Quickstart Guide with Incredible Squaring

What is Incredible Squaring?

Incredible Squaring is a demo of a minimum viable AVS with full Eigenlayer integration. The purpose of this demo is to illustrate how a value computed offchain (in this case, the square of a number) can be constructed as part of the work operators have signed up to perform as well as how this business logic relates to the AVS contracts. We take the offchan computation, have it signed by multiple Operators, then aggregate the Operators signatures, before finally validating and writing the value onchain. For a video walkthrough please see: Incredible SquaringOverall 5 min Walk-Through Incredible SquaringTaskManager 5 min Walk-Through

Prior to the AVS becoming available for use, the following prerequisite steps must occur:

1. Operators register with the EigenLayerDelegationManager
2. contract.
3. Incredible Squaring AVS is deployed and registered to an implementation of theAVSDirectory contract
4. .
5. Operators register with the AVS through its RegistryCoordinator.

Incredible Squaring AVS Lifecycle (Flow)

Each request for a number to be squared, goes through the following lifecycle (flow)

1. The Task Generator entity sends the number to be squared to the AVS contract (IncredibleSquaringTaskManager.sol).
2. AVS contract emits an event (NewTaskCreated) to represent the new number to be squared.
3. Operators listen to the AVS contract for the event, square the number, sign the result with aBLS signature
4. and send their signature to the Aggregator entity.
5. The Aggregator combines each into a single aggregated signature usingBLS signature aggregation
6. . Once the quorum threshold is met the Aggregator sends the aggregated signature back to the AVS contract.
7. AVS contract verifies that the quorum thresholds were met and that the aggregated signature is valid. If so, the squared number is accepted.

Note theIncredible Squaring repo includes specific links to source code files for additional information.

Incredible Squaring Architecture Unique Characteristics

The Incredible Squaring architecture includes the following components that are unique to its architecture, but not required for all AVSs:

1. BLS signature aggregation is used for aggregation of the Operator response (signatures). AnECDSA version
2. is also available for testing (beta).
3. Task Generator: on-chain task generation was used for Incredible Squaring, however other AVSs may choose to implement off-chain task generation. Please seeEigenDA repo
4. as an example of an off-chain AVS.
5. Aggregator: an entity created to collect the signatures from the operators and aggregate them using BLS aggregation.
6. Centralized entities: the Aggregator, Task Generator entities are centralized in the Incredible Squaring demo. However, decentralizing each component of your architecture over time could be explored
7. RegistryCoordinator contract implementation: unique implementation ofBLSRegistryCoordinatorWithIndices
8. that allows any EigenLayer operator with at least 1 delegated mockerc20 token to opt-in.

BLS vs ECDSA Use Cases within EigenLayer AVSs

BLS (Boneh-Lynn-Shacham) is the default signature scheme used and is considered the most secure option. It is used for potential AVS logic if the AVS design requires aggregating AVS tasks.

- Used for potential AVS logic if the AVS design requires aggregating AVS tasks.
- Optional for registration to AVSs.

ECDSA (Elliptic Curve Digital Signature Algorithm) is an alternative signature scheme that AVS developers may choose to reduce on-chain verification costs, although it is less secure than BLS.

- Required for Operator registration to the EigenLayer core protocol.
- Optional for registration to AVSs.

Note: please do not share (reuse) BLS and ECDSA keys across different AVSs.

Deploy the Incredible Squaring Demo Locally

Visit theIncredible Squaring repo and run the local demo.Previous AVS Development Overview Next How to Build an AVS