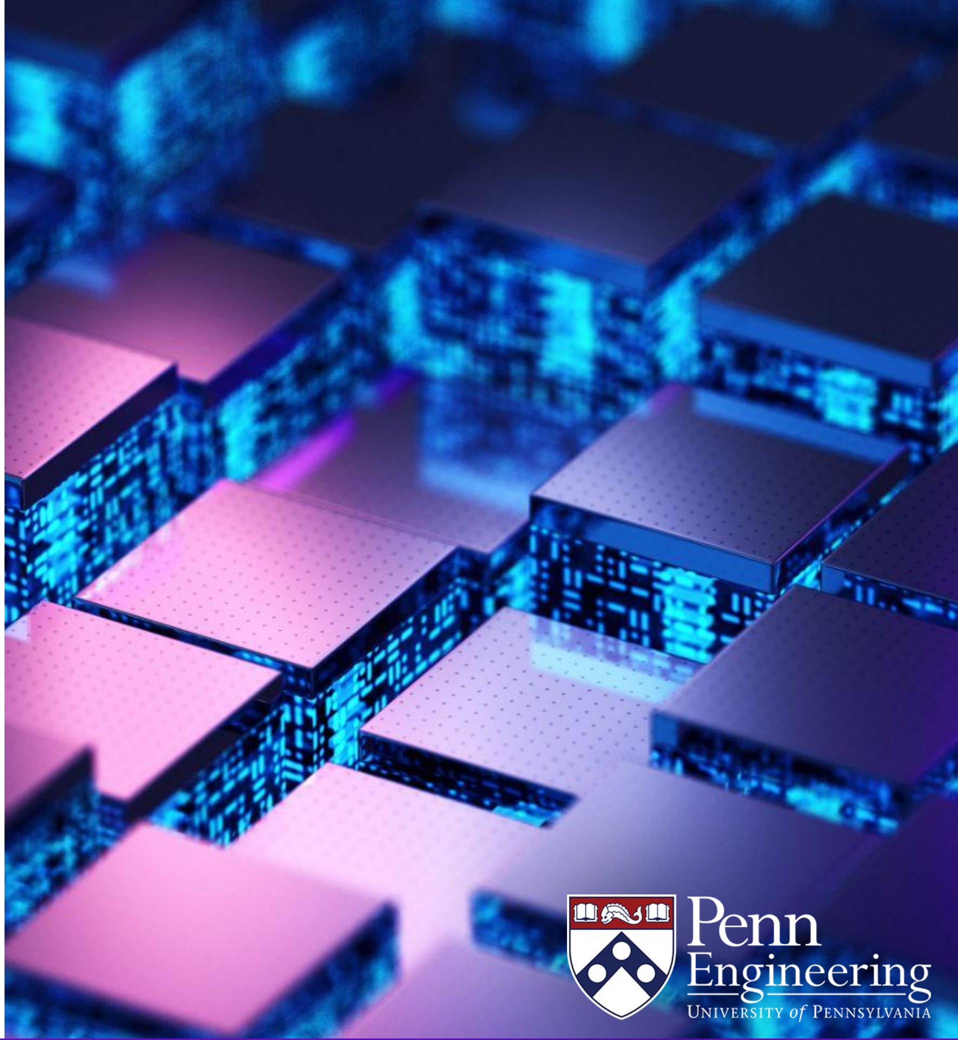
EAS 5830: BLOCKCHAINS

RANDAO

Dr. Brett Hemenway Falk



RANDAO

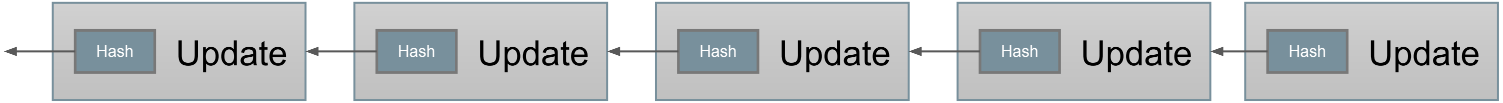
- RANDAO is a scheme for generating randomness to elect block producers in Ethereum
  - Introduced when Ethereum transitioned to PoS

