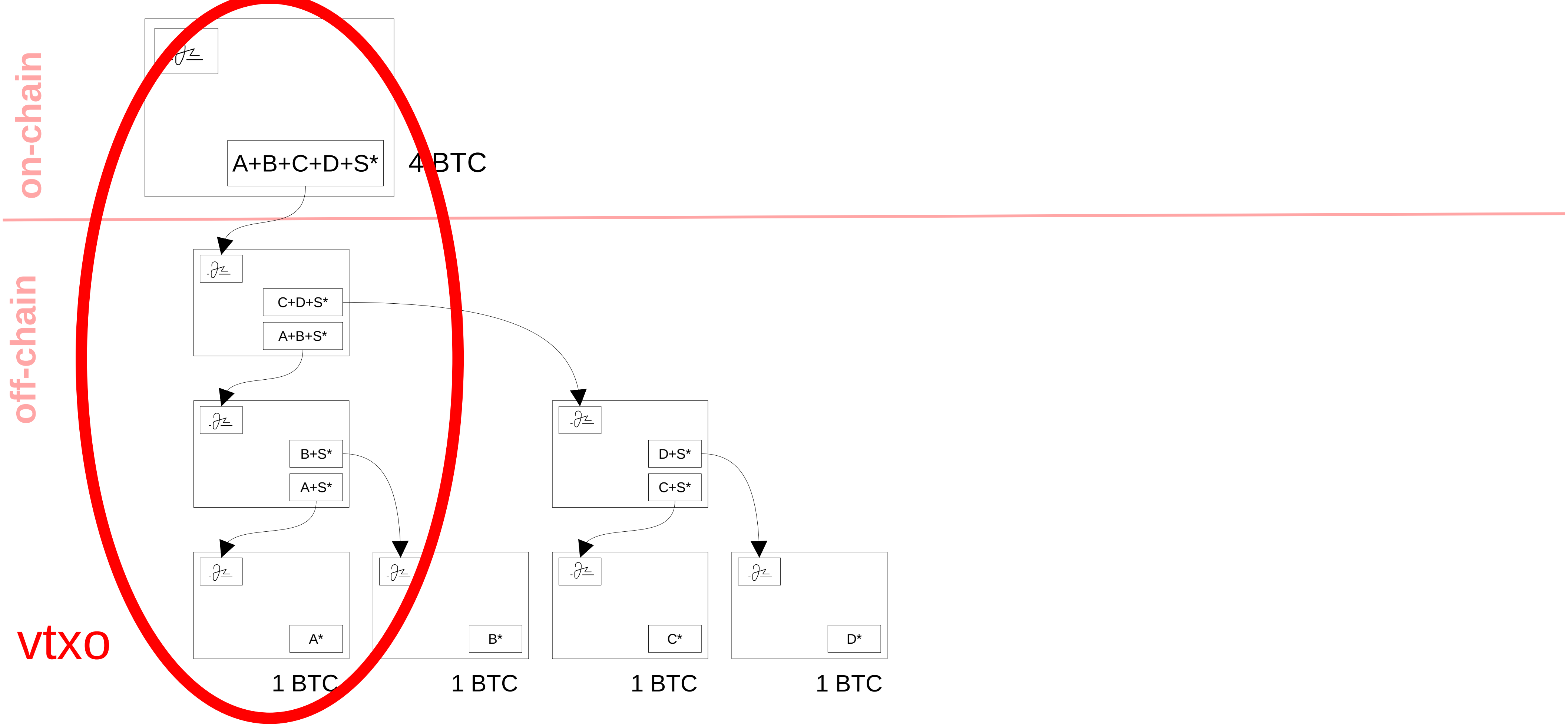
- new layer-two protocol
- non-custodial
- based on shared utxos
- good Lightning interoperability
- no channel or liquidity management
  - → client-server: server manages liquidity



S = Ark server pubkey

A* = A+S OR (A after Δt)

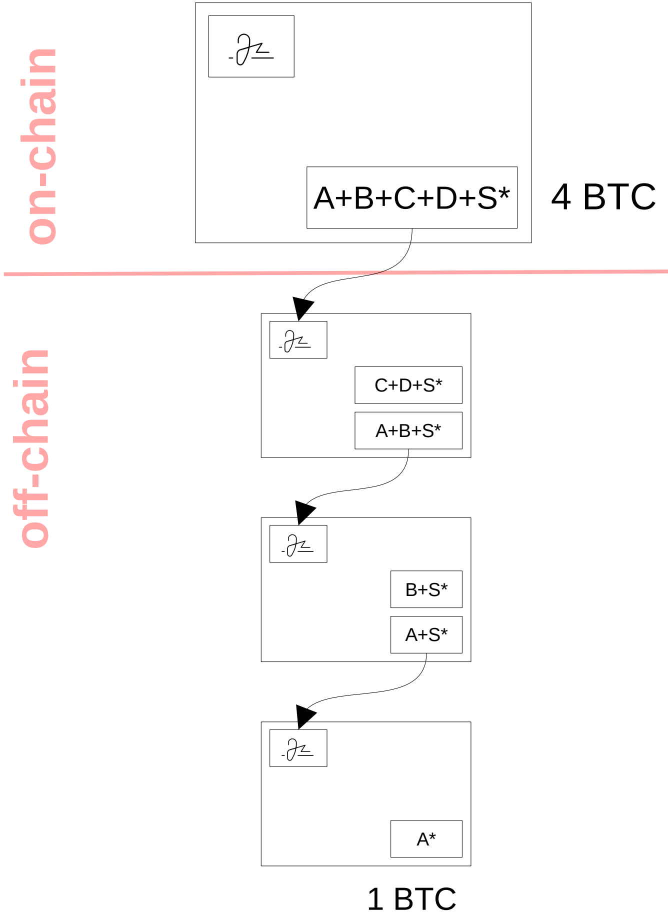
A+…+S* = A+...+S OR (S after T_{exp})



S = Ark server pubkey

A* = A+S OR (A after Δt)

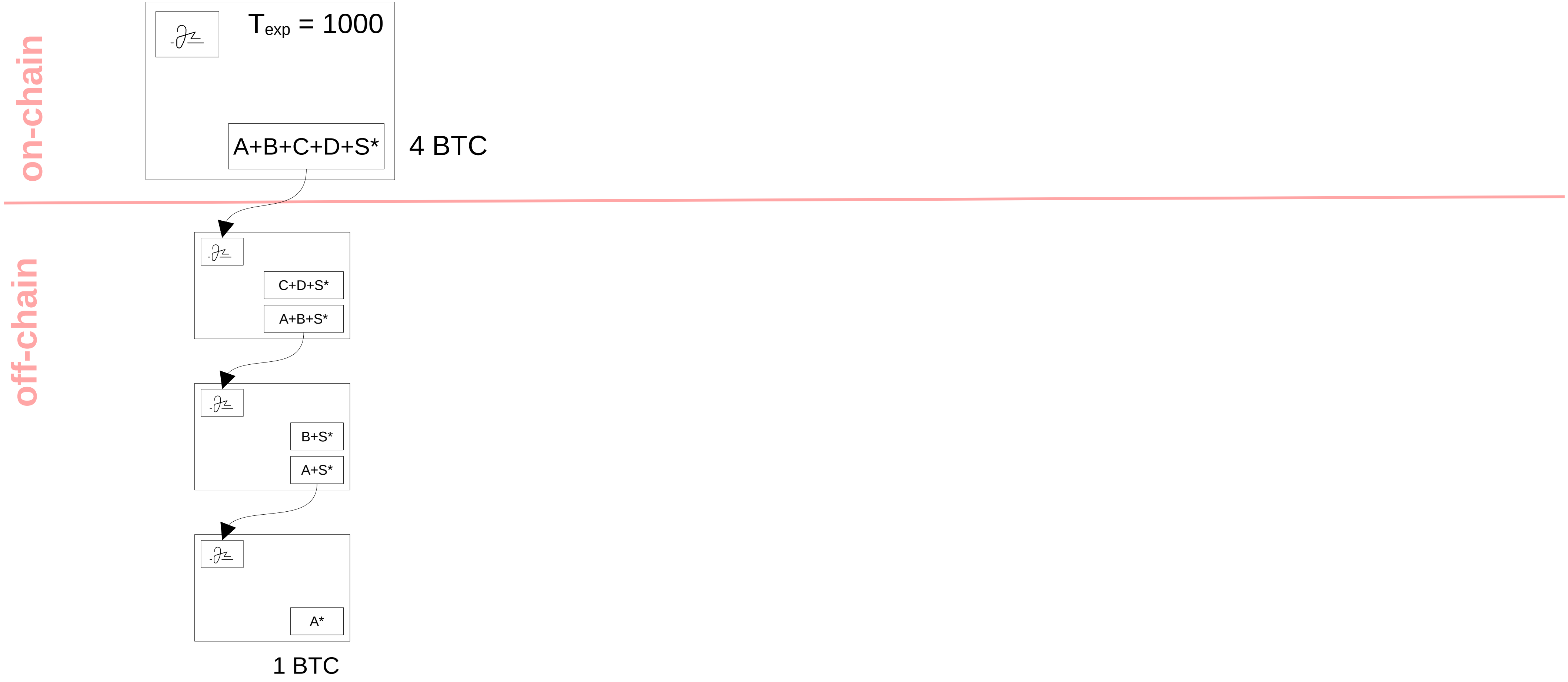
A+…+S* = A+...+S OR (S after T_{exp})



S = Ark server pubkey

A* = A+S OR (A after Δt)

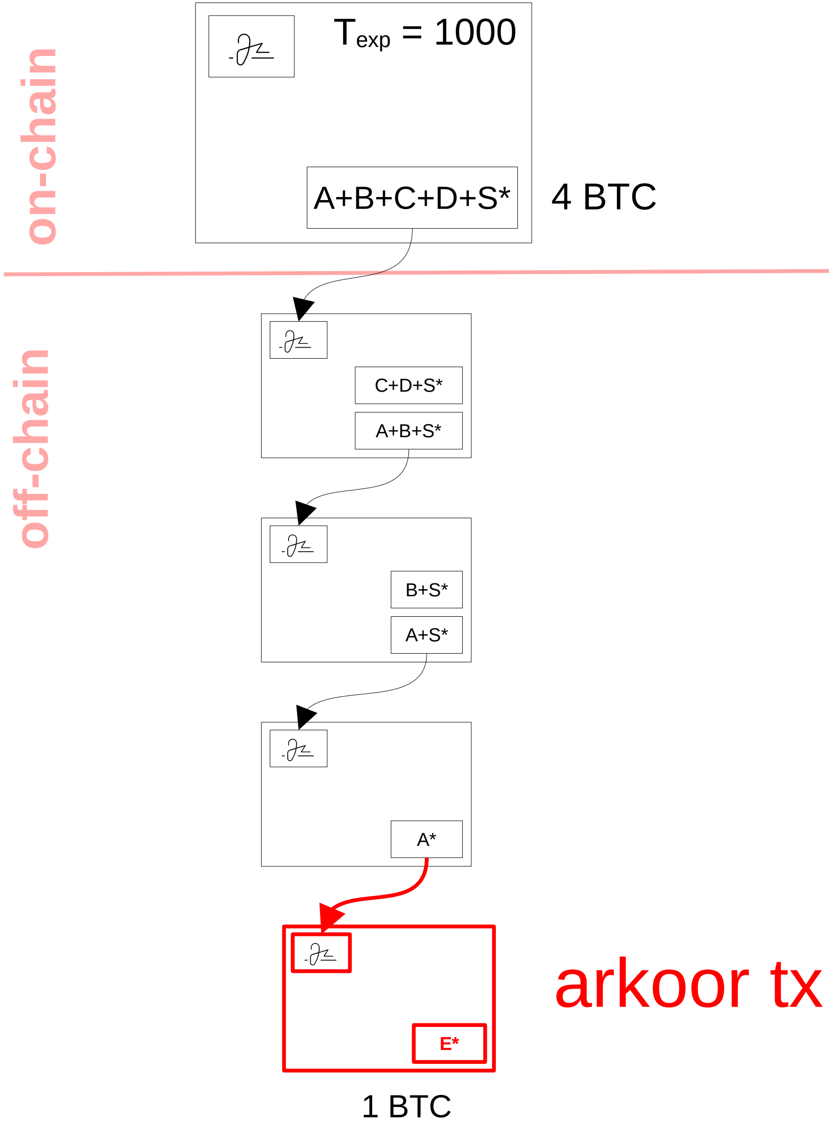
A+…+S* = A+...+S OR (S after T_{exp})



S = Ark server pubkey

A* = A+S OR (A after Δt)

A+...+S* = A+...+S OR (S after T_{exp})

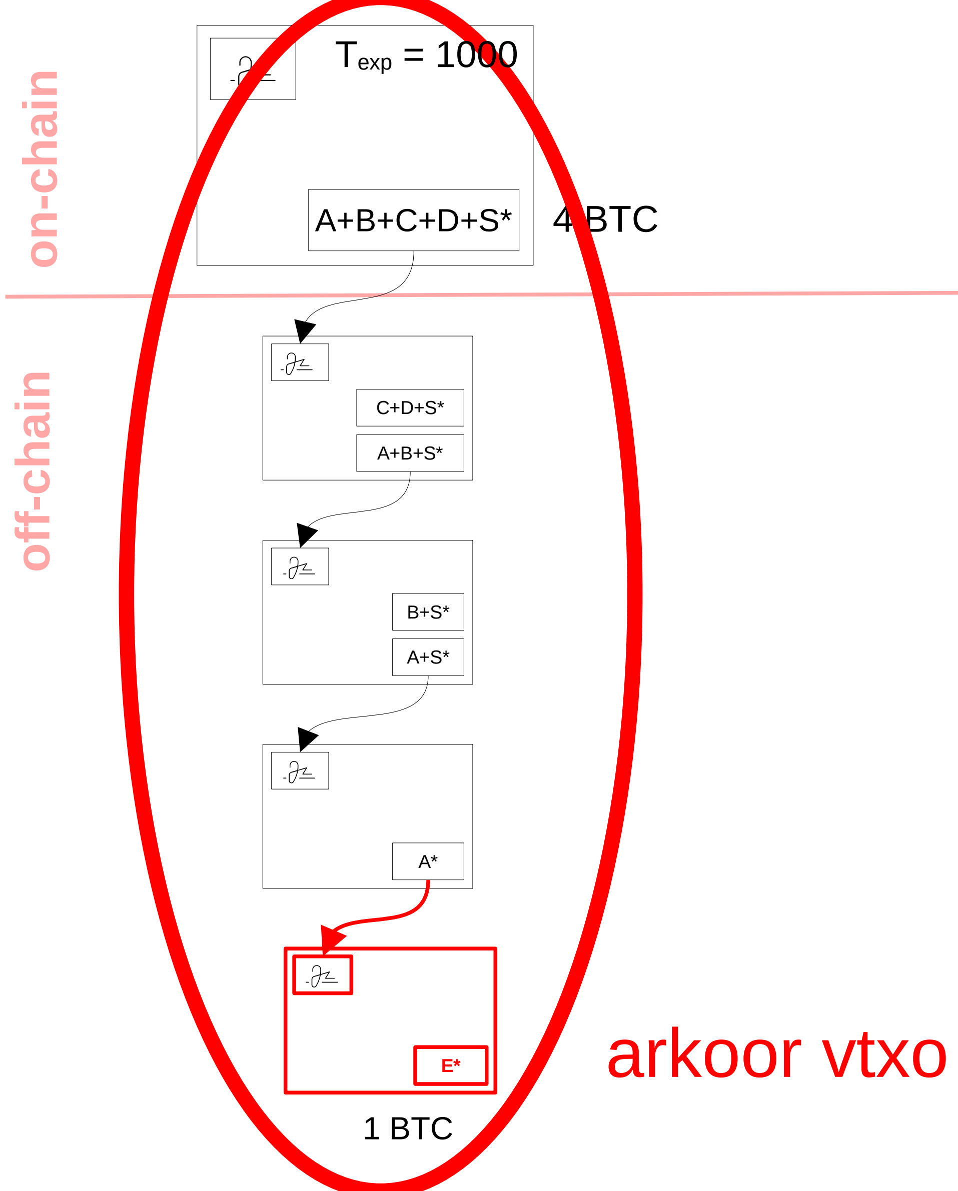


arkoor tx

S = Ark server pubkey

A* = A+S OR (A after Δt)

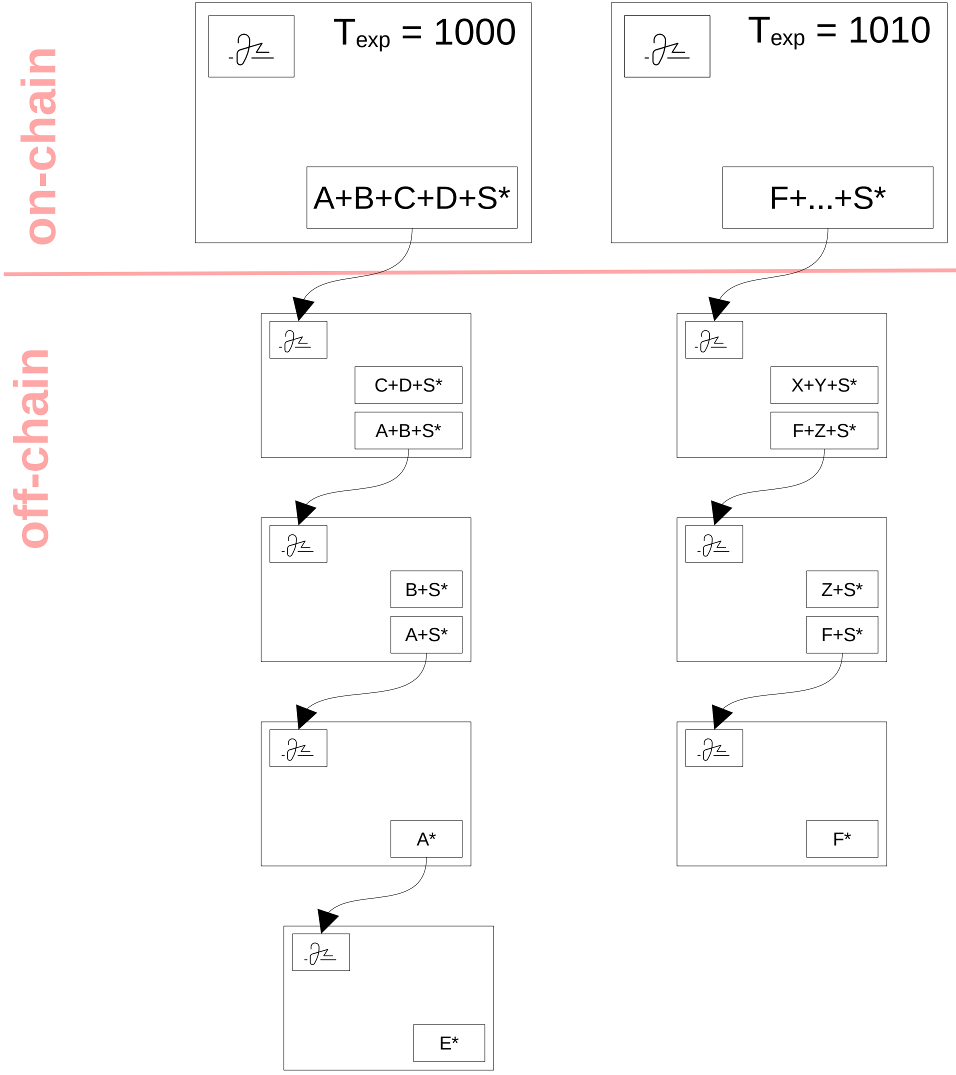
A+...+S* = A+...+S OR (S after T_{exp})



