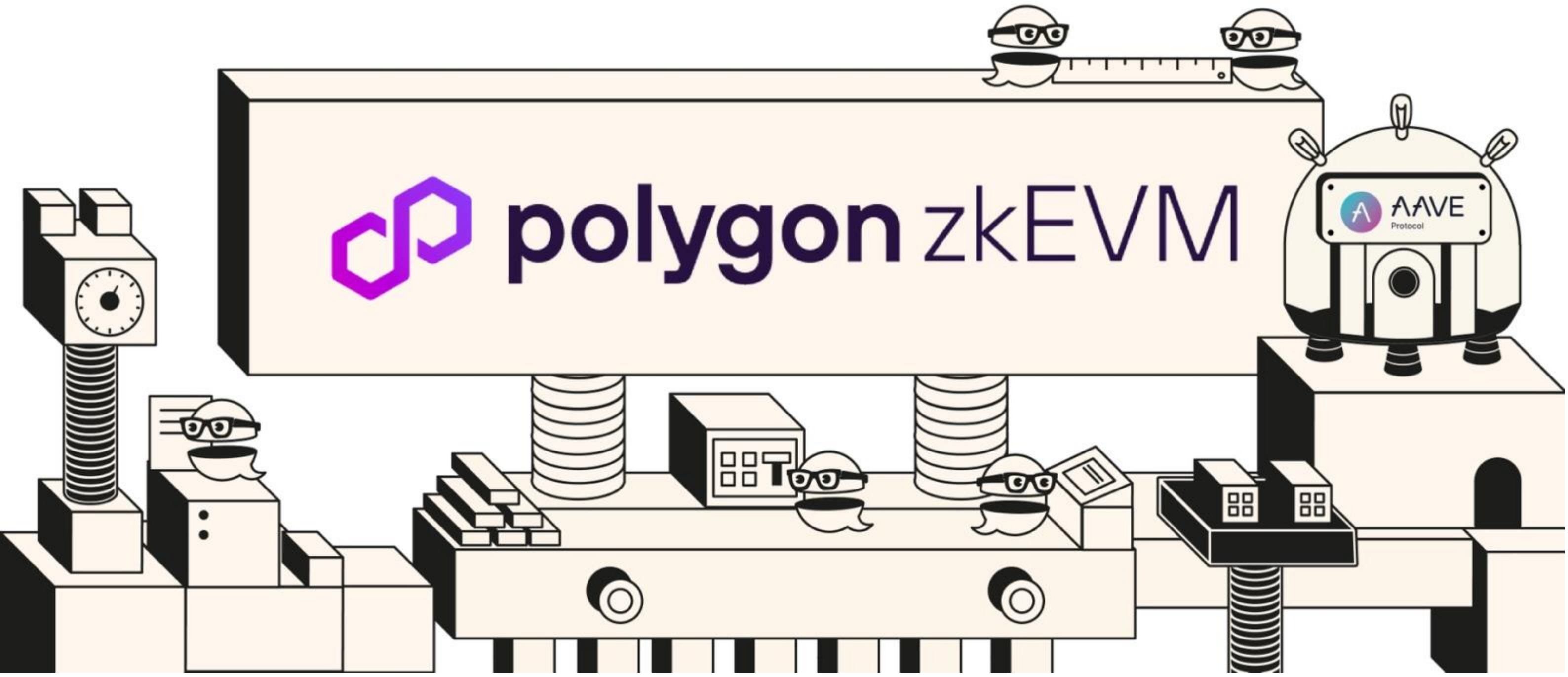
# BGD. AAVE <> Polygon zkEVM.

# INFRASTRUCTURE / TECHNICAL EVALUATION



# Table of Contents

# Introduction

Following our framework of Aave's infrastructure evaluation and the positive outcome of the temperature check, we present the **analysis of Polygon zkEVM regarding its suitability to deploy an instance of the Aave protocol**.

# Disclosure

*This is an independent assessment of different technical components that we consider important for the Aave software to run optimally, not a categorical analysis stating the network is "good" or "bad", and no kind of "requirement" for Aave to be deployed on the candidate network. That decision is up to the Aave governance, no matter our opinion.*

*In addition, currently, we have absolutely no financial/investment/services-engagement/any kind of interest in the Polygon ecosystem.*

*When doing the evaluation, we contacted the Polygon zkEVM team as an important source of information, which has always been exemplary and supportive, but everything in this report comes finally and exclusively from our independent criteria.*

# Report

## 1. Introduction to Polygon zkEVM

Polygon zkEVM is a Layer 2 ZK/validity rollup, or more precisely, as defined in the documentation, a *decentralized Ethereum Layer 2 scalability solution that uses cryptographic zero-knowledge proofs to offer validity and quick finality to off-chain transaction computation*.

zkEVM was released on mainnet on March 27th, 2023, explicitly stating its Beta nature, which means the network is in an early stage, even if completely functional.