Ethereum Epochs

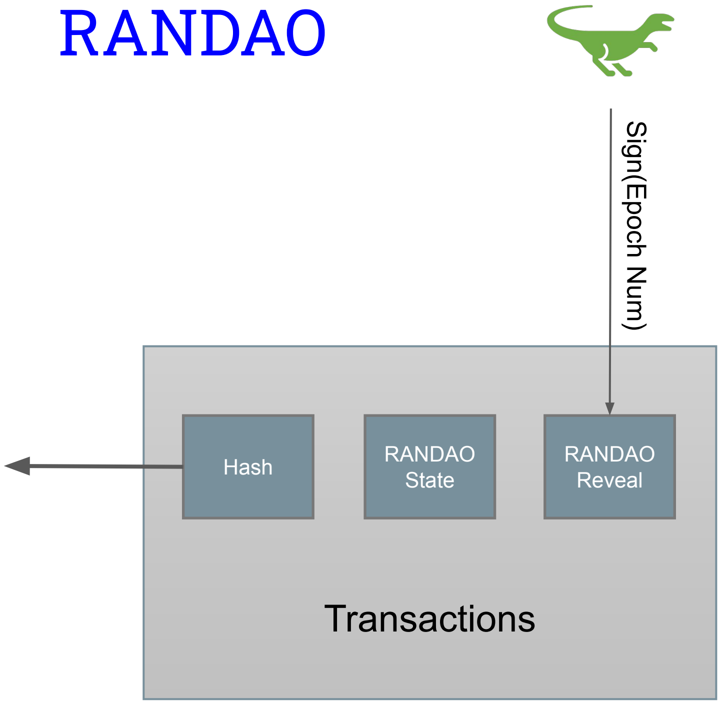
- Ethereum “Epochs” are 32 slots long
  - Slot time is 12s
  - 32·12s = 6.4 minutes
  - Each slot has a single block producer
    - Slot either has a block from that producer
    - Or is empty

RANDAO

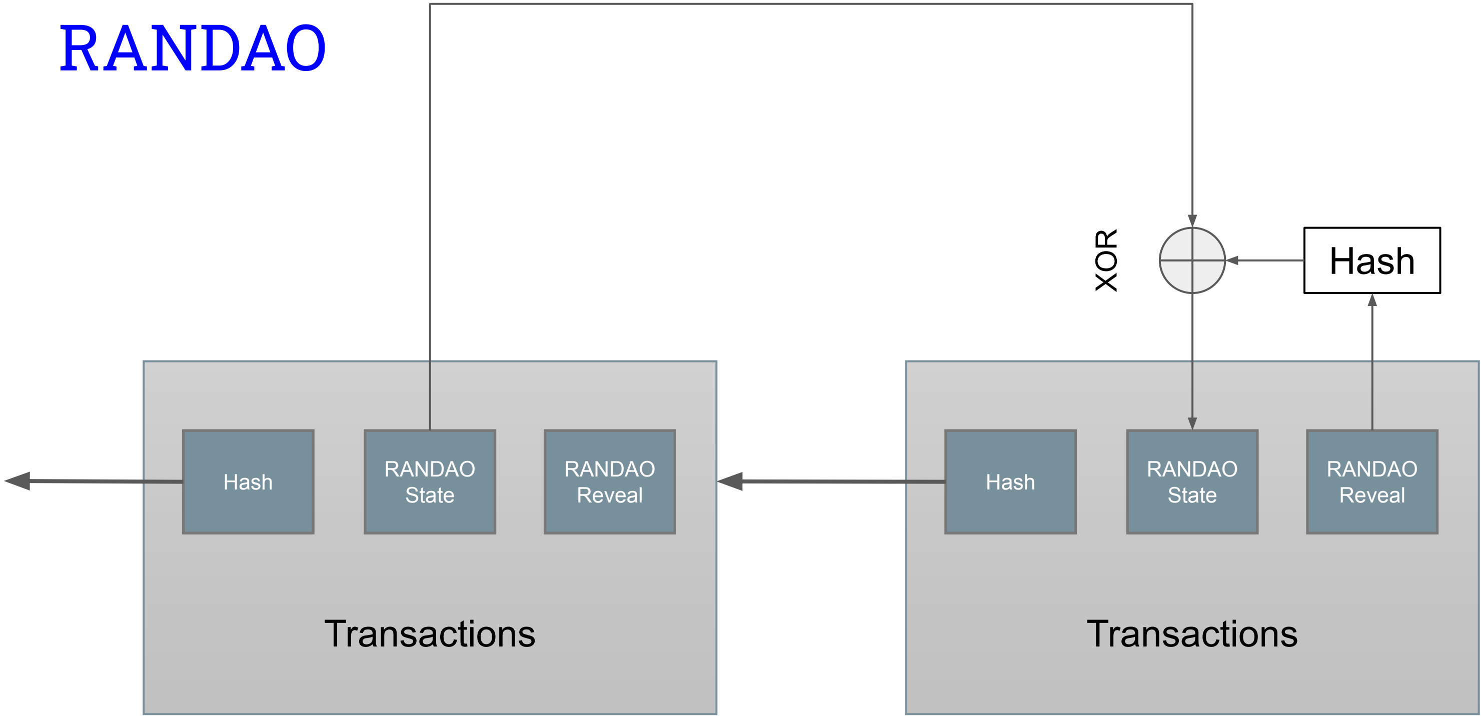
- Block producer provides value: randao_reveal
  - randao_reveal = Validator signature on epoch number
  - This is a verifiable random function (VRF)
  - Block is invalid if signature is invalid