S = Ark server pubkey

$A^* = A+S$ OR $(A$ after $\Delta t)$

$A+\ldots+S^* = A+...+S$ OR $(S$ after $T_{exp})$



on-chain

off-chain

$T_{exp} = 1000$

A+B+C+D+S*

C+D+S*

A+B+S*

B+S*

A+S*

A*

E*

$T_{exp} = 1010$

F+...+S*

X+Y+S*

F+Z+S*

Z+S*

F+S*

F*

S = Ark server pubkey

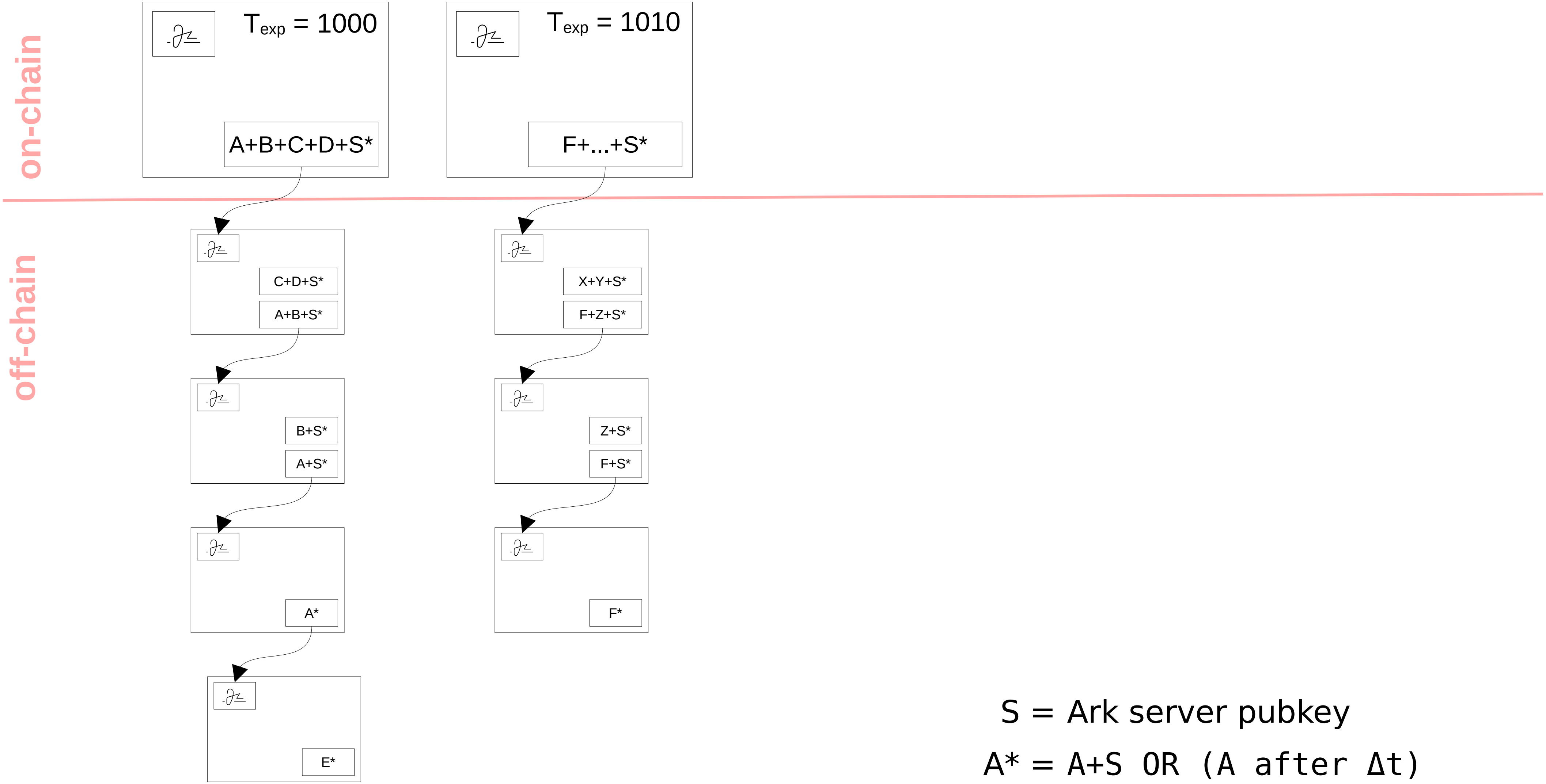
$A^* = A+S$ OR (A after Δt)

$A+\ldots+S^* = A+\ldots+S$ OR (S after T_{exp})

clArk round

1 input submission

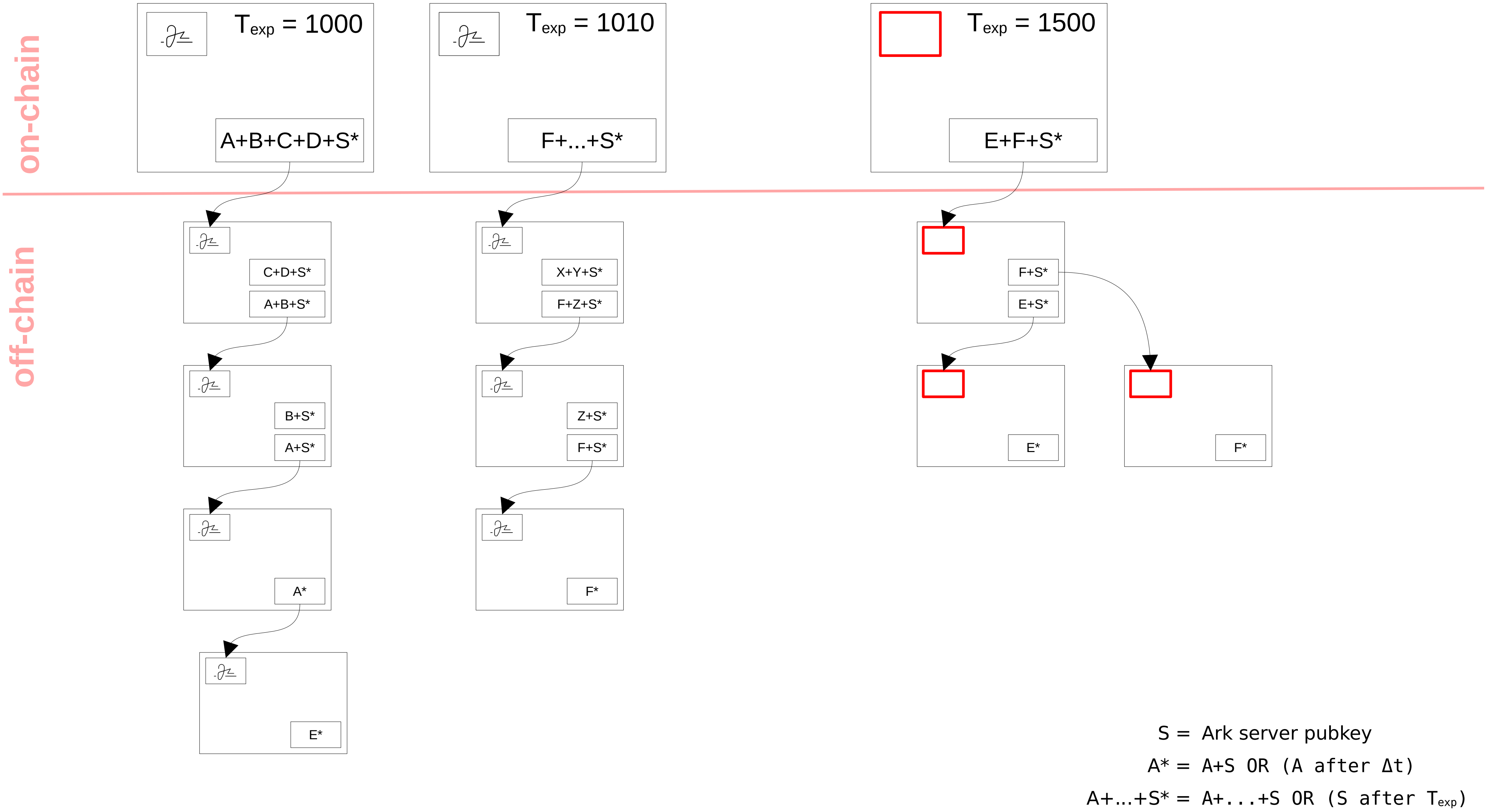
- → server creates unsigned vtxo tree



S = Ark server pubkey

A* = A+S OR (A after Δt)

A+...+S* = A+...+S OR (S after T_{exp})



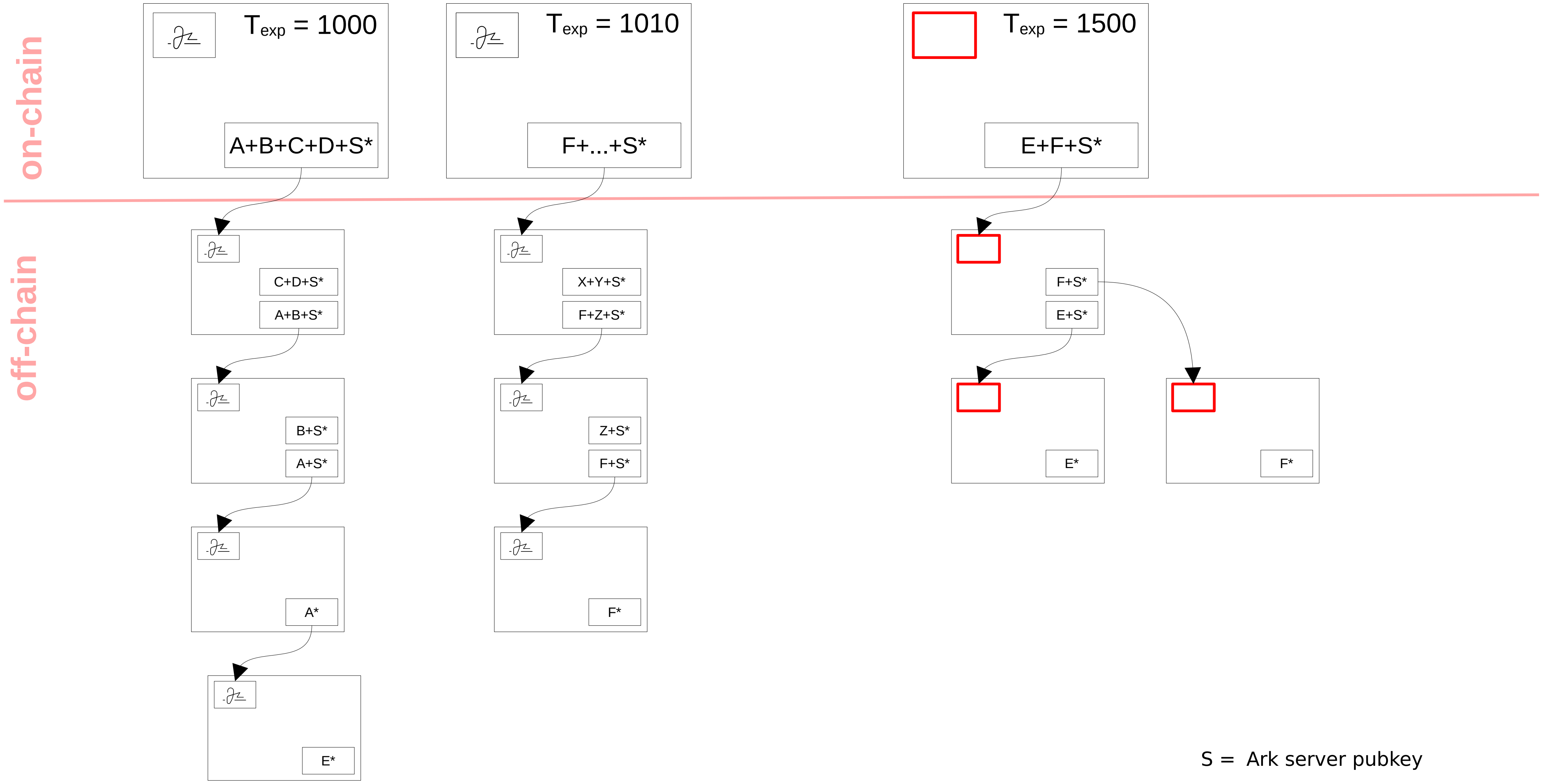
S = Ark server pubkey

A* = A+S OR (A after Δt)

A+...+S* = A+...+S OR (S after T_{exp})

clArk round

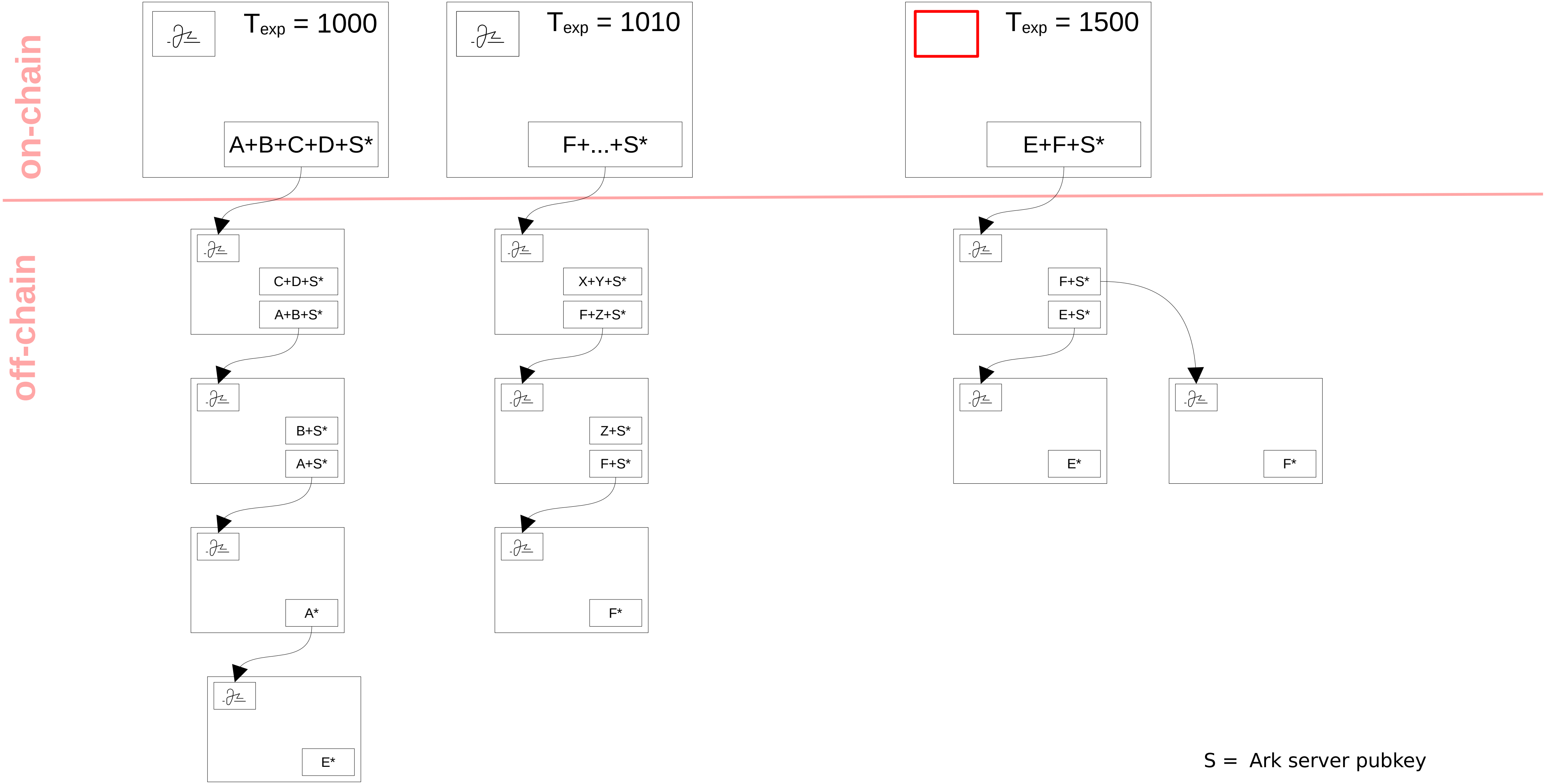
1. input submission
   - → server creates unsigned vtxo tree
2. users sign vtxo tree
   - → server creates signed vtxo tree



S = Ark server pubkey

A* = A+S OR (A after Δt)

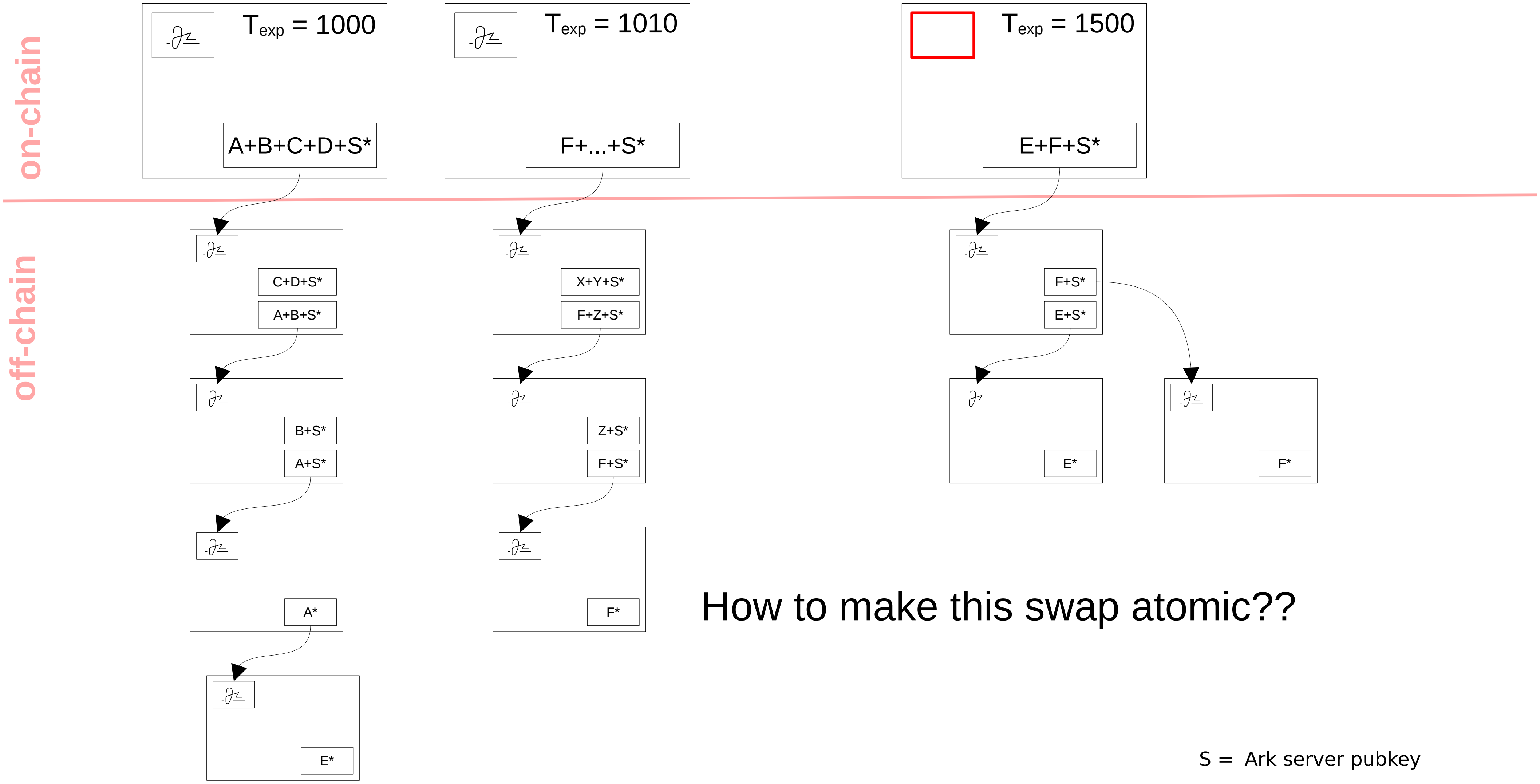
A+…+S* = A+...+S OR (S after T_{exp})