## 2. Our methodology

This report is not trying to be a full analysis of the Polygon zkEVM network, but focusing on those aspects that are important from a technical/infrastructural perspective if the Aave community decides to expand the protocol there.

In addition to targeting full coverage of everything affecting Aave, this report tries to be simple enough for participants in the Aave governance to understand. But given its technical nature, it is unavoidable to assume a certain familiarity with the basic concepts touched, like rollups, oracles, RPC nodes, or blockchain explorers, amongst others.

In order to simplify the interpretation of this report, we will evaluate each component important for Aave separately, and assign simplified "grades", defined as follows:

**Optimal.** Fulfilling all minimal requirements, and with extra positive aspects.

**Good.** Fulfilling the requirements, but improvements can be made.

**Acceptable.** Fulfilling the requirements, but with “buts”.

**Needs improvement.** Mandatory requirements not fulfilled at all. Aave will not work properly

# 3. Evaluation

## 3.1. Oracle Infrastructure

At the moment, the Aave protocol uses 3 types of oracles: prices, sequencer uptime, and Proof-of-Reserve. We consider that having Chainlink providing oracles, especially for the most important type (prices), is a must for any deployment, with alternatives only considered really ad-hoc, given the additional complexity added on evaluation/integration.

### 3.1.1. Price feeds

Used by the Aave protocol to price assets listed.

Polygon zkEVM **HAS** Chainlink price feeds available for the major assets on the network.

*The Chainlink team has confirmed with us the price feeds will be available for Aave at launch*.

Additionally, there are other price feed oracles on Polygon zkEVM, but we don't think they should be considered for Aave.

### 3.1.2. L2 Sequencer Uptime feed

"flag" parameter indicating in real-time to the Aave protocol if a sequencer of a rollup (or any network involving some centralization on sequencing of transactions) is properly running.

Polygon zkEVM **DOESN'T HAVE** Chainlink L2 Sequencer Uptime. *The Polygon zkEVM has confirmed us that they have set as priority providing the equivalent of this oracle, or natively, or via Chainlink.*

### 3.1.3. Proof-of-Reserve feeds

The utility of Proof-of-Reserve on zk-rollups still needs further research, in what concerns the "official" bridge of the network. Therefore not having them available at the moment doesn't have any influence on this analysis.

## 3.2. Blockchain explorer

For Aave, same as for any other blockchain project, block explorers like Etherscan are a fundamental component, specifically for the following:

1. Verifiable smart contracts code visualization.
2. Read/write interface with smart contracts.
3. Basic data analysis tool for misc aspects like token holders.

The official blockchain explorer of the Polygon zkEVM network can be found at https://zkevm.polygonscan.com/, and it is a white-labeled instance of Etherscan.

### 3.3. Compatibility with Ethereum RPC standard

Basic compatibility with the Ethereum nodes RPC de-facto standard (eth_, *web3_*) is quite an important requirement for Aave or any other protocol, given that it helps to have tools built for Ethereum (or other similar networks) working out-of-the-box just by plugging them to node, of Polygon zkEVM in this case.

Polygon zkEVM **HAS** complete compatibility with the standard, including some extra custom endpoints, and lacking some non-standard.

### 3.4 Compatibility with Ethereum account format (addresses)

One of the strengths of non-Ethereum networks (e.g. Polygon, Avalanche C-Chain, etc.) is its compatibility with Ethereum private/public keys of accounts. This allows existing account holders on those networks to use the others without creating an ad-hoc wallet for it.

Polygon zkEVM **IS** fully compatible with the Ethereum account format.

## 3.5. RPC public endpoints and providers

Basic and reliable public RPC infrastructure is a must for Aave, as it is the way to connect to the network, both for data reading and transaction submission.

Polygon zkEVM has available the following public RPC endpoints:

- zkevm-rpc.com
- polygon-rpc.com/zkevm

Additionally, Node-as-a-Service providers like Alchemy or Ankr are available on Polygon zkEVM.

During testing, we noticed some temporary but generalized (cross-provider) instability on RPCs, which the Polygon zkEVM team has confirmed they are improving.

### 3.6. Custom behavior (lack of) of the execution layer

Whenever a network has custom/extended behavior with respect to Ethereum, it is important to be aware of it and evaluate if it has any impact on the Aave protocol.