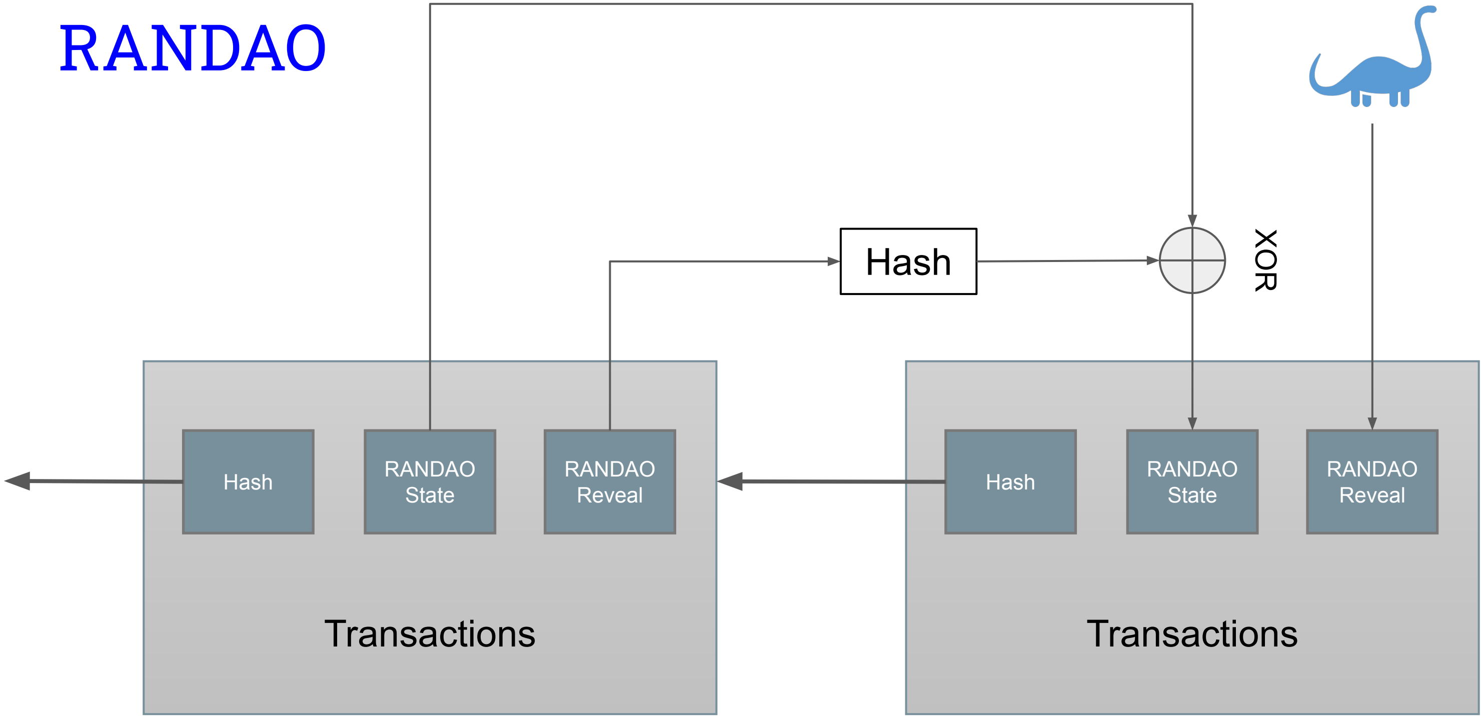
RANDAO

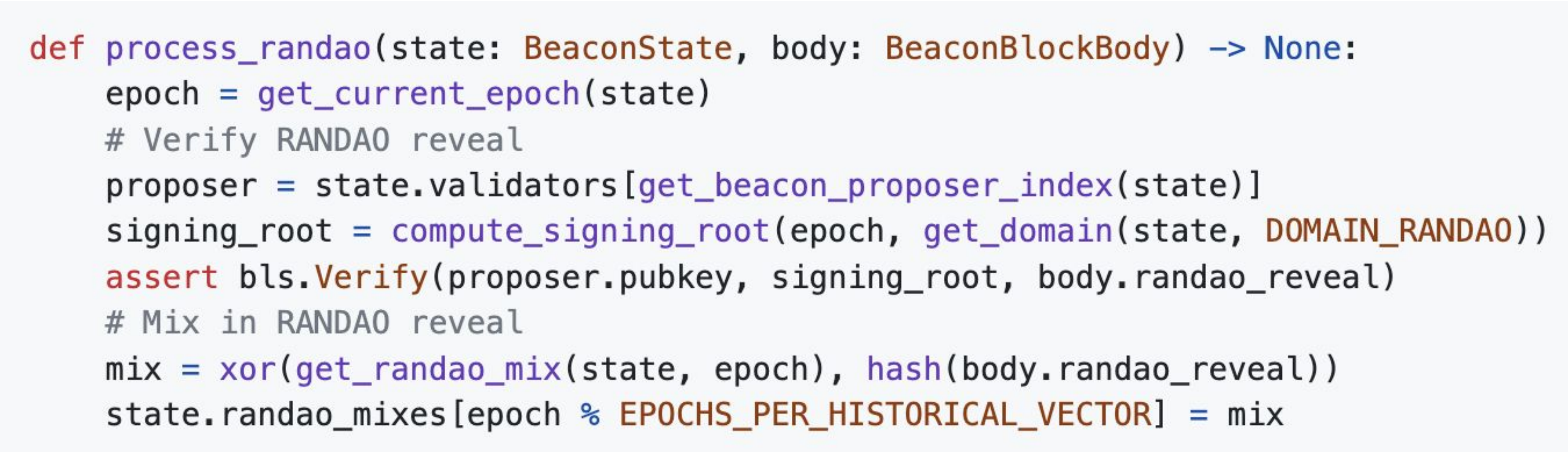

Sign(Epoch Num)
Hash
RANDAO State
RANDAO Reveal
Transactions

RANDAO


XOR
Hash
Hash
RANDAO State
RANDAO Reveal
Transactions
Hash
RANDAO State
RANDAO Reveal
Transactions