S = Ark server pubkey

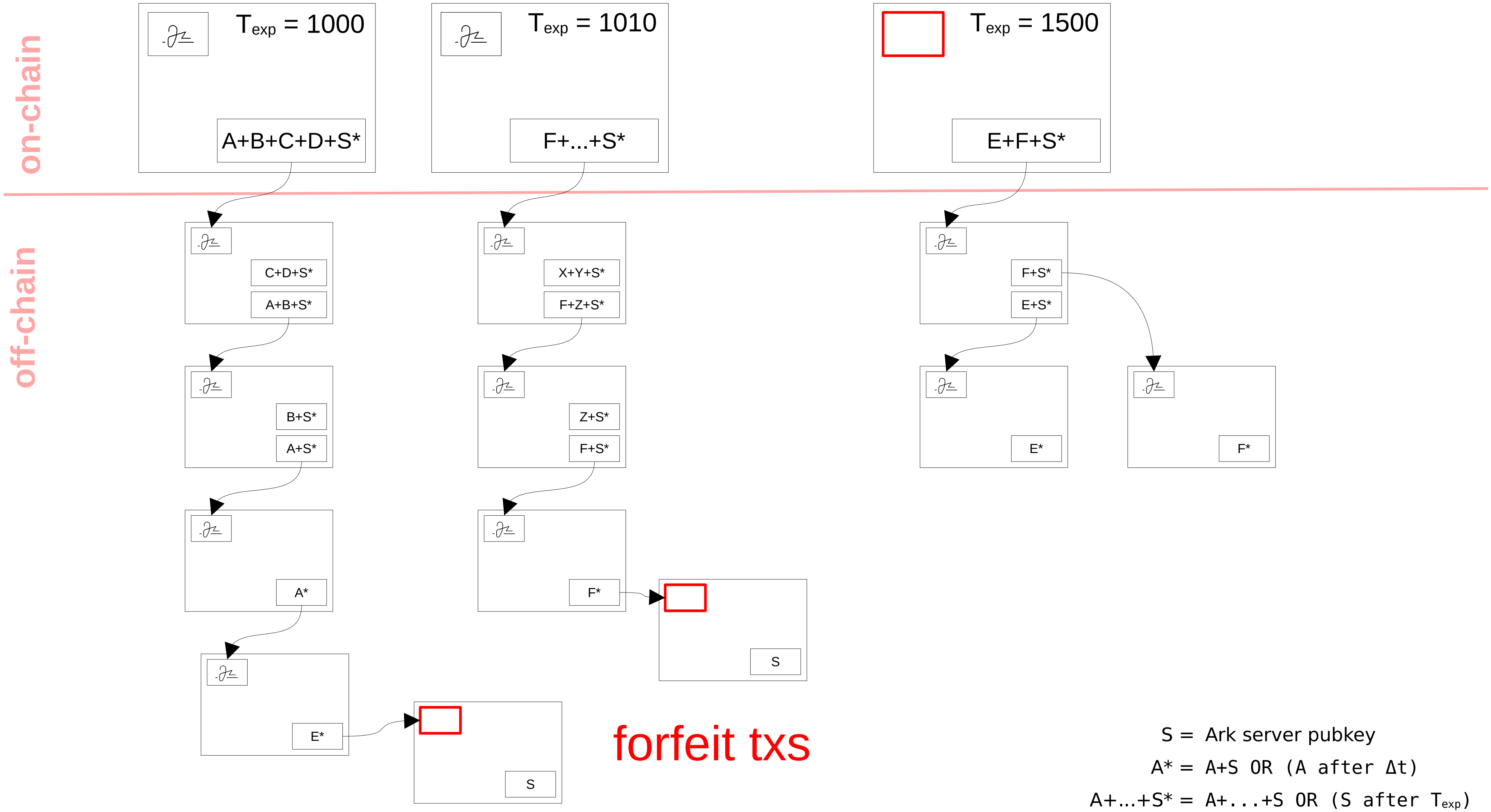
A* = A+S OR (A after Δt)

A+…+S* = A+...+S OR (S after T_{exp})



How to make this swap atomic??

S = Ark server pubkey
A* = A+S OR (A after Δt)
A+...+S* = A+...+S OR (S after T_{exp})

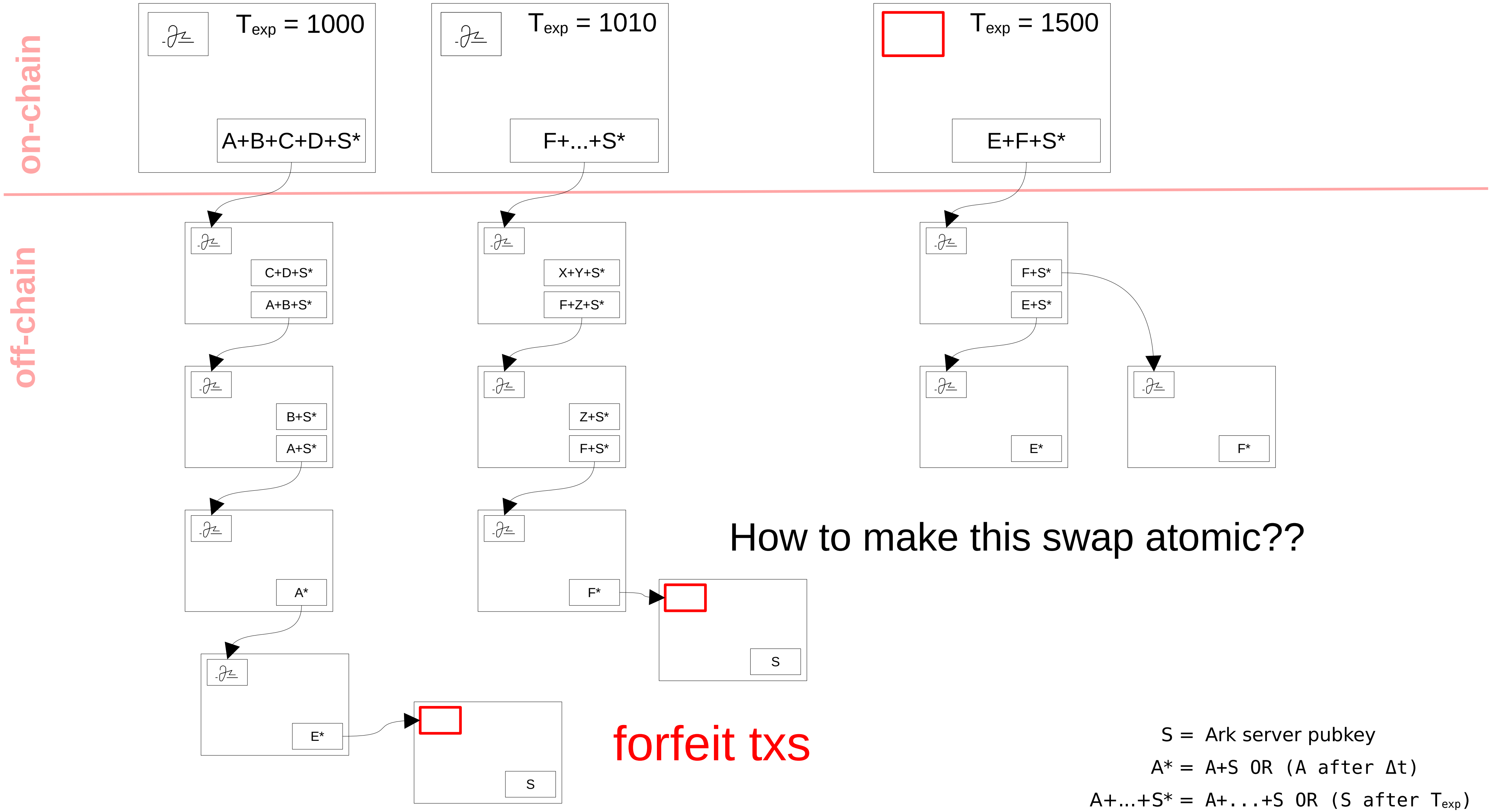


forfeit txs

S = Ark server pubkey

$A^* = A+S$ OR (A after Δt)

$A+\ldots+S^* = A+\ldots+S$ OR (S after T_{exp})



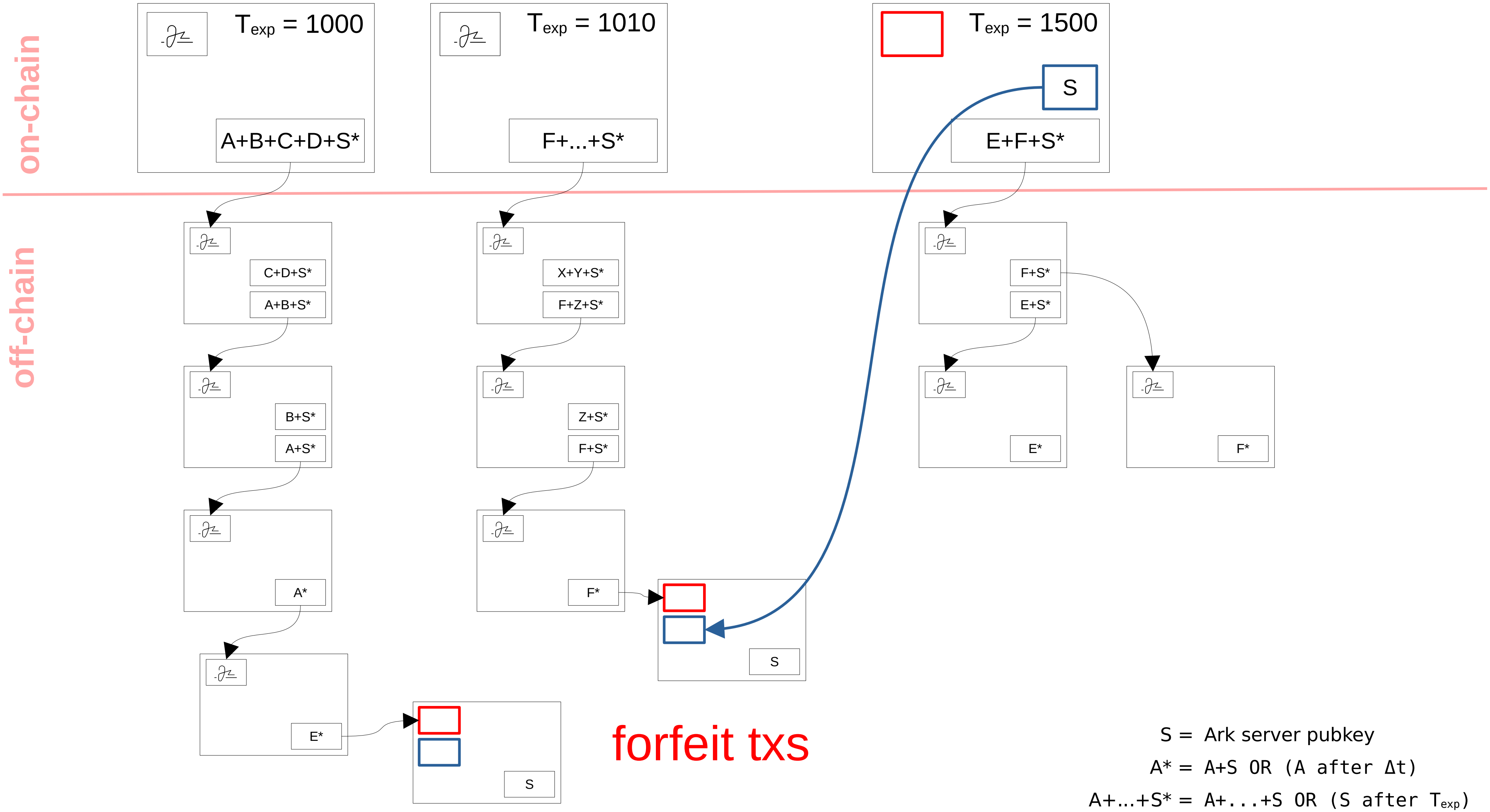
How to make this swap atomic??

forfeit txs

S = Ark server pubkey

$A^* = A+S$ OR (A after Δt)

$A+\ldots+S^* = A+\ldots+S$ OR (S after T_{exp})



on-chain

off-chain

T_{exp} = 1000

A+B+C+D+S*

T_{exp} = 1010

F+...+S*

T_{exp} = 1500

S

E+F+S*

C+D+S*

A+B+S*

B+S*

A+S*

A*

E*

S

X+Y+S*

F+Z+S*

Z+S*

F+S*

F*

S

F+S*

E+S*

E*

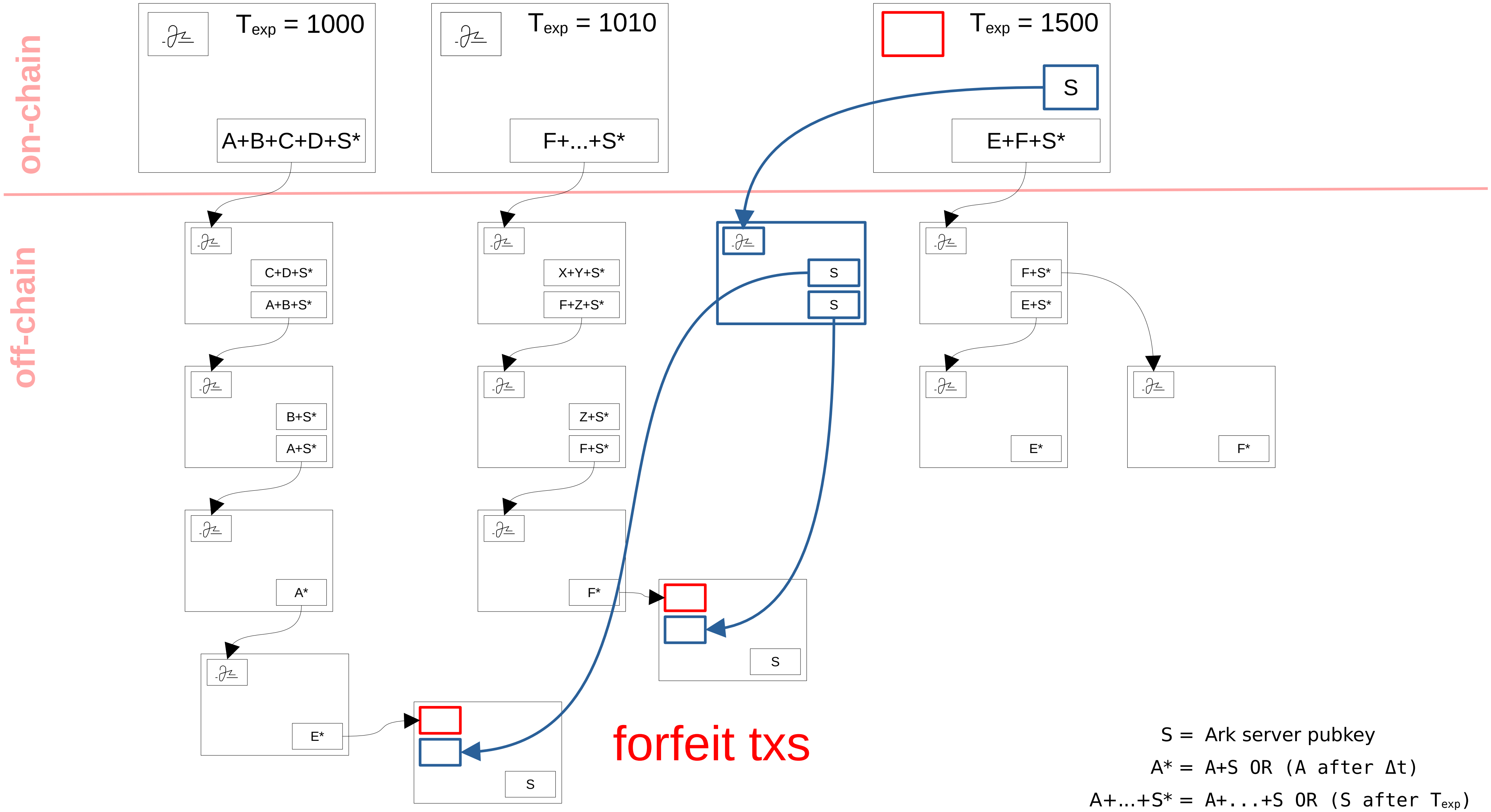
F*

forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

A+…+S* = A+...+S OR (S after T_{exp})



forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

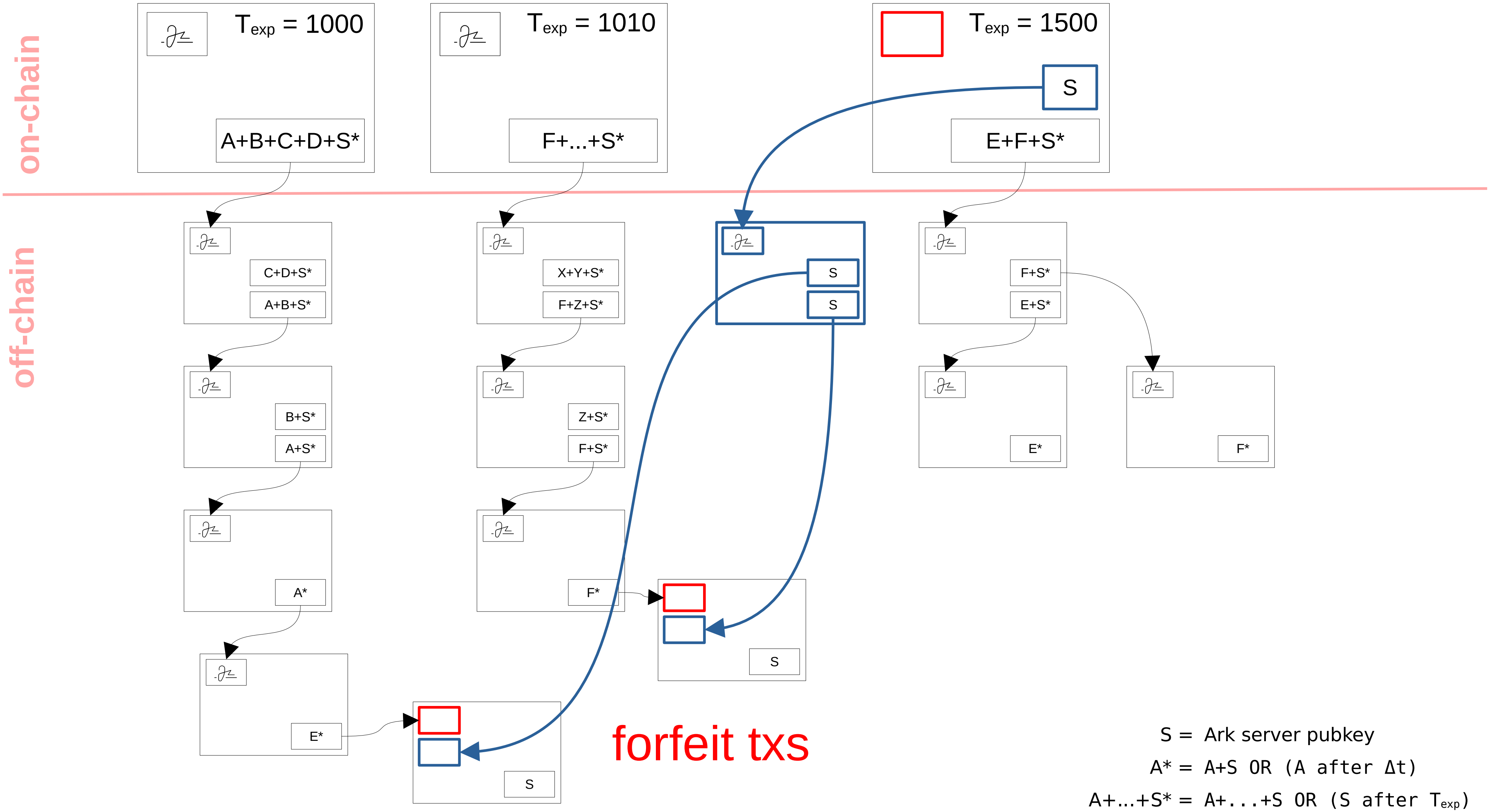
A+…+S* = A+...+S OR (S after T_{exp})

clArk round

1. input submission
   - → server creates unsigned vtxo tree
2. users sign vtxo tree
   - → server creates signed vtxo tree
   - → server creates connector txs

clArk round

1. input submission
   - → server creates unsigned vtxo tree
2. users sign vtxo tree
   - → server creates signed vtxo tree
   - → server creates connector txs
3. users sign forfeit txs