Examples of this potential behavior are the presence of new pre-compiles (compared with Ethereum or similar rollups like Optimism), EVM opcodes, native account abstraction/meta-transactions, chainId definition out of the norm, etc.

Independently, even if not doing a full evaluation of the zkEVM implementation, we have checked the following, given that historically have been critical aspects:

- The chainId behavior is appropriate, with the id 1101 for Polygon zkEVM not clashing with any other.
- Polygon zkEVM has equivalence (address and logic) with the ecRecover and identity Ethereum pre-compiles, which covers the needs of Aave.
- Polygon zkEVM has equivalence with Ethereum pre-1559, to be upgraded to post-1559 in early 2024.

Additionally, we have a confirmation from the Polygon zkEVM team that there is no custom basic behavior affecting the model of execution on the virtual machine.

### 3.7. Support of wallet providers

Wallet products like Metamask, Ledger, Coinbase Wallet, and others, are fundamental pieces of the infrastructure for users to access the Aave protocol. So it is a strong requirement for a network to be supported by a subset of them.

Given its EVM compatibility in the context of this document, Polygon zkEVM is transparently supported by the majority of all the chain-agnostic wallets, like Metamask, Ledger, Coinbase Wallet, or Frame.

It is possible that other types of wallets (e.g. based on smart contracts) don’t support Polygon zkEVM, but it is something expected in a young network and doesn’t have any negative consequence from the infrastructure perspective.

## 3.8. On-chain multi-signature infrastructure

The permissions on the Aave ecosystem are directly held by on-chain governance smart contracts or scheduled to be like that once cross-chain governance infrastructure can be applied across all the networks.

However, different protection/emergency mechanisms, like the capability of canceling cross-chain governance proposals, or pausing an Aave asset/pool, depend on the Aave Guardian, who is capable of acting faster than the governance process.

Consequently, having on-chain multi-signature contracts is a requisite to have Aave on a different network, with a high preference for industry-standard tools like Gnosis Safe.

Polygon zkEVM **HAS** an instance of the Gnosis Safe contracts on-chain, but the user interface and server infrastructure are not the official Safe, but a fork on https://zksafe.quickswap.exchange/.

The Polygon zkEVM team has confirmed the maintainer team (Quickswap) is in close contact with the Safe team, and the upstream is fully aligned.

Additionally, an official Safe instance should be live soon.

### 3.9. Transactions simulation infrastructure (fork)

Lately, a really important development experience component is the ability to execute test transactions (simulations) on forked production networks.

A good part of the tooling around Aave depends on simulations by using different libraries/frameworks like Hardhat, Foundry, or Tenderly. This way, it is possible to rapidly prototype new developments, get extra assurances on governance proposals and protocol upgrades, change risk parameters, etc.

As it is our main smart contracts development framework, we have tested that it is possible to do fork simulations on Polygon zkEVM with Foundry. Given its EVM compatibility, it should be perfectly doable with Hardhat too.

Currently, Tenderly is not integrating Polygon zkEVM, but the Polygon team is working on a potential integration.

### 3.10. Chain data/indexing solutions

For different projects and entities integrating Aave, and even if not a blocker for deployment, it is important that solutions like TheGraph or Dune are operating on the candidate network, to avoid building from scratch data pipelines.

Polygon zkEVM is supported on TheGraph.

Polygon zkEVM is not supported by Dune.

### 3.11. Bridging infrastructure: assets, messages

Given the central role of Ethereum in the DeFi and Aave ecosystems, proper bridging infrastructure to/from is a must for any candidate network.

There are 2 types of bridging affecting Aave: assets and generic messaging. In the case of Polygon zkEVM, the state of these 2 types is as follows:

- **Assets.** sub-use case the generic messaging infrastructure supported by Polygon zkEVM. For end users, it is possible to bridge assets via https://wallet.polygon.technology/.

  Regarding the security of the ERC20 smart contracts for bridged assets, we have checked the main tokens share the same implementation: a simple ERC20 based on the OZ version, with burn/mint capabilities by an entity defined as “bridge”.

- **Generic messaging.** zkEVM supports bi-directional generic message passing.

In addition to the default bridging mechanism of zkEVM, there are other providers available, but this is not so important for Aave at the moment, as a.DI will use mainly the canonical one.

## 3.12. Commitment in security incidents

Having proper mechanisms and procedures to prevent and react to security incidents is something quite fundamental for any platform and application, and networks like Polygon zkEVM are no exception.