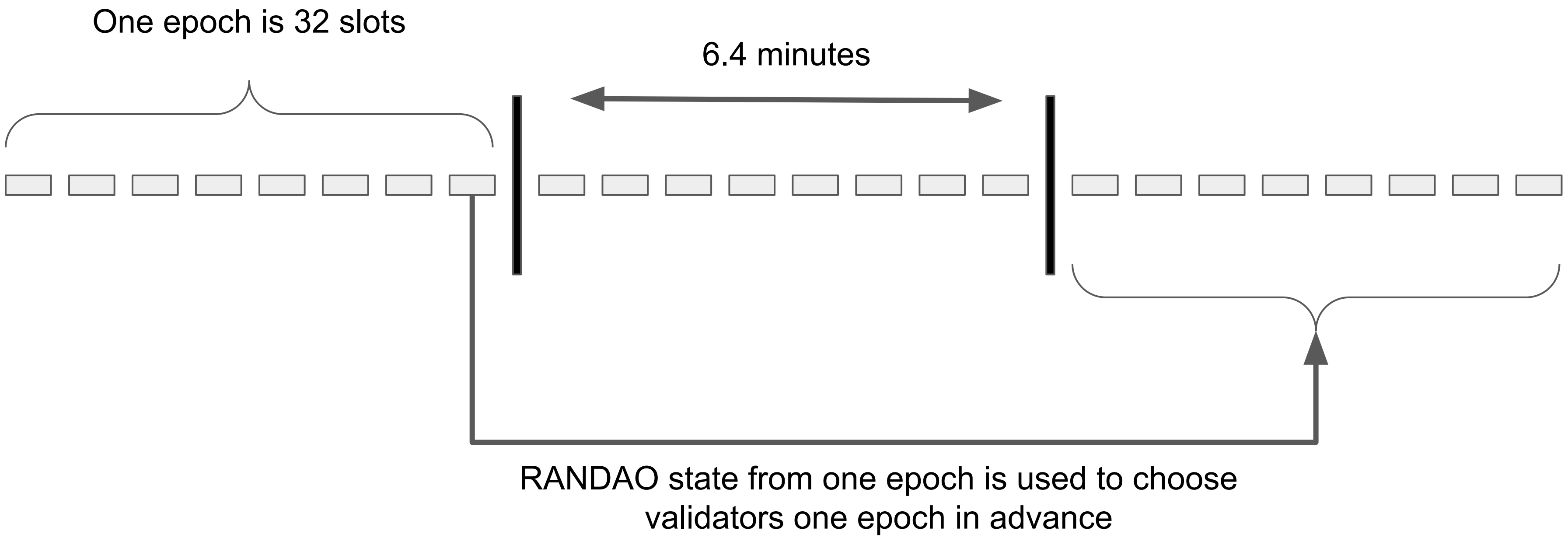
RANDAO


Hash
XOR
Hash
RANDAO State
RANDAO Reveal
Transactions
Hash
RANDAO State
RANDAO Reveal
Transactions



```python
def process_randao(state: BeaconState, body: BeaconBlockBody) -> None:
    epoch = get_current_epoch(state)
    # Verify RANDAO reveal
    proposer = state.validators[get_beacon_proposer_index(state)]
    signing_root = compute_signing_root(epoch, get_domain(state, DOMAIN_RANDAO))
    assert bls.Verify(proposer.pubkey, signing_root, body.randao_reveal)
    # Mix in RANDAO reveal
    mix = xor(get_randao_mix(state, epoch), hash(body.randao_reveal))
    state.randao_mixes[epoch % EPOCHS_PER_HISTORICAL_VECTOR] = mix
```

RANDAO


One epoch is 32 slots

6.4 minutes

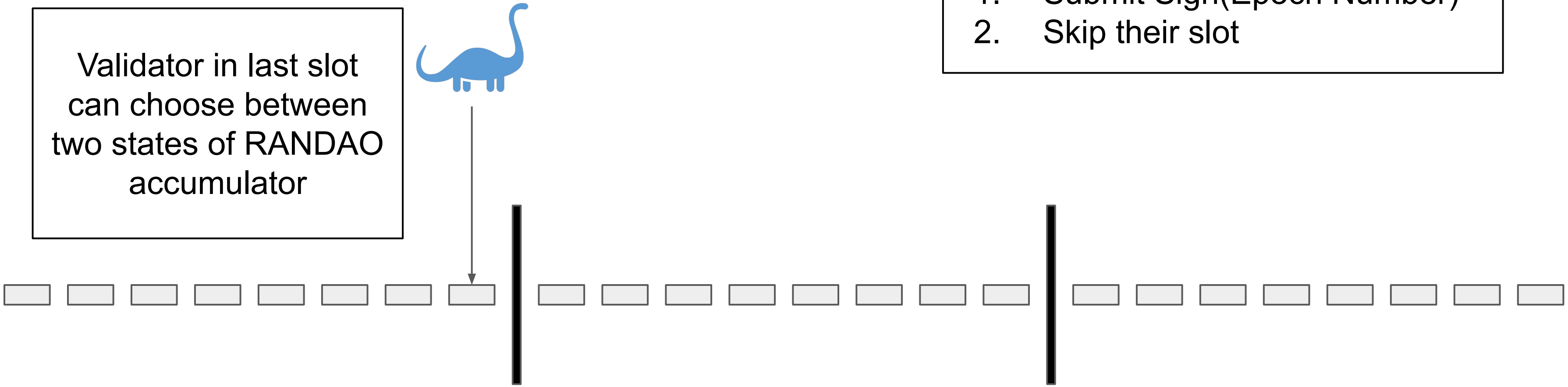
RANDAO state from one epoch is used to choose validators one epoch in advance