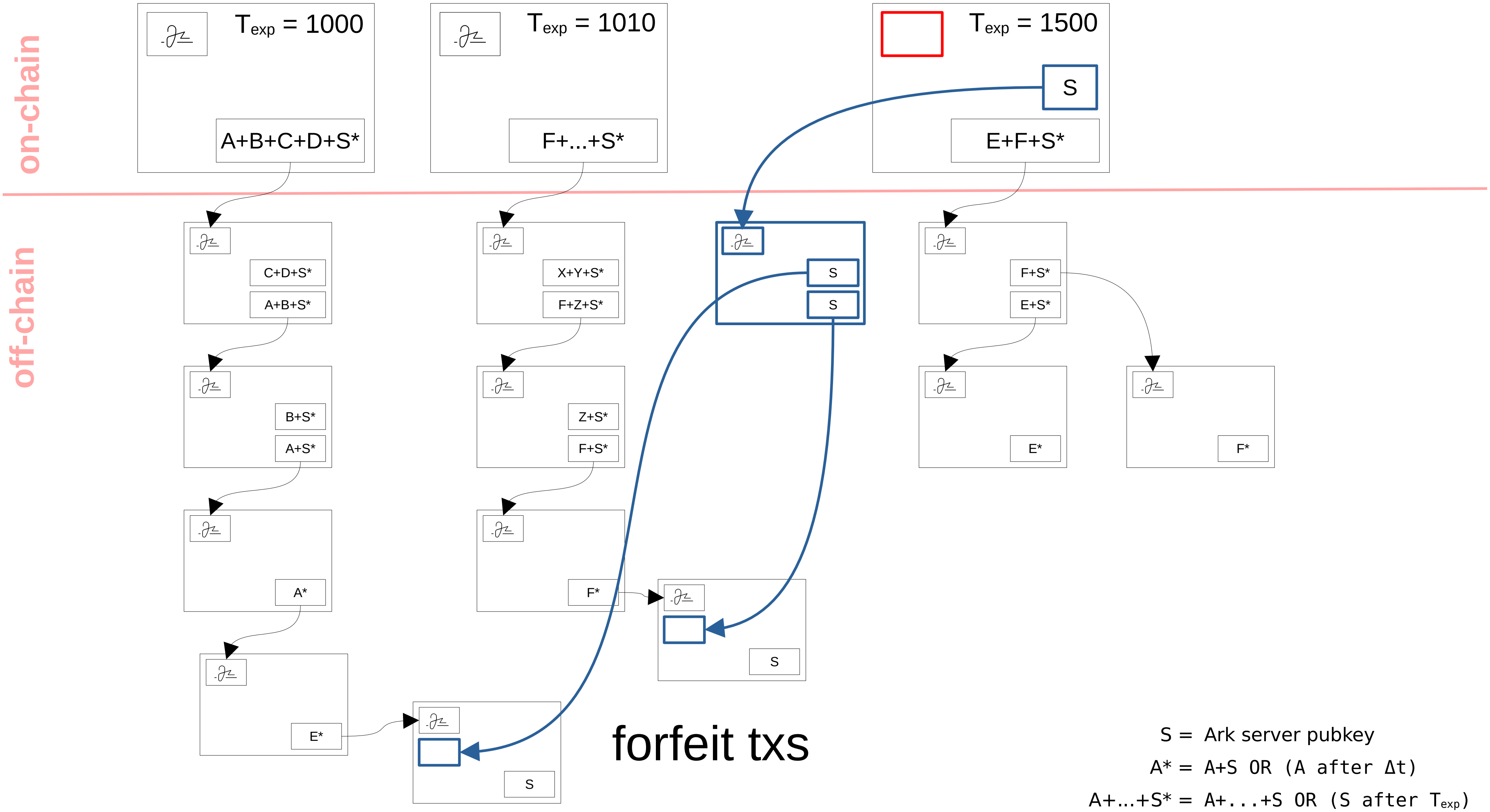


forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

A+...+S* = A+...+S OR (S after T_{exp})



forfeit txs

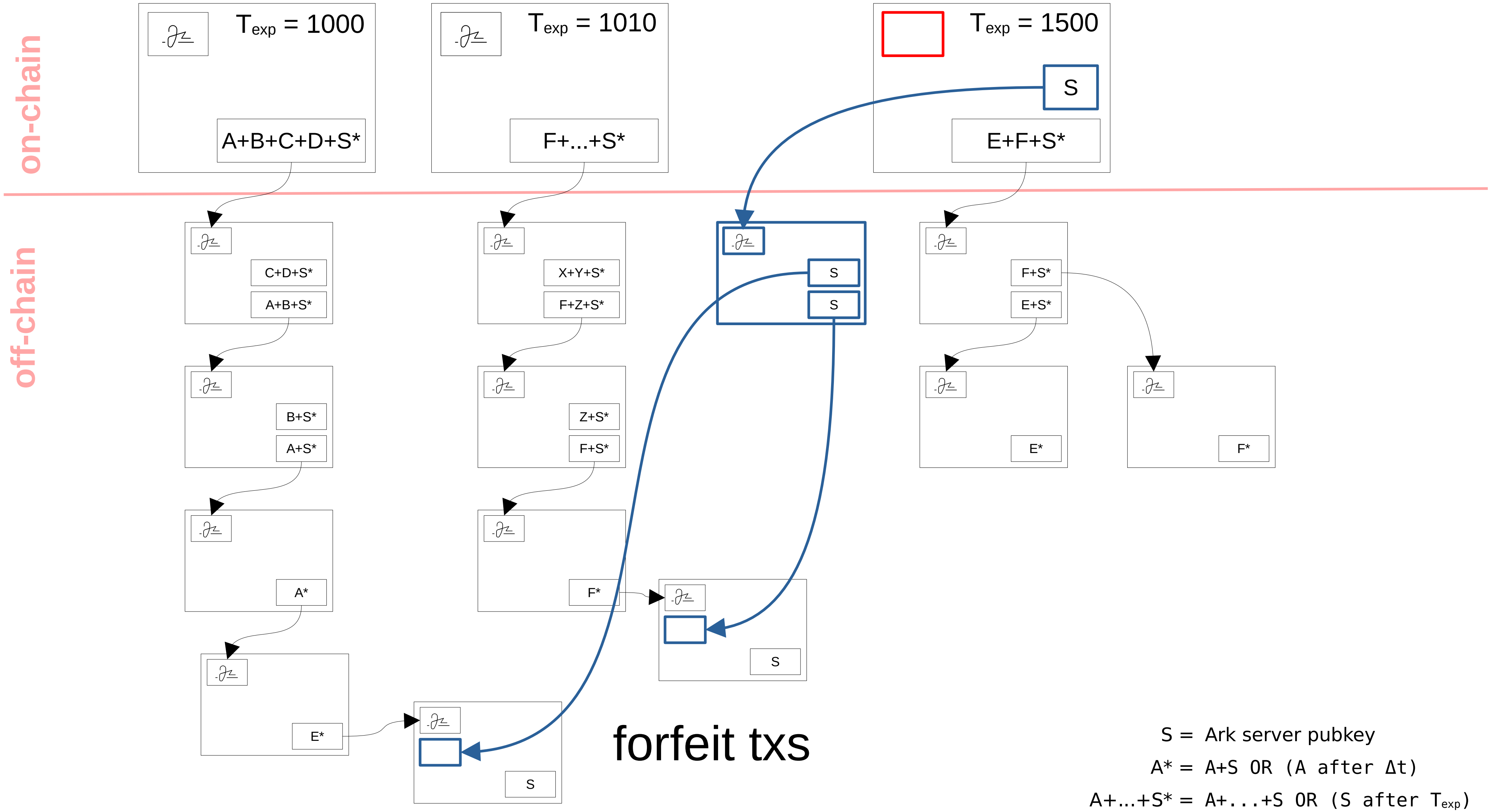
S = Ark server pubkey

A* = A+S OR (A after Δt)

A+…+S* = A+...+S OR (S after T_{exp})

clArk round

1. input submission
   - → server creates unsigned vtxo tree
2. users sign vtxo tree
   - → server creates signed vtxo tree
   - → server creates connector txs
3. users sign forfeit txs
   - → server signs round tx and broadcast

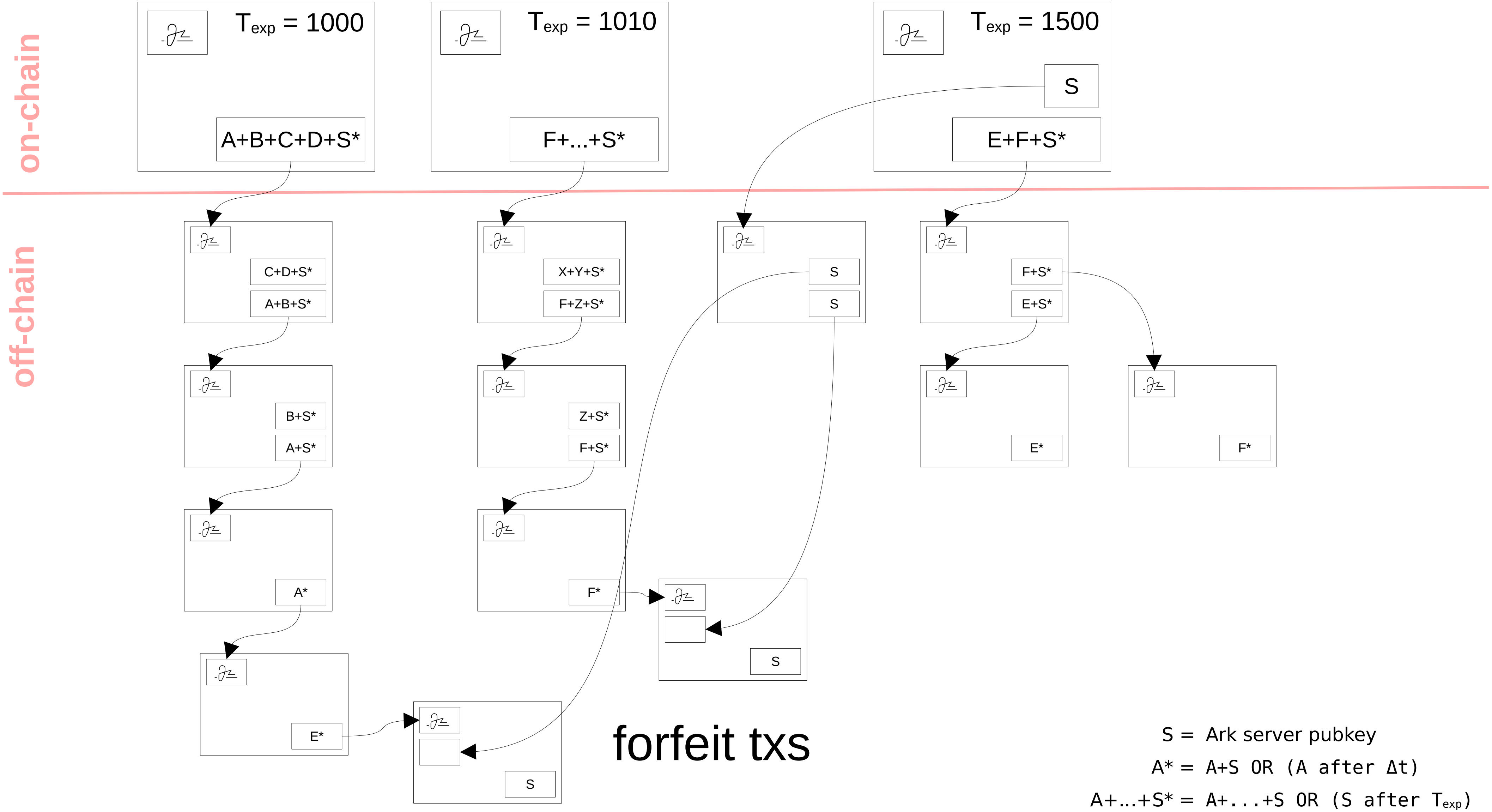


forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

A+…+S* = A+...+S OR (S after T_{exp})



forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

A+…+S* = A+...+S OR (S after T_{exp})

clArk round

1 input submission

- → server creates unsigned vtxo tree

2 users sign vtxo tree

- → server creates signed vtxo tree
- → server creates connector txs

3 users sign forfeit txs

- → server signs round tx and broadcast

clArk round

1. input submission
2. users sign vtxo tree
3. users sign forfeit txs

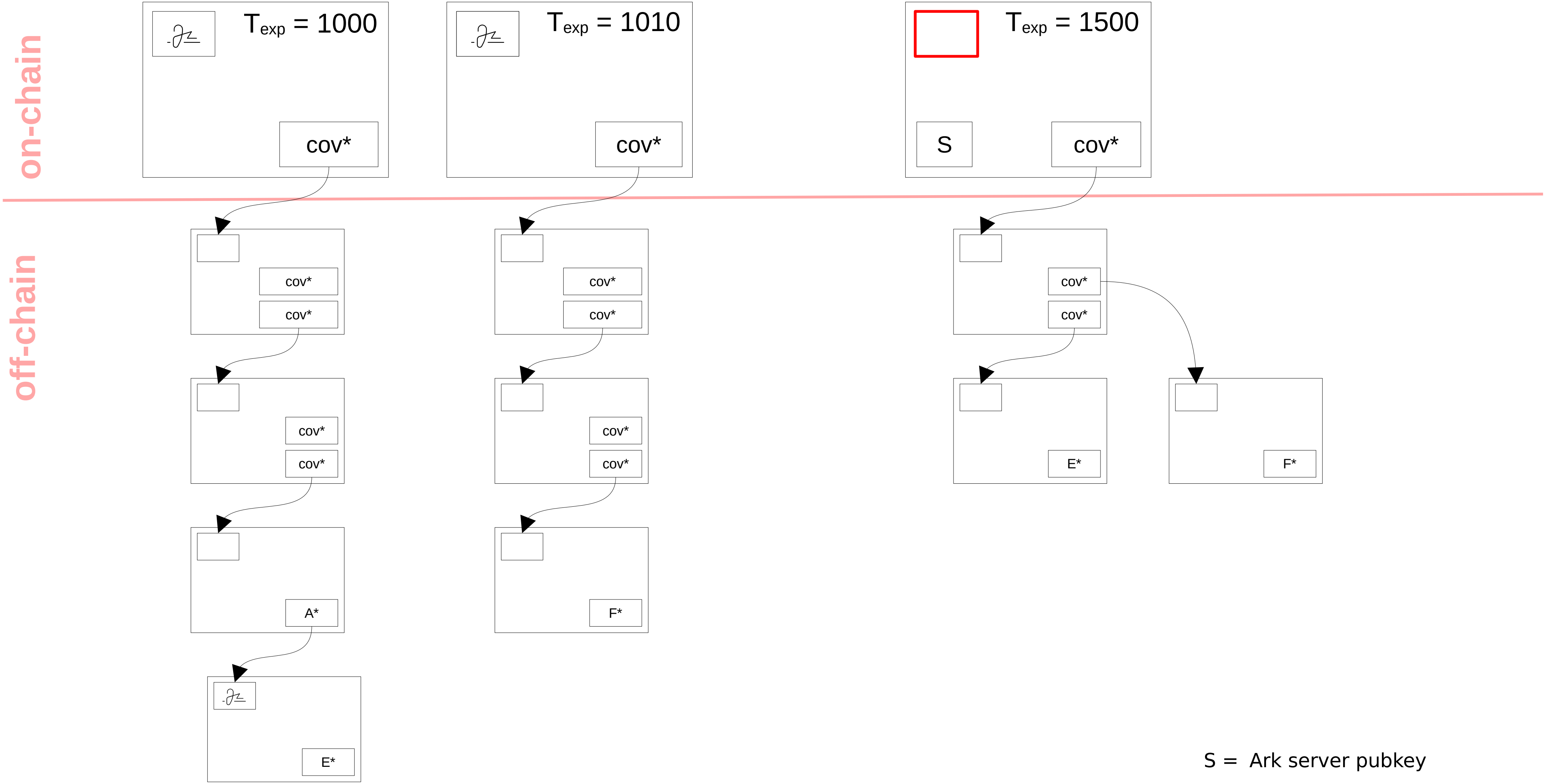
→ denial-of-service risk

Ark with CTV

Ark round

1 input submission

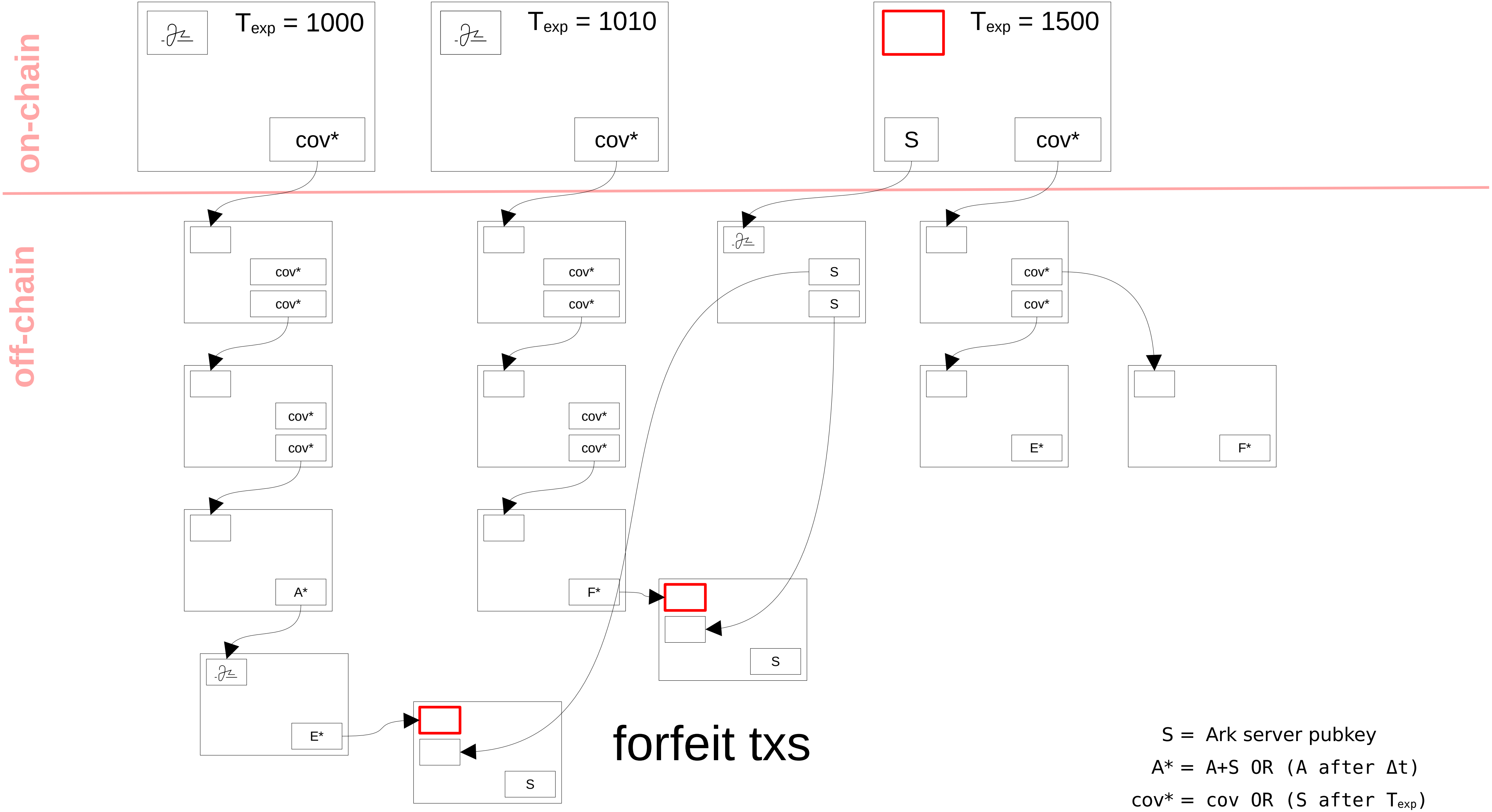
- → server creates vtxo tree
- → server creates connector txs