We have directly checked with the team and confirmed the following:

- On the prevention side, Polygon zkEVM has an Immunefi bug bounty campaign running.
- Polygon zkEVM has a sizeable in-house team of security experts.
- We are aware Polygon zkEVM collaborates with external thread monitoring platforms in aspects like bridge security.
- If any incident happens, Polygon has confirmed to us that will immediately react and execute at least the following measures:
    - Act as fast as possible to protect against damage.
    - Contact the technical side of Aave.
    - Engage independent security experts to assess the security problem and the reaction to it.
    - Properly community the Aave community about the incident, and the next steps.
- A private channel of communication will be kept between the Polygon zkEVM team and the assigned technical team of the Aave community (e.g. BGD), for any necessary update concerning the network and consequently, Aave on Polygon zkEVM. During our evaluation, the team has always been really responsive.

## 3.13. Network security/technical model

At the core of any candidate network analysis are its morphology (which type of network it is) and security/operational models (how it works and which parties are involved in the control over the network; decentralization degree).

Regarding its morphology/type, Polygon zkEVM is a zk/validity roll-up, meaning that it leverages zk-knowledge technology (a combination of SNARK and STARK) to provide high scalability for blockchain transactions.

Regarding its security/operational model, there are multiple aspects to analyze.

### 3.13.1. Transactions flow

A detailed explanation can be found HERE, but to summarize:

1. A user interacting directly with zkEVM builds a transaction and submits it to the JSON RPC API endpoint of a network node.
2. The received transactions are stored in the Pool component database of pending L2 transactions. *Additionally, users will be also able to submit transactions to the zkEVM bridge contract on Ethereum (forced batches), which will introduced in the Pool. This mechanism is not active yet.*
3. The Sequencer picks transactions from the Pool, orders and batches them, and stores them to the node's StateDB.

4. The Prover(s) queries the node’s StateDB to read the data of the new batches to be proved.
5. The transaction batches are executed by the prover(s), generating all the necessary metadata and the zk-proof(s). In case of multiple provers (as per network capacity), the aggregator aggregates the proofs into a single proof. Then it stores the proof and its related data in the node’s StateDB.
6. Periodically, a SequenceSender component checks for and takes “closed” batches from the stateDB, and forwards them to the EthTxManager
7. The EthTxManager then forwards them to Ethereum to be included in a block.
8. Transactions get verified via the L1 zkEVM smart contract and included in Ethereum.
9. Finally, the Synchronizer, monitoring events of the L1 zkEVM smart contract realises that a new batch is consolidated and stores this information in the node’s StateDB.

### 3.13.2. Data availability

On Polygon zkEVM, all the transaction data is submitted to Ethereum, so data availability boils down to Ethereum, which can be considered the highest standard at the moment.

### 3.13.3. Upgradeability and control model

The main centralization point of the network at the moment is the Sequencer component, in charge of gathering and sending to proof (via the Aggregator) the transactions to be included in the rollup.

It could be possible to skip the Sequencer by using so-called forced transactions, but currently, this mechanism is not active.

For self-verification of smart contract roles, the major Polygon zkEVM contract addresses can be found HERE.

### 3.13.4. Security audits

The security audits of zkEVM can be found HERE.