Manipulating RANDAO

Validators have 2 options

1. Submit Sign(Epoch Number)
2. Skip their slot

Validator in last slot can choose between two states of RANDAO accumulator

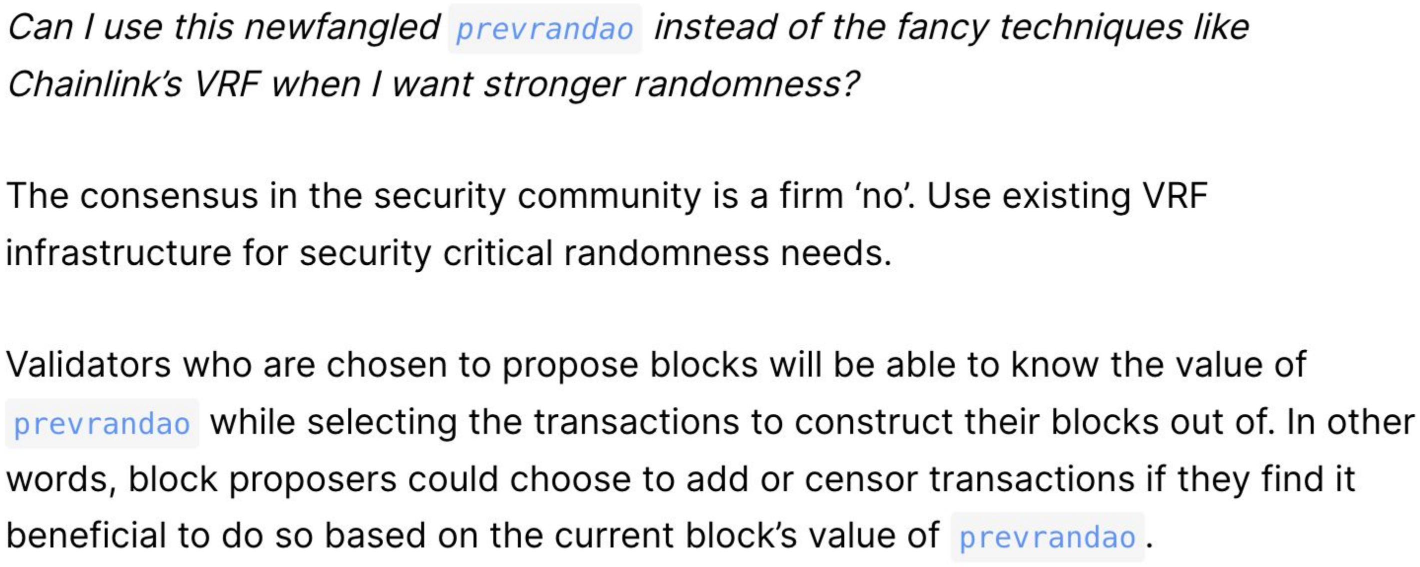


An adversary who is assigned last k slots in an epoch can choose between 2^k possible values of RANDAO – the adversary can choose the one that gives them the most slots in the epoch

Other uses of RANDAO

- RANDAO values are stored in the block
- Can we use RANDAO for other things?

Don’t use RANDAO values for on-chain randomness



Can I use this newfangled `prevrandao` instead of the fancy techniques like Chainlink’s VRF when I want stronger randomness?

The consensus in the security community is a firm ‘no’. Use existing VRF infrastructure for security critical randomness needs.

Validators who are chosen to propose blocks will be able to know the value of `prevrandao` while selecting the transactions to construct their blocks out of. In other words, block proposers could choose to add or censor transactions if they find it beneficial to do so based on the current block’s value of `prevrandao`.