S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})



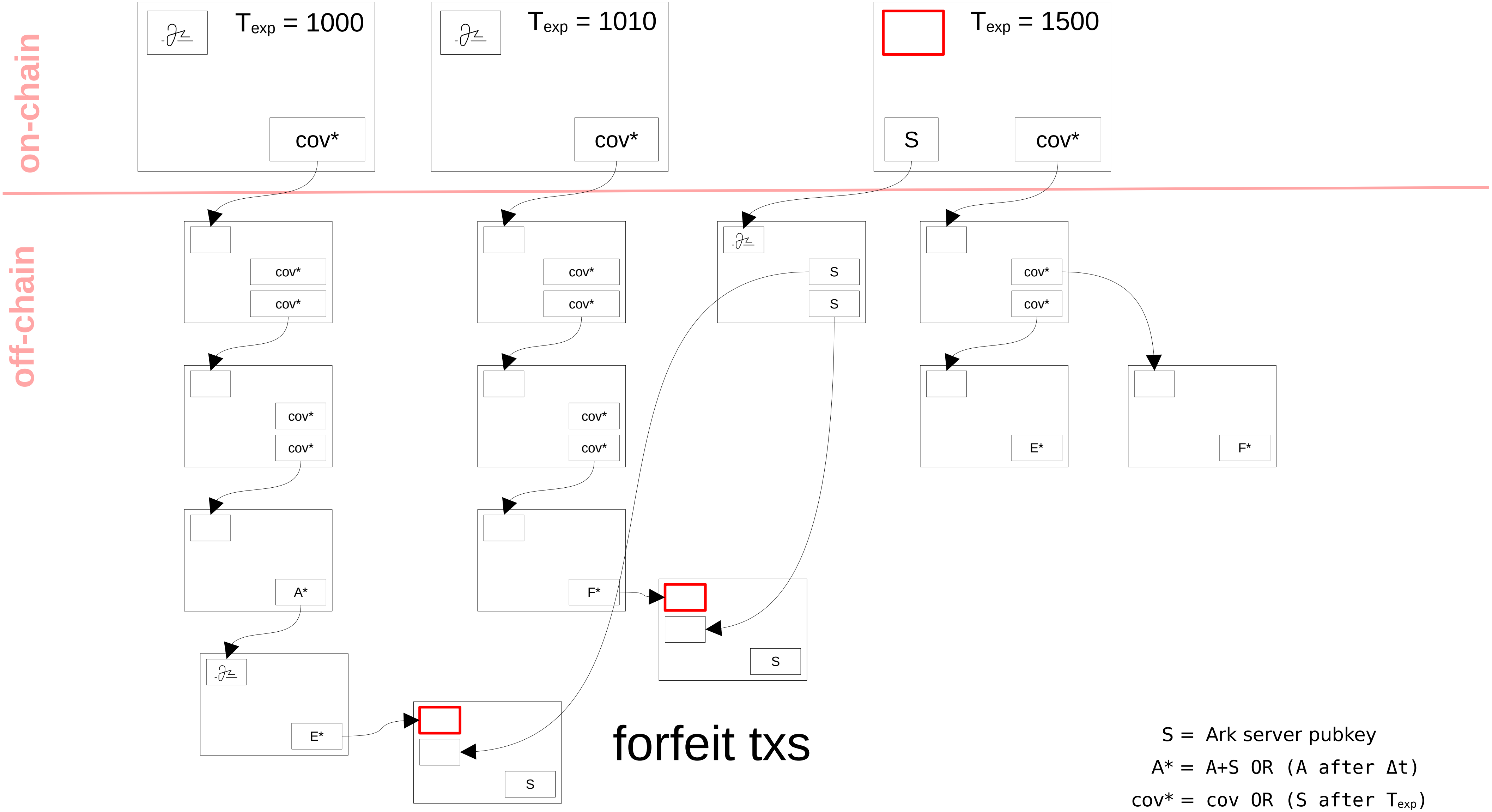
S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})

Ark round

1. input submission
   - → server creates vtxo tree
   - → server creates connector txs
2. users sign forfeit txs
   - → server signs round tx and broadcast

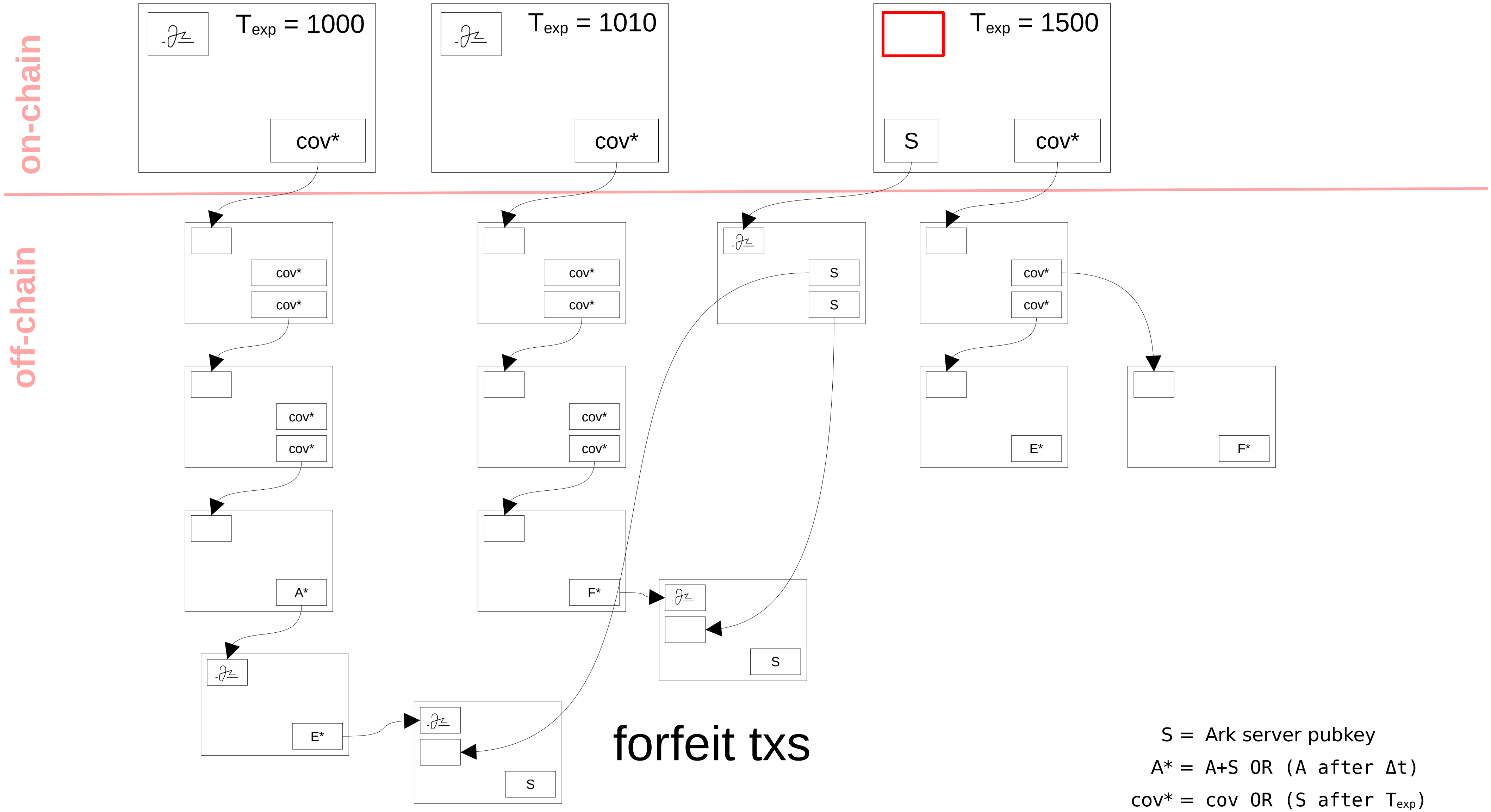


forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})

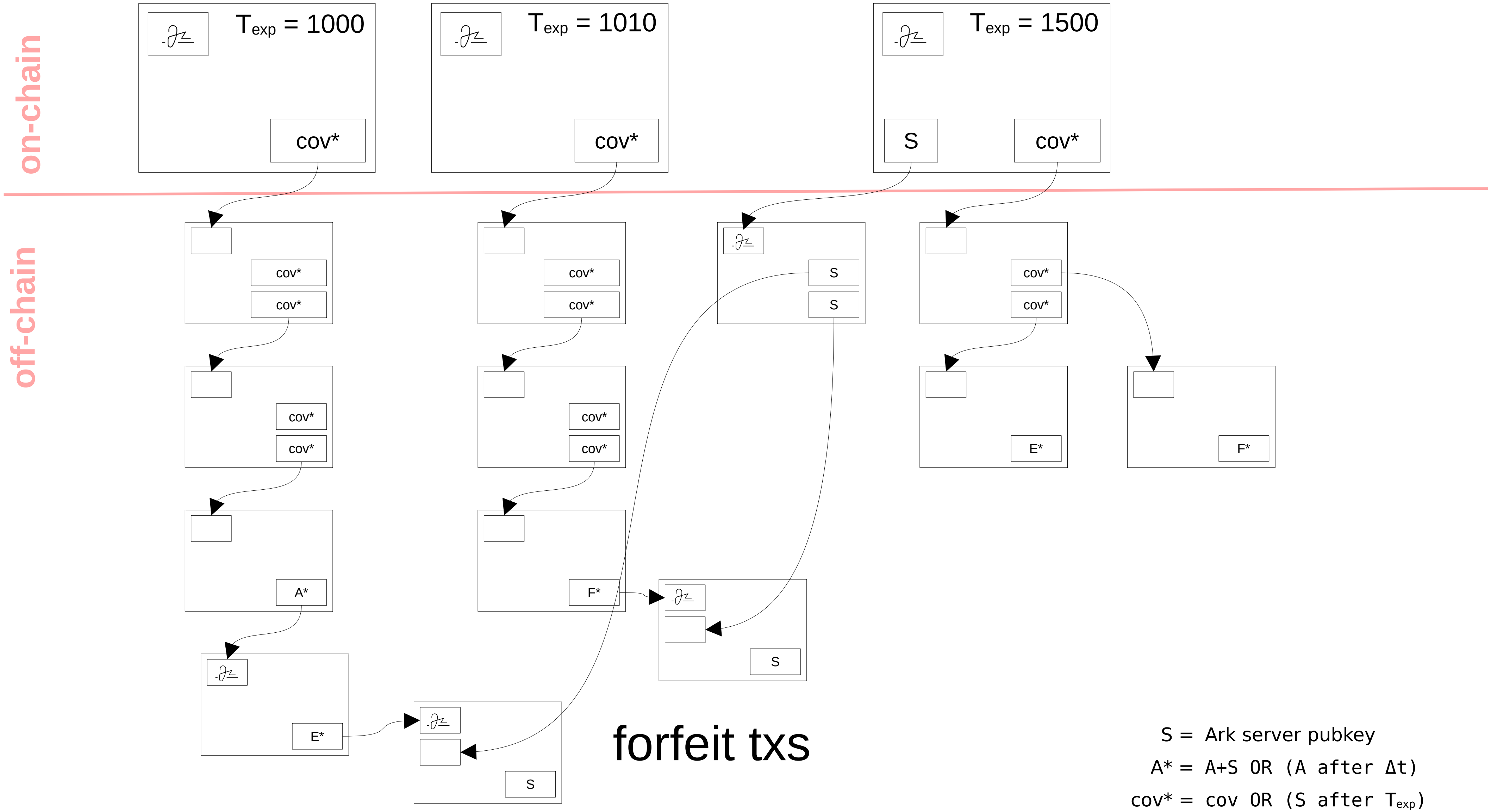


forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})



forfeit txs

S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})

