BGD. AAVE <> Scroll.

INFRASTRUCTURE / TECHNICAL EVALUATION

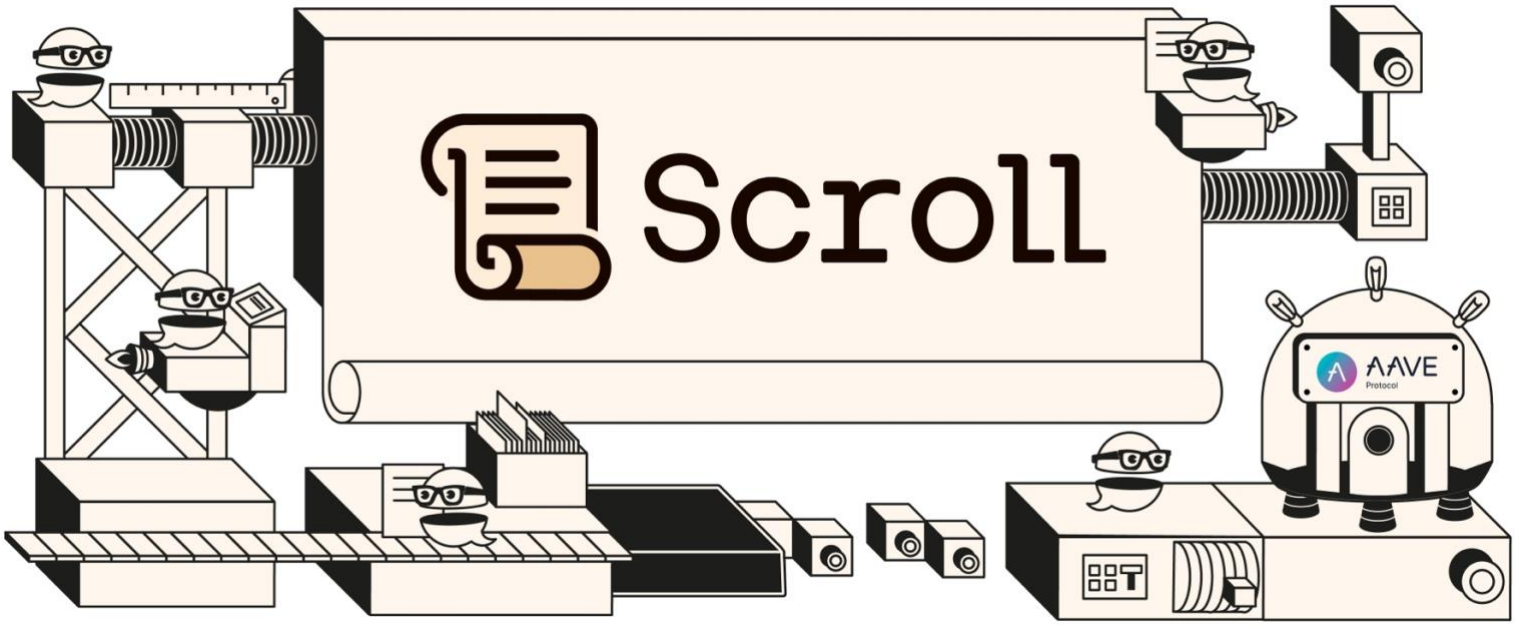


Table of Contents

Introduction

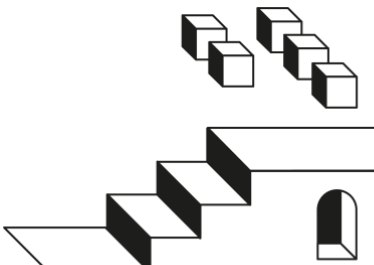
Following our framework of Aave's infrastructure evaluation and the positive outcome of the temperature check, we present the **analysis of Scroll regarding its suitability to deploy an instance of the Aave protocol**.

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 Scroll ecosystem.

When doing the evaluation, we contacted the Scroll 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 Scroll

Scroll is a Layer 2 ZK rollup, or more precisely, as defined in the documentation, "*Scroll is a general-purpose ZK rollup that uses the EVM for off-chain computations".*

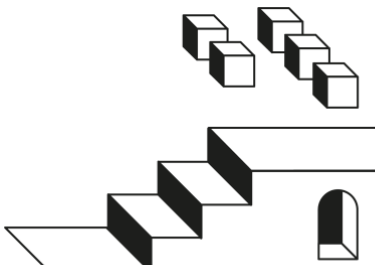
Scroll was released on mainnet on October 8th, 2023.

2. Our methodology

This report is not trying to be a full analysis of the Scroll 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.

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

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.

Scroll **DOESN’T HAVE** Chainlink L2 Sequencer Uptime, but is working with Chainlink to provide one in the future.

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 Scroll network can be found at https://scrollscan.com/, and it is a white-labeled instance of Etherscan.

Additionally, there's an instance of blockscout, dora, and l2scan.

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 Scroll in this case.

Scroll **HAS** complete compatibility with the standard, including some extra custom endpoints, and lacks some non-standard. It's important to note that Scroll uses zktrie for state storage, so state proofs returned by eth_getProof will not be compatible with existing tooling.

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.

Scroll **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.

Scroll has available the following public RPC endpoints:

- https://rpc.scroll.io/

Additionally, Node-as-a-Service providers like Ankr, 1rpc are available. Major NaaS providers like Alchemy and Infura are missing.

There **IS** support by POKT.

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

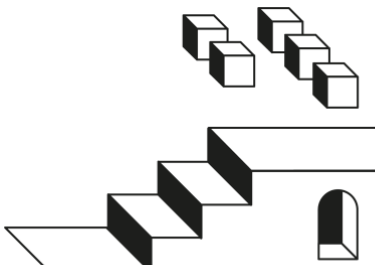
Whenever a network has custom/extended behavior with respect to Ethereum, it is important to be aware of it and evaluate if it has any impact on the Aave protocol.

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 Scroll-zkEVM implementation, we have checked the following, given that historically have been critical aspects:

- The chainId behavior is appropriate, with the id 534352 for Scroll not clashing with any other.
- Scroll has equivalence (address and logic) with the ecRecover and identity