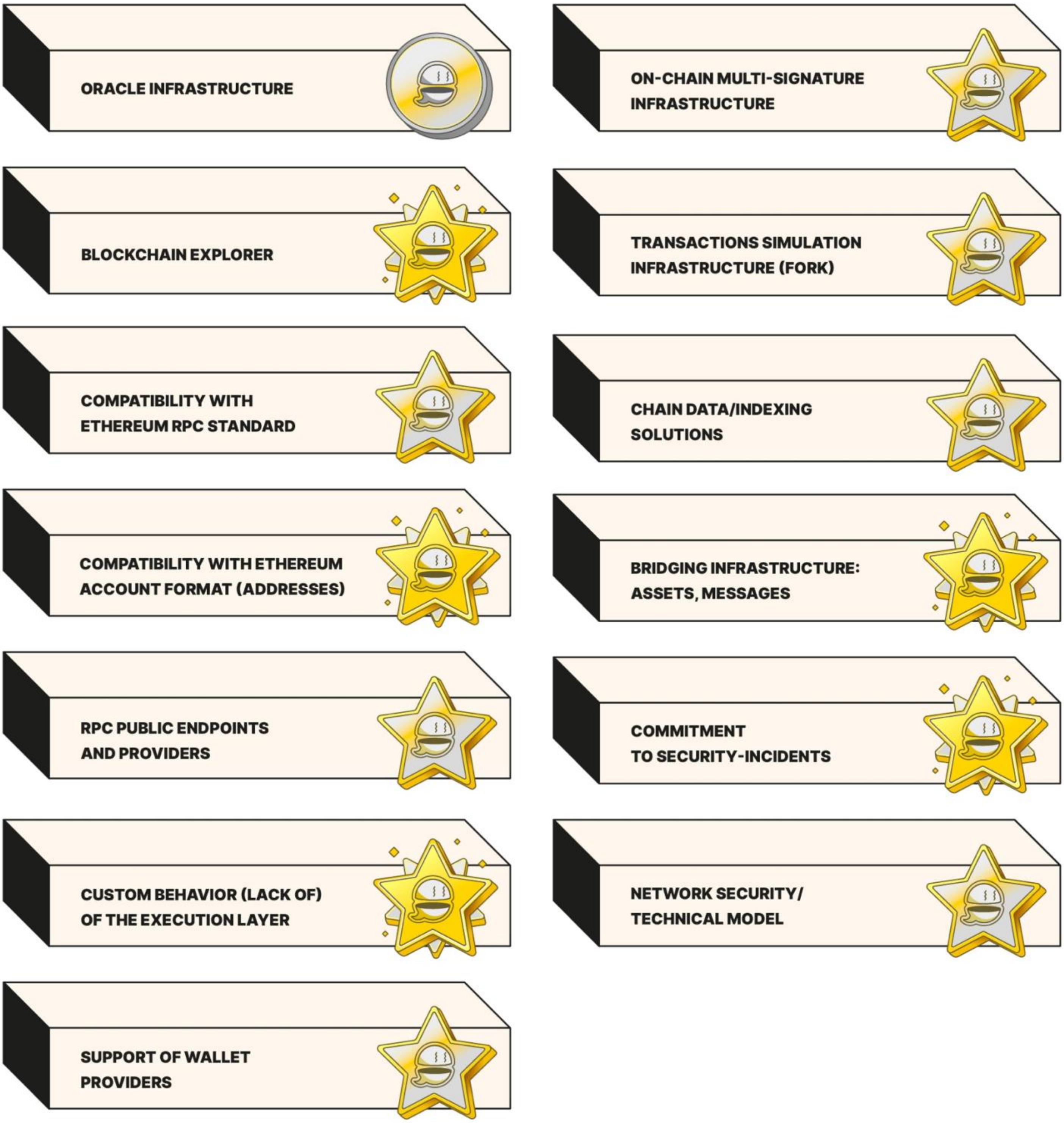
### 3.13.5. Network upgrade procedures

Whenever a network upgrade or any type of patch is applied to Polygon zkEVM, there is an extensive multi-step procedure involving the following:

- Different process-specific roles are defined, in order to maximize coordination and efficiency.
- Both security researchers and engineers participate.
- Audits are done in different steps, by internal and in some cases external parties.
- Updates of the different open-sourced repositories, to avoid information asymmetry.
- There is a time lock for the application of changes in the network.

It is worth to mention that the network has gone through important upgrades during this initial period, consequently causing some meaningful downtimes. This should be taken into account by risk providers when recommending the initial listing parameters.

# Summary



ORACLE INFRASTRUCTURE

BLOCKCHAIN EXPLORER

COMPATIBILITY WITH ETHEREUM RPC STANDARD

COMPATIBILITY WITH ETHEREUM ACCOUNT FORMAT (ADDRESSES)

RPC PUBLIC ENDPOINTS AND PROVIDERS

CUSTOM BEHAVIOR (LACK OF) OF THE EXECUTION LAYER

SUPPORT OF WALLET PROVIDERS

ON-CHAIN MULTI-SIGNATURE INFRASTRUCTURE

TRANSACTIONS SIMULATION INFRASTRUCTURE (FORK)

CHAIN DATA/INDEXING SOLUTIONS

BRIDGING INFRASTRUCTURE: ASSETS, MESSAGES

COMMITMENT TO SECURITY-INCIDENTS

NETWORK SECURITY/ TECHNICAL MODEL

**From our analysis, we conclude that Polygon zkEVM, even if in a pretty early stage, is an acceptable network candidate in regard to technical requirements, with no current hard blocker for the Aave v3 protocol to work properly.**

However, we think the community should start with conservative caps during a warm-up period of 1 months, in order to not grow liquidity too fast.

An expansion of Aave there will imply allocating some development resources for both the initial setup, together with some overhead of maintenance and monitoring over time, similar to other networks.

Same as with other rollups, there is an important degree of centralization, but this is expected given the early stage of this technology. However, the validity-rollup nature of zkEVM is a pretty strong aspect to consider.