Ark round

1. input submission
2. users sign forfeit txs

→ denial-of-service risk

Erk* round

1 input submission

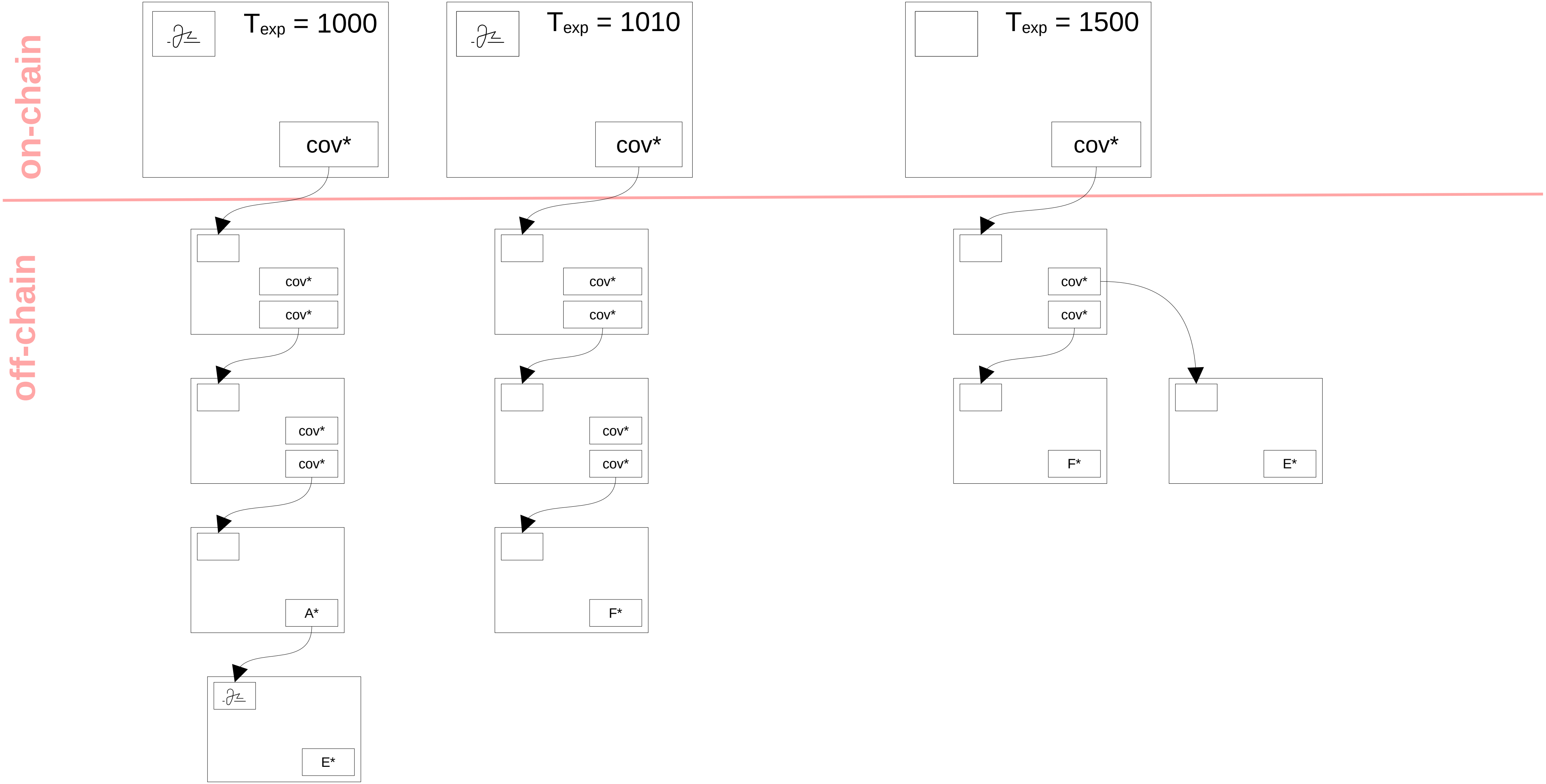
→ server creates vtxo tree

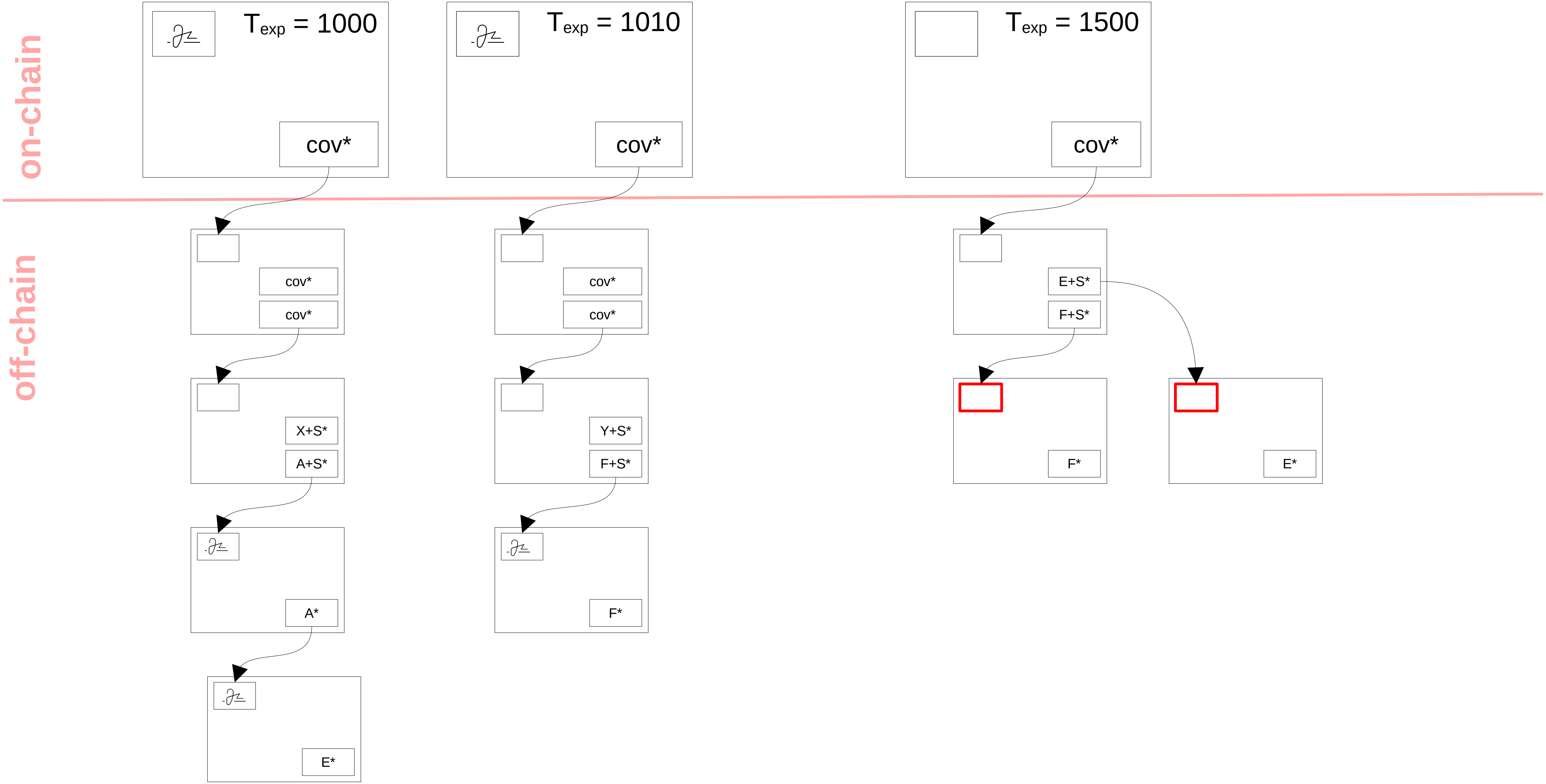
→ server creates ***refund*** txs

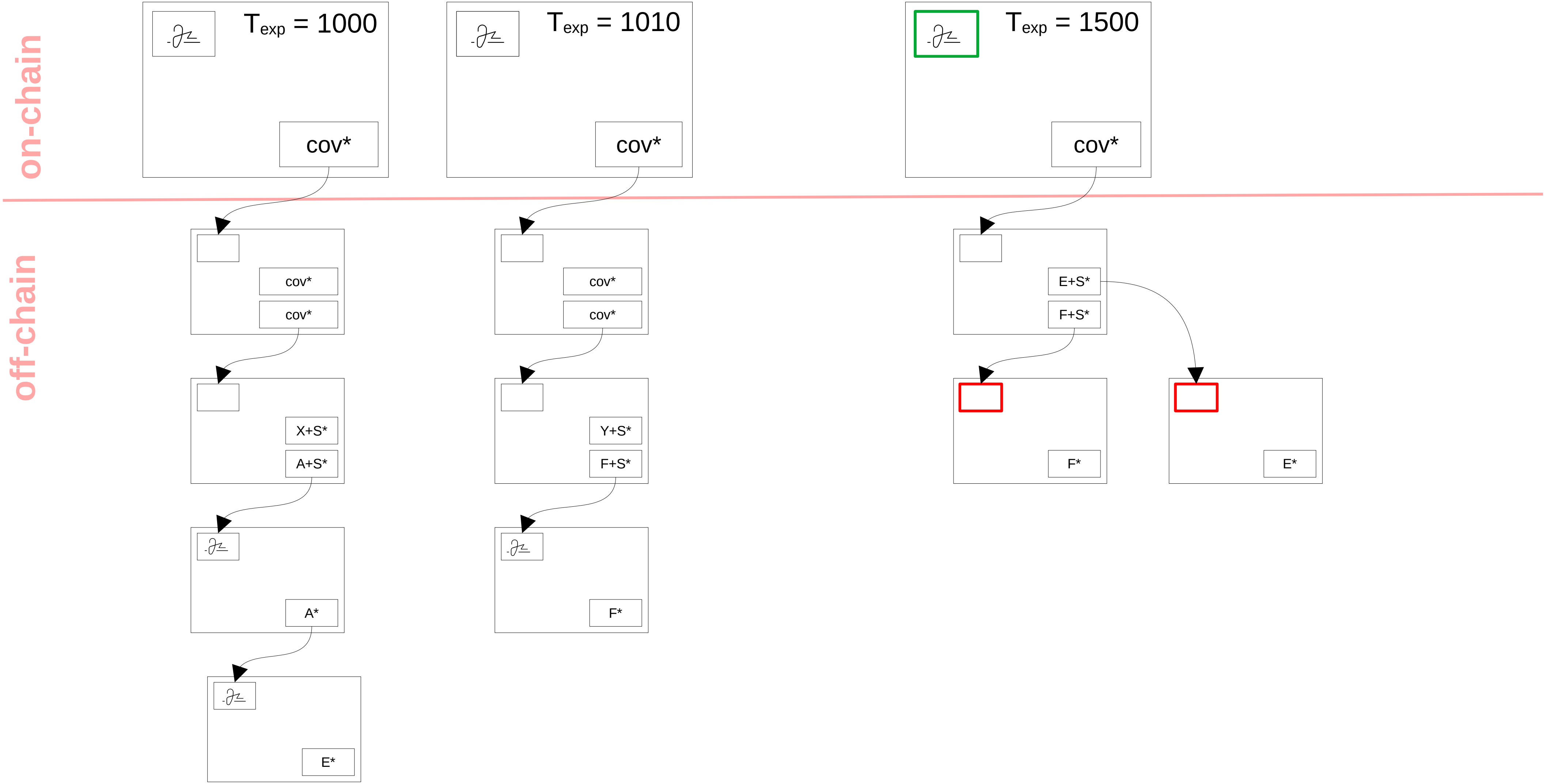
→ server signs round tx and broadcast

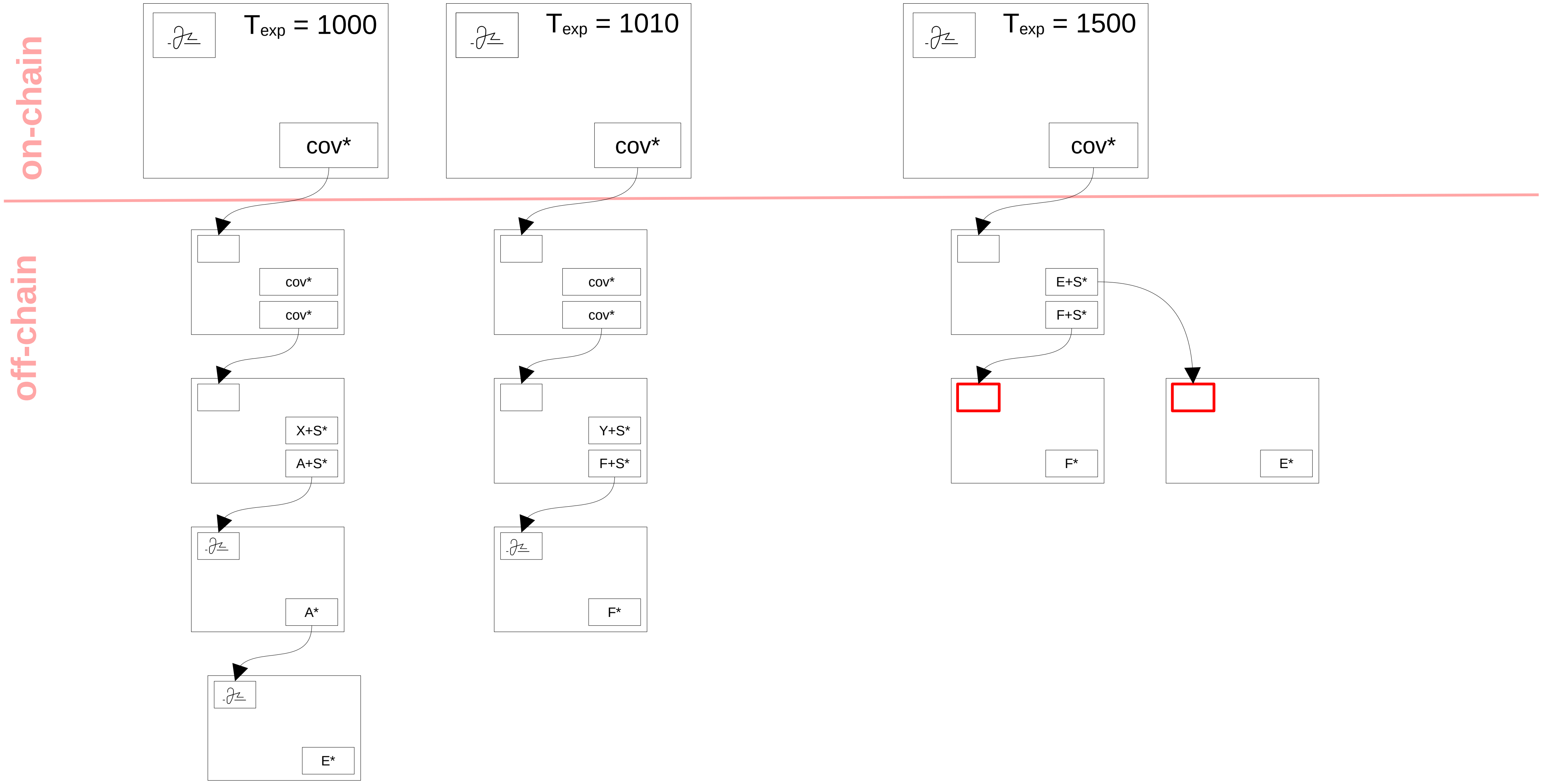
→ users sign refund txs any time later

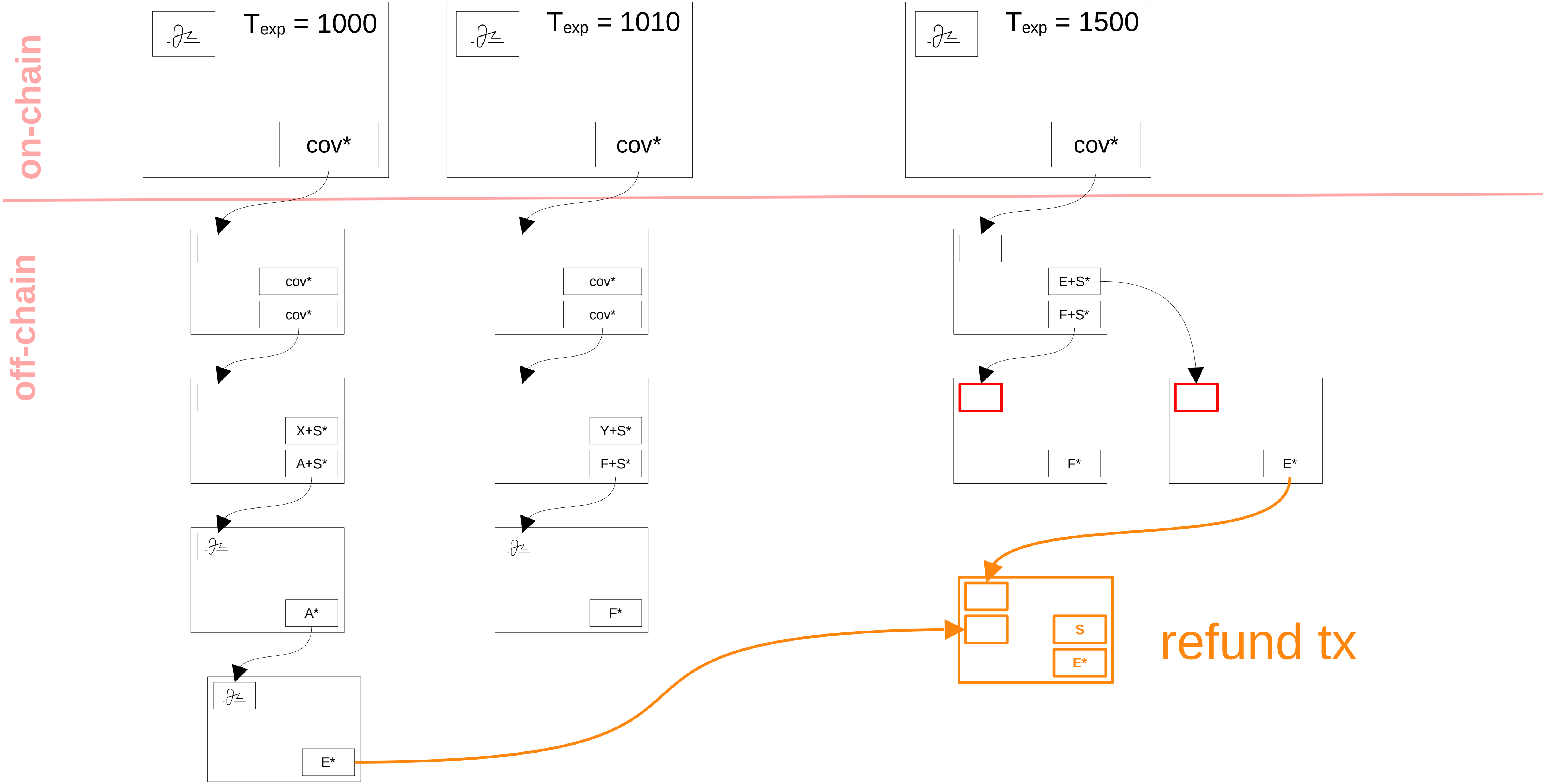
→ server gives user vtxo signature

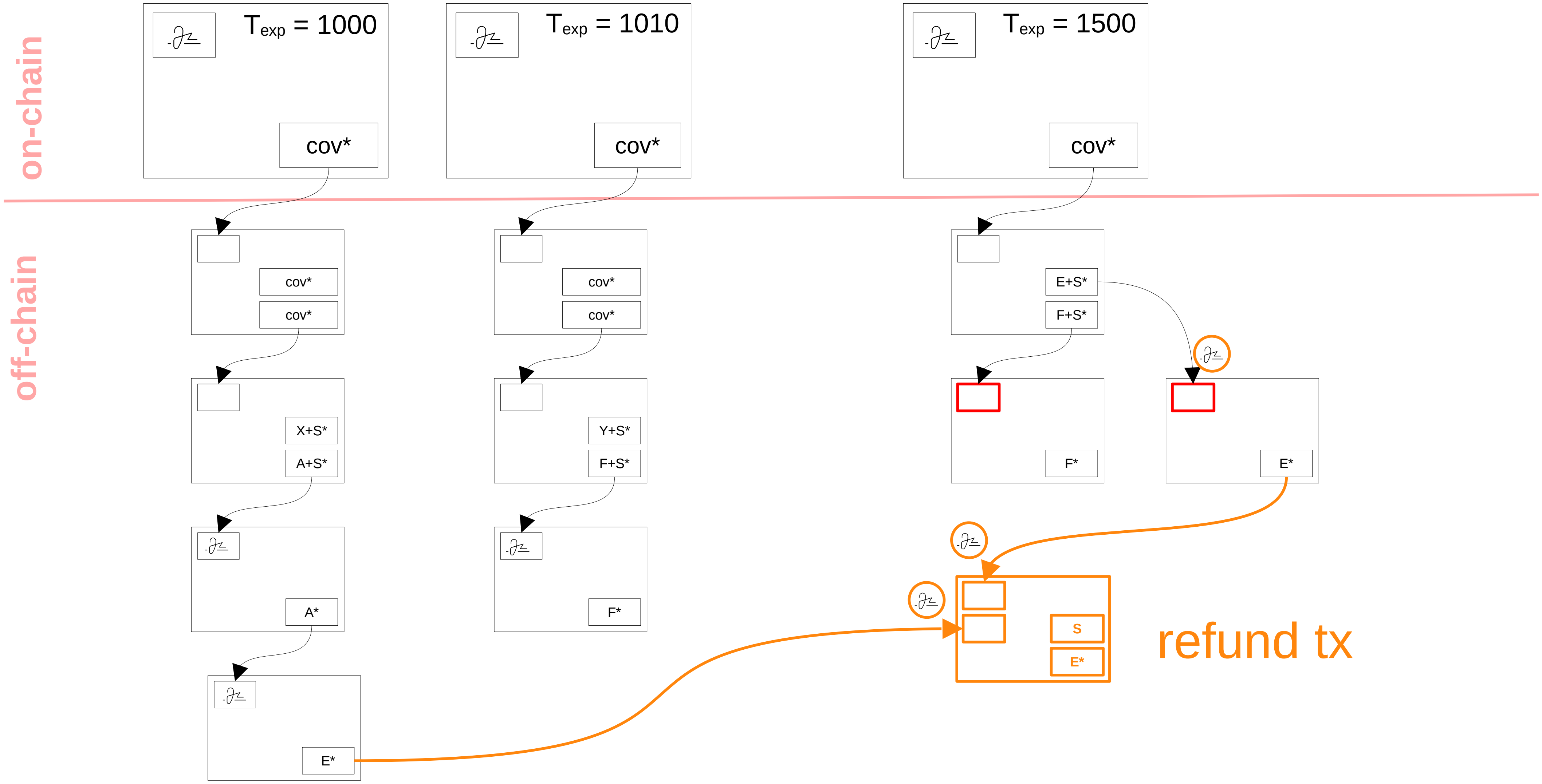

on-chain
off-chain
T_{exp} = 1000
T_{exp} = 1010
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
cov*
cov*
cov*
cov*
cov*
cov*
cov*
cov*
A*
F*
F*
E*
E*

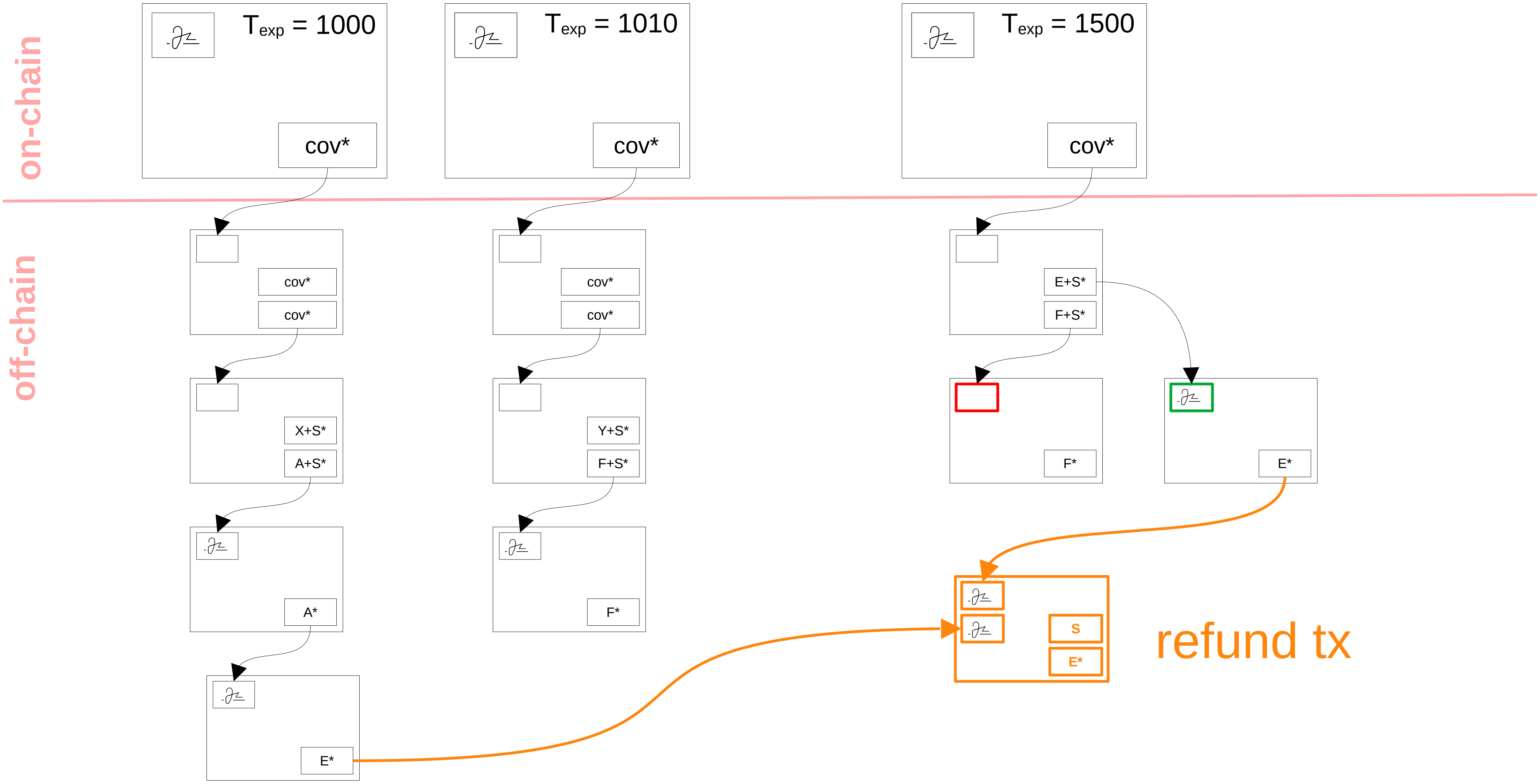

on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
X+S*
A+S*
A*
E*
cov*
cov*
Y+S*
F+S*
F*
E+S*
F+S*
F*
E*

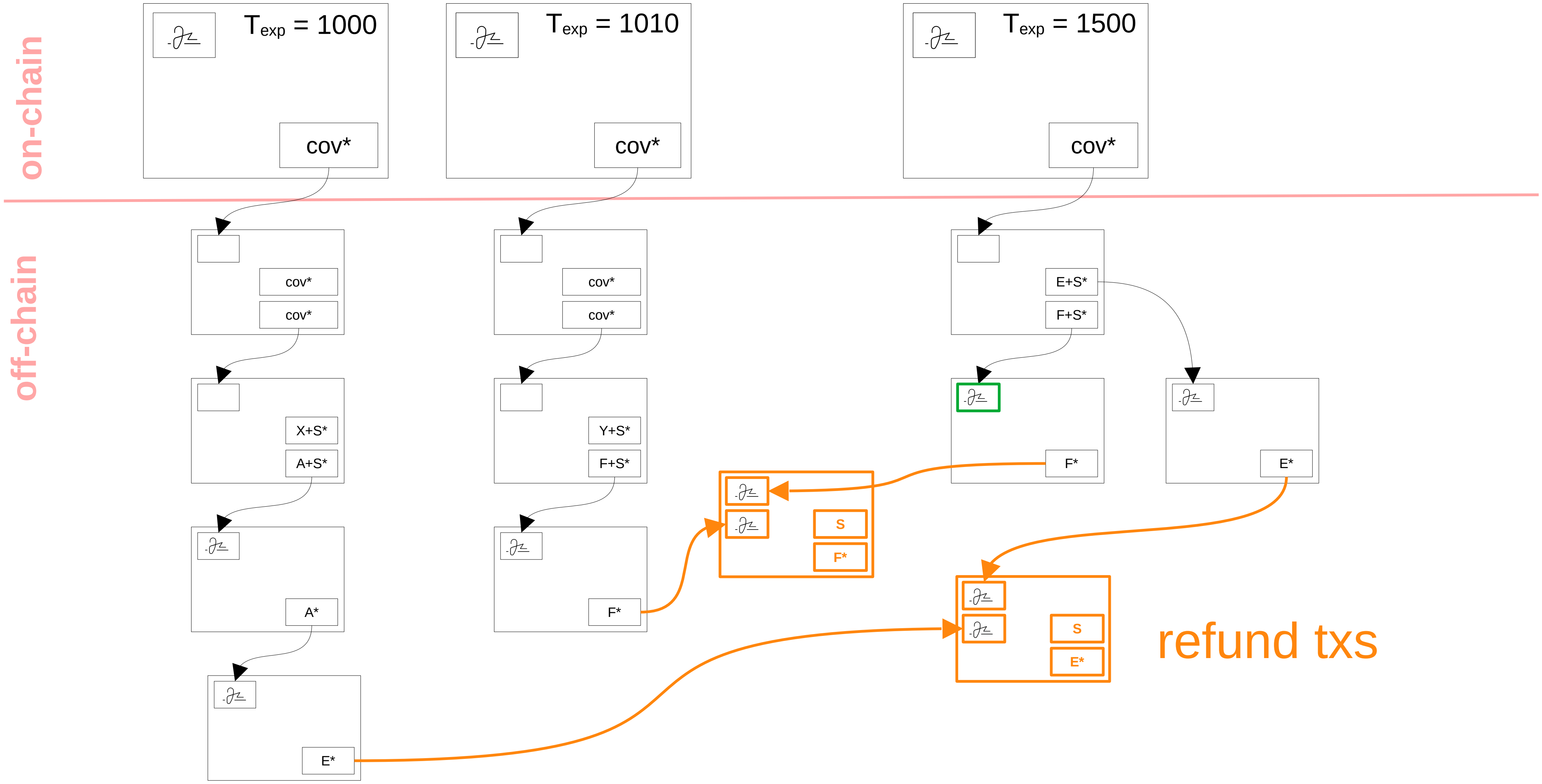

on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
X+S*
A+S*
A*
E*
cov*
cov*
Y+S*
F+S*
F*
E+S*
F+S*
F*
E*

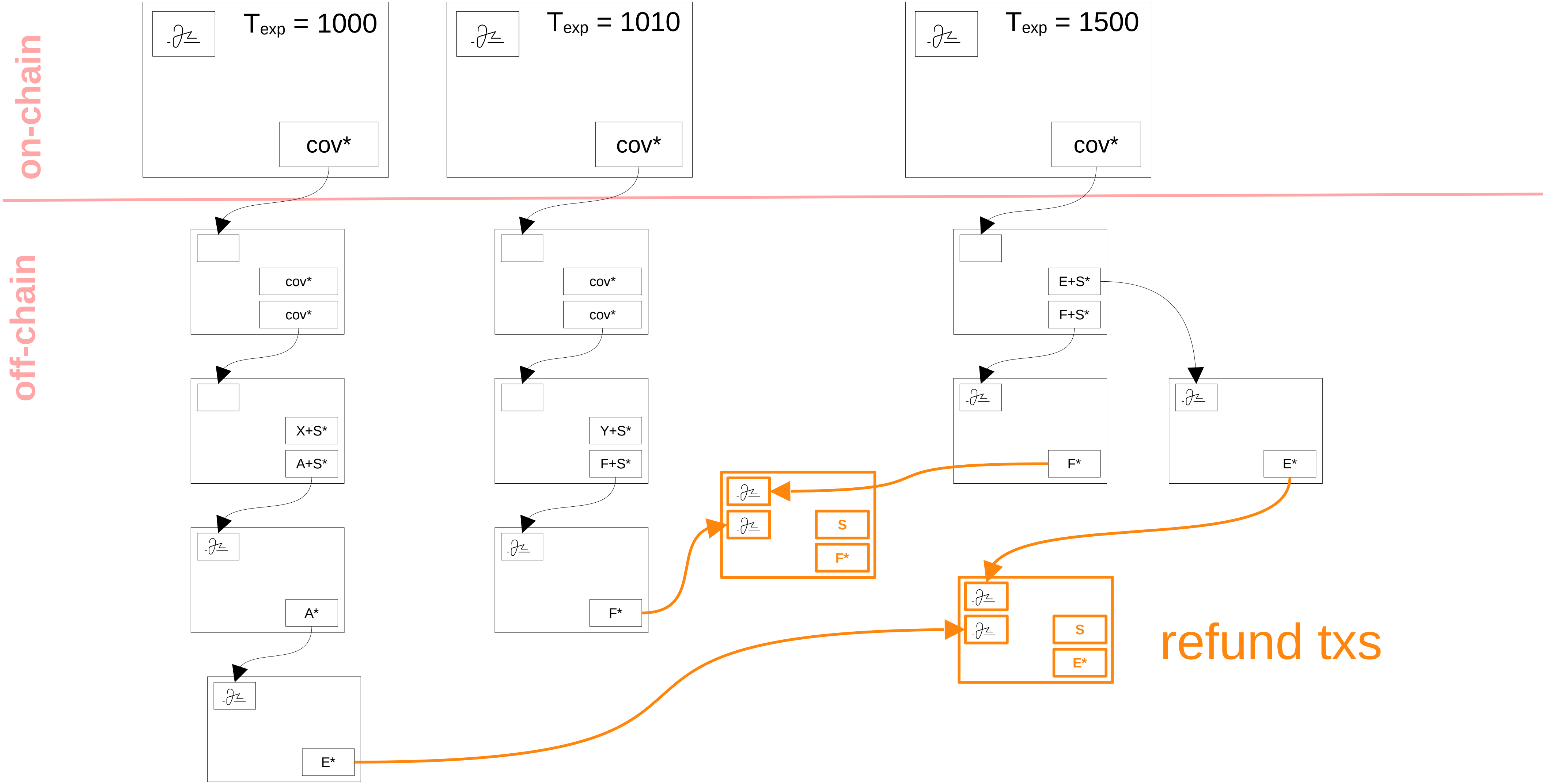

on-chain
off-chain
T_exp = 1000
cov*
T_exp = 1010
cov*
T_exp = 1500
cov*
cov*
cov*
cov*
cov*
E+S*
F+S*
X+S*
A+S*
Y+S*
F+S*
F*
E*
A*
F*
E*


on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
E+S*
F+S*
X+S*
A+S*
Y+S*
F+S*
F*
E*
A*
F*
S
E*
refund tx
E*


on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
E+S*
F+S*
X+S*
A+S*
Y+S*
F+S*
F*
E*
A*
F*
E*
S
E*
refund tx


on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
E+S*
F+S*
X+S*
A+S*
Y+S*
F+S*
F*
E*
A*
F*
E*
S
E*
refund tx

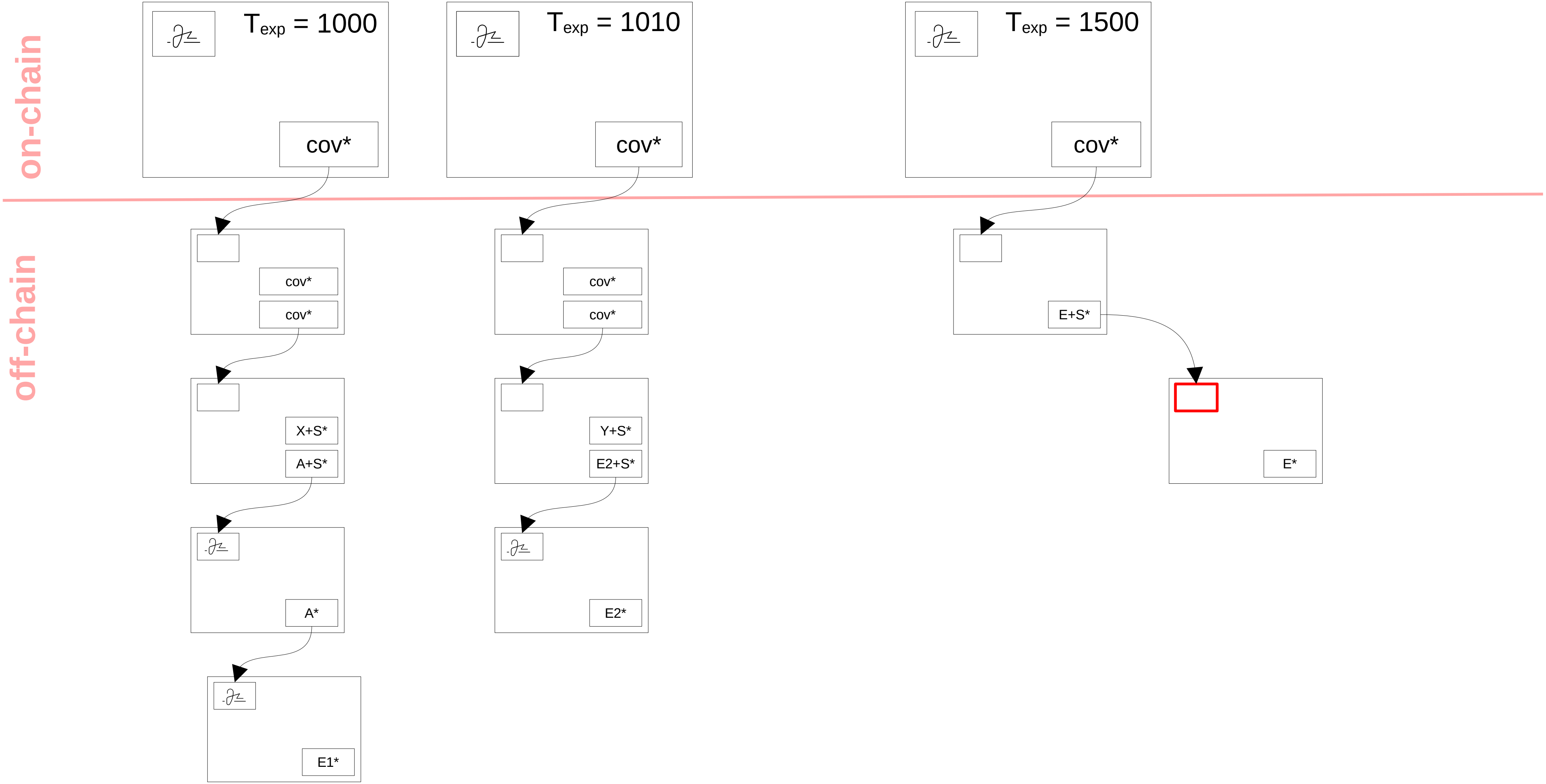

on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
X+S*
A+S*
A*
E*
cov*
cov*
Y+S*
F+S*
F*
S
F*
E+S*
F+S*
F*
E*
S
E*
refund txs

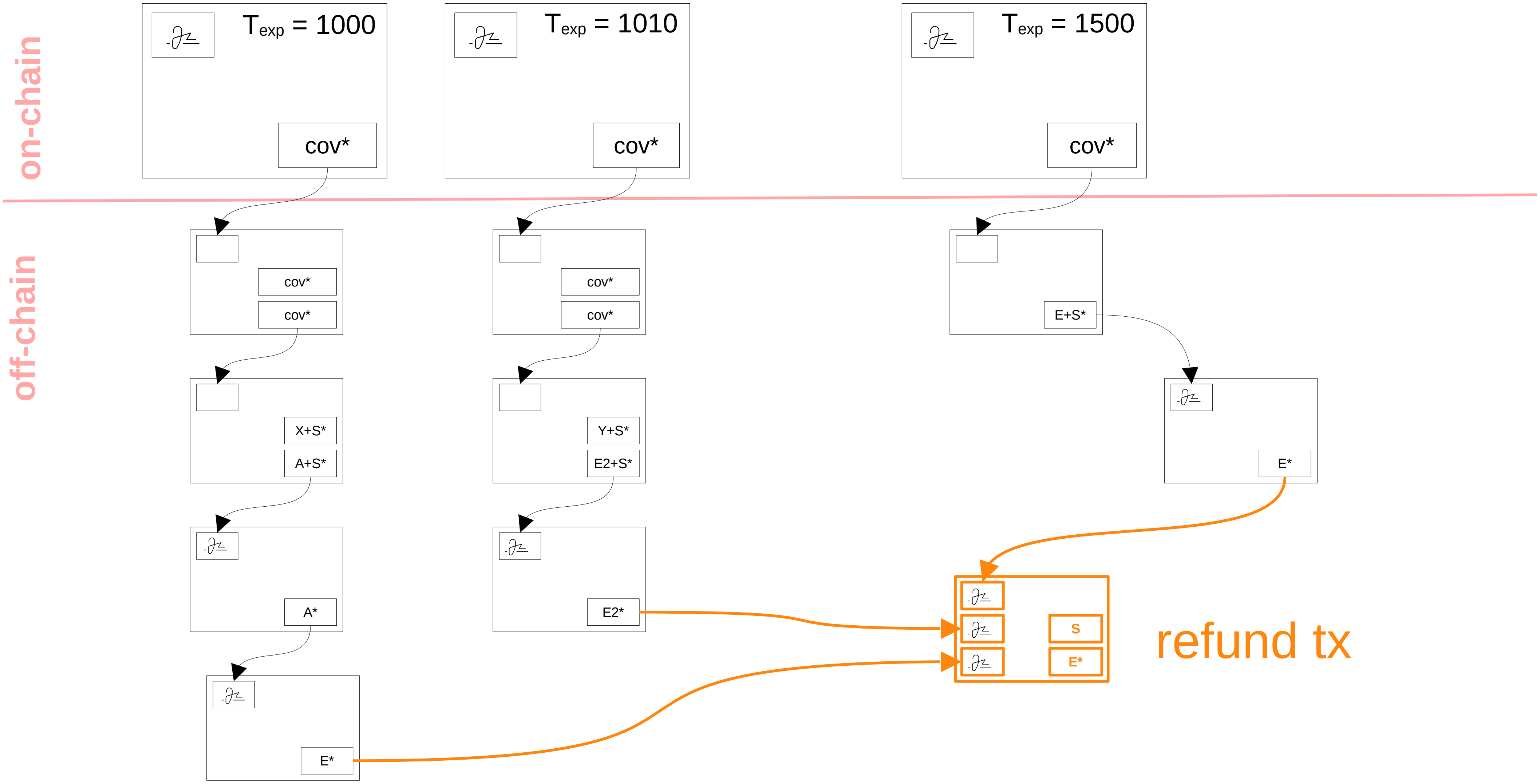

on-chain
off-chain
T_{exp} = 1000
T_{exp} = 1010
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
cov*
cov*
E+S*
F+S*
X+S*
A+S*
Y+S*
F+S*
F*
E*
A*
F*
S
F*
E*
S
E*

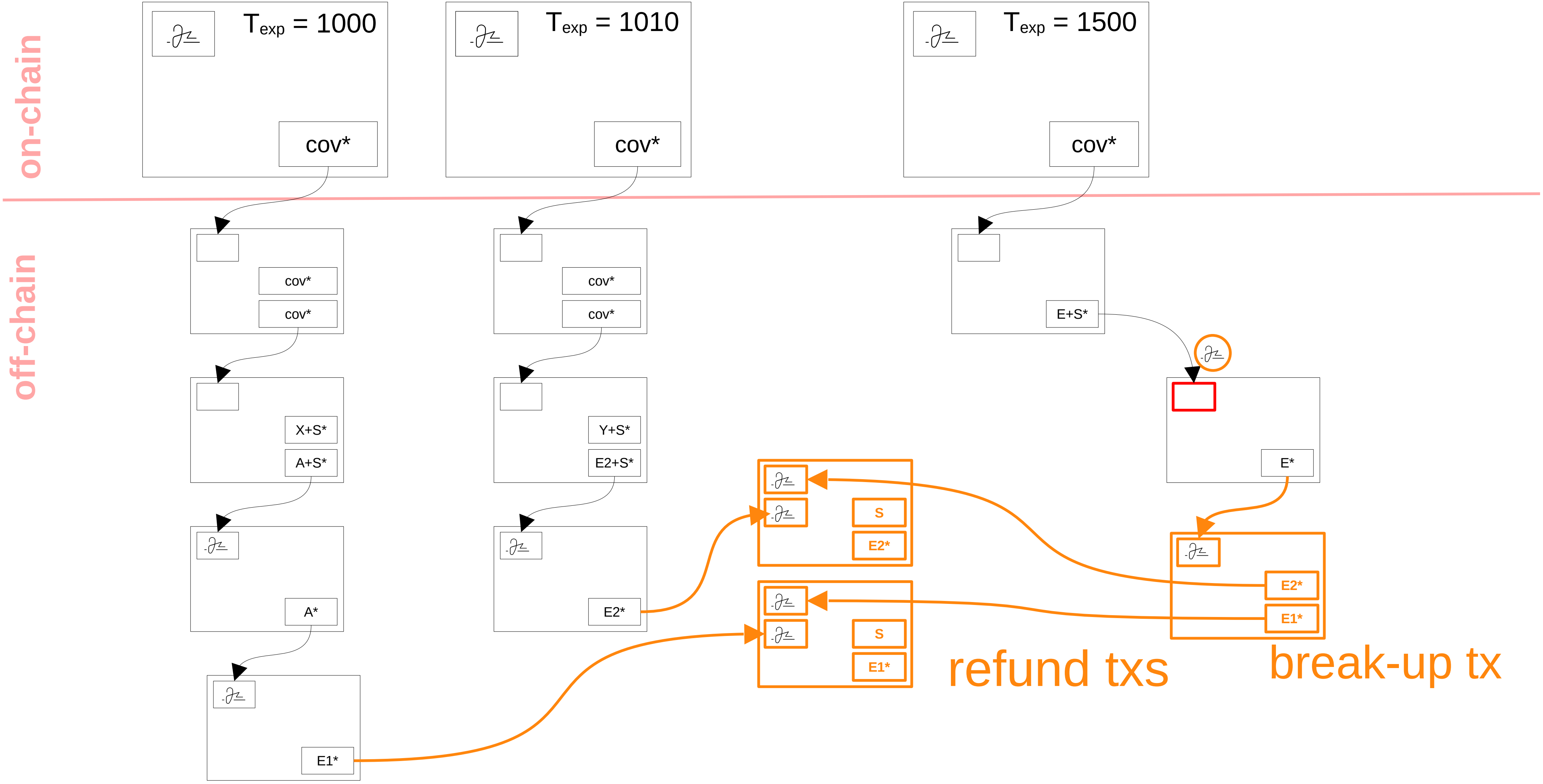
refund txs

Erk* round

- let’s try multi-input


on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
E+S*
X+S*
A+S*
Y+S*
E2+S*
E*
A*
E2*
E1*


on-chain
off-chain
T_{exp} = 1000
T_{exp} = 1010
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
cov*
cov*
E+S*
X+S*
A+S*
Y+S*
E2+S*
E*
A*
E2*
E*
S
E*
refund tx


on-chain
off-chain
T_{exp} = 1000
cov*
T_{exp} = 1010
cov*
T_{exp} = 1500
cov*
cov*
cov*
cov*
cov*
E+S*
X+S*
A+S*
Y+S*
E2+S*
E*
A*
E2*
E1*
S
E2*
S
E1*
E2*
E1*
refund txs
break-up tx

Erk*

- Ark rounds fully async
- no denial-of-service issue
- mobile compatible

Additional CTV benefits

Additional CTV benefits

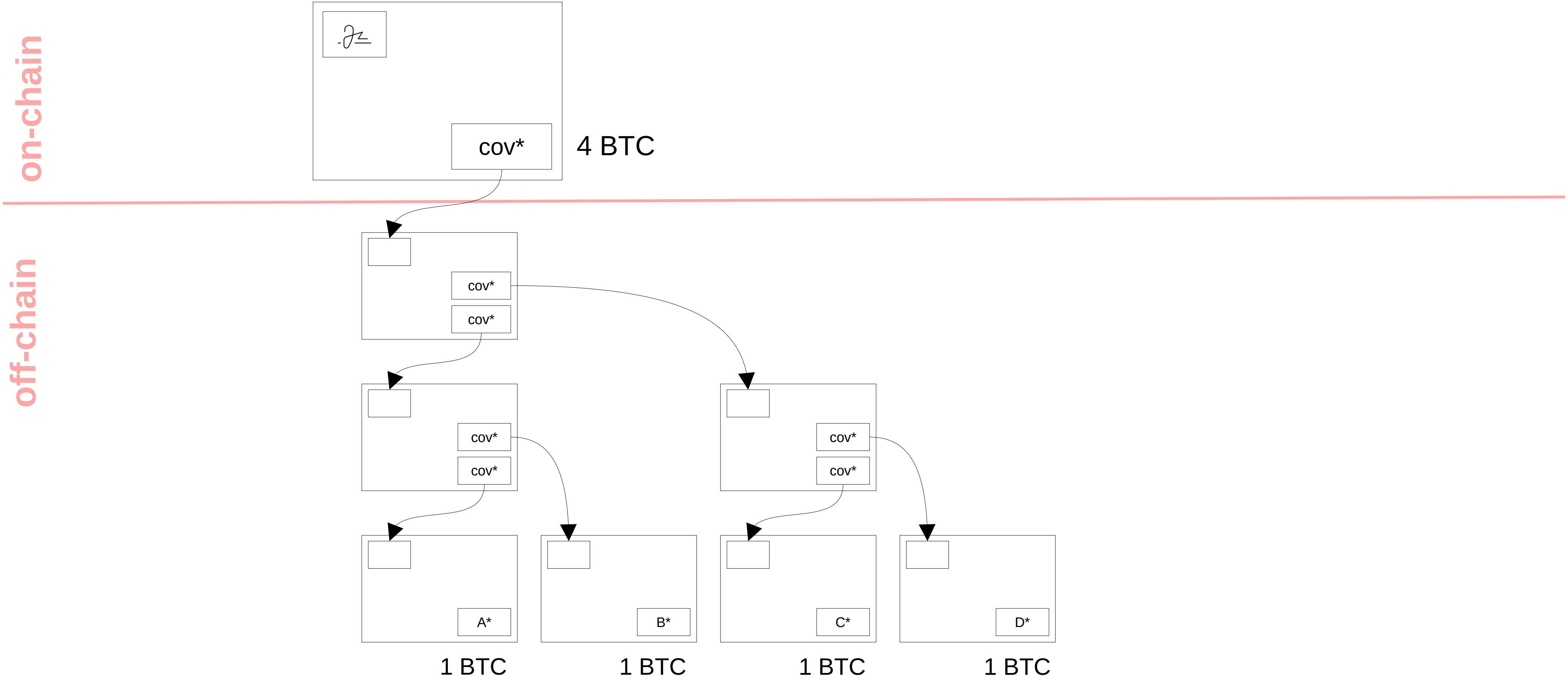
- direct user-to-user in-round payment

Additional CTV benefits

- direct user-to-user in-round payment
- channel-less Lightning receive
  - → HTLC in vtxo

Additional CTV benefits

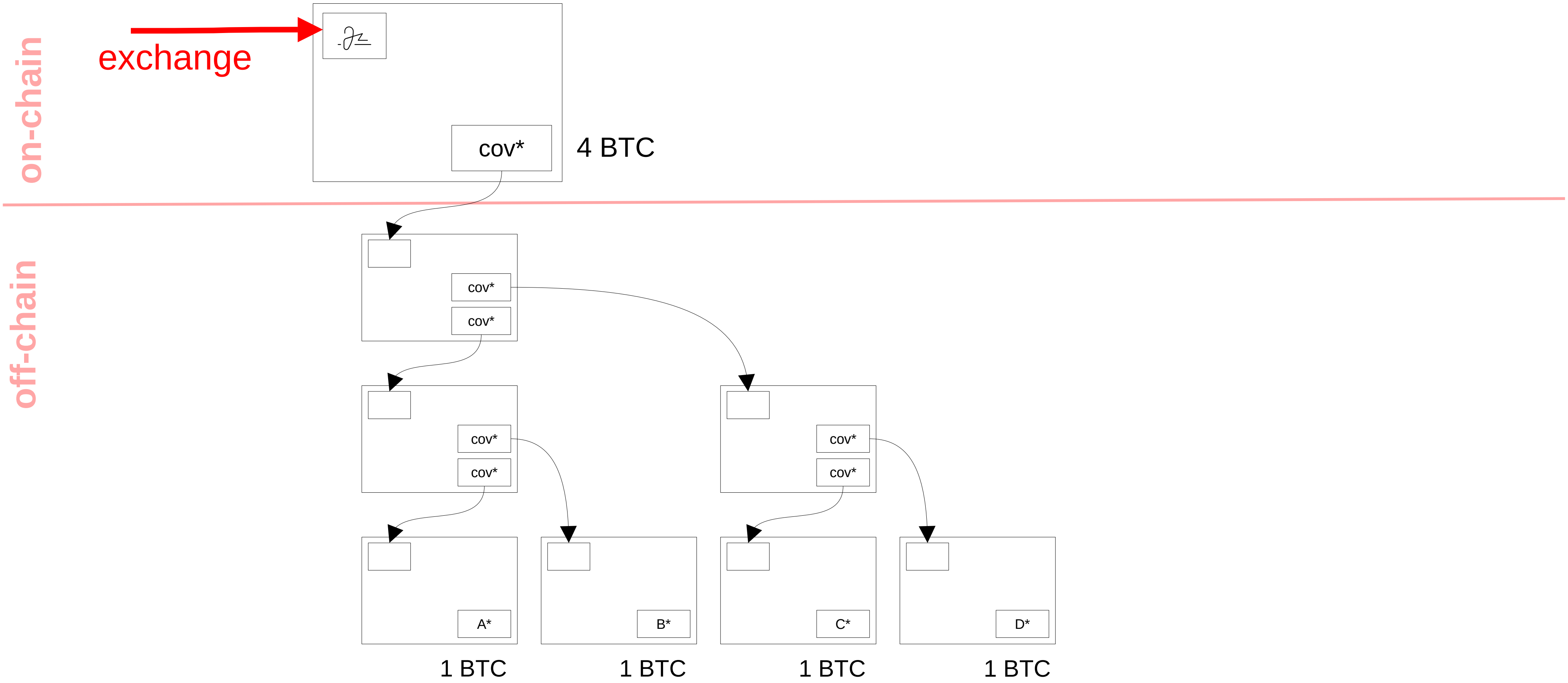
- direct user-to-user in-round payment
- channel-less Lightning receive
- mass pay-outs



S = Ark server pubkey

$A^* = A+S \text{ OR } (A \text{ after } \Delta t)$

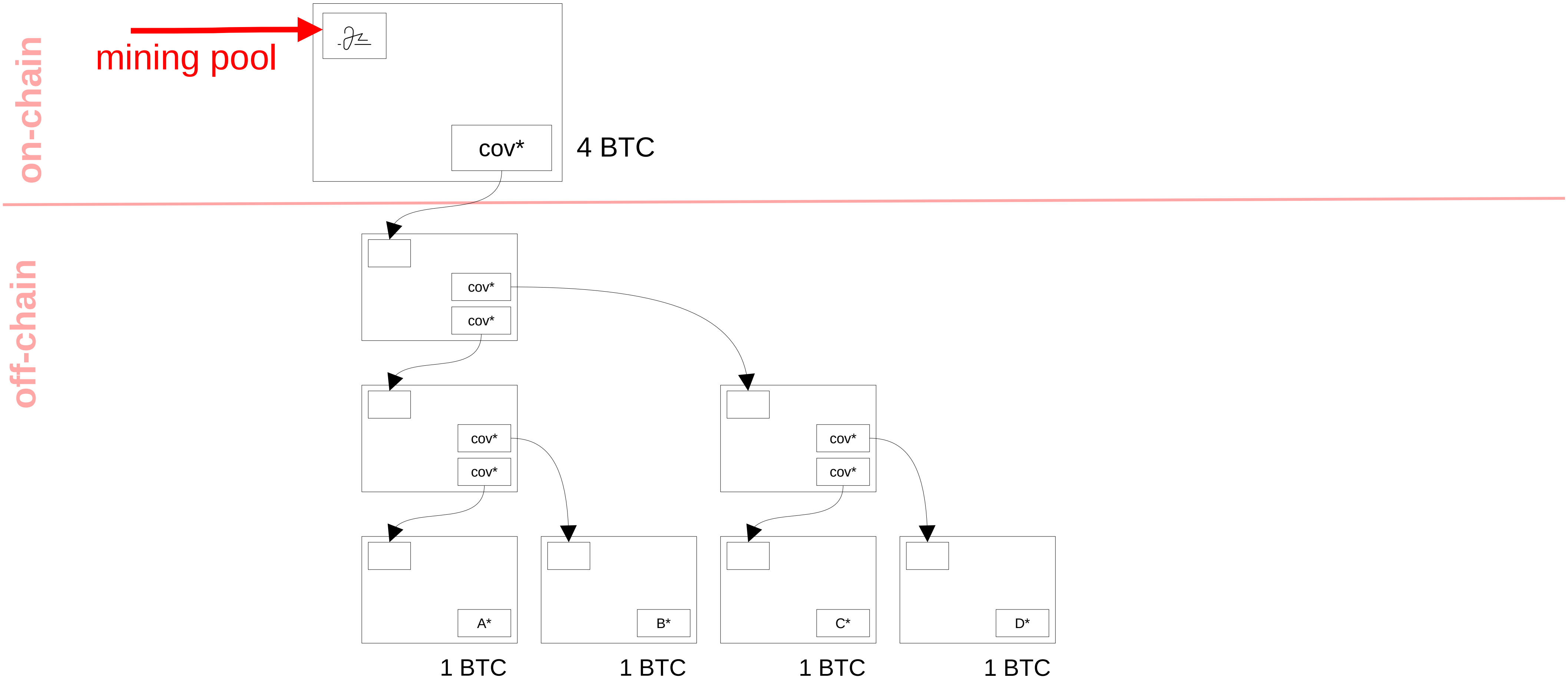
$cov^* = cov \text{ OR } (S \text{ after } T_{exp})$



S = Ark server pubkey

A* = A+S OR (A after Δt)

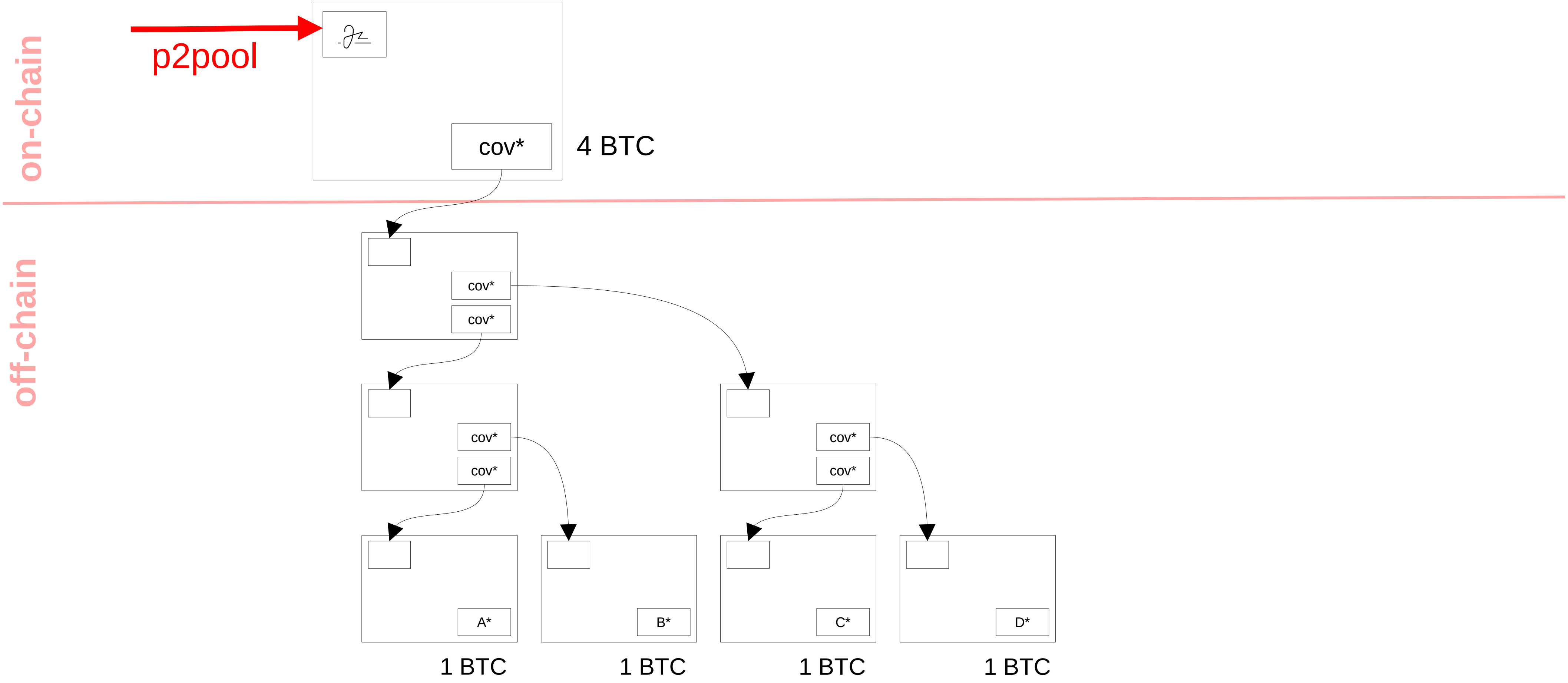
cov* = cov OR (S after T_{exp})



S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})



S = Ark server pubkey

A* = A+S OR (A after Δt)

cov* = cov OR (S after T_{exp})

Additional CTV benefits

- direct user-to-user in-round payment
- channel-less Lightning receive
- mass pay-outs
- vtxo re-issuance by server

Additional CTV benefits

- direct user-to-user in-round payment
- channel-less Lightning receive
- mass pay-outs
- vtxo re-issuance by server
  - → towards trustless re-issuance

Questions?

- slides
  - roose.io
- our Ark client: bark
  - second.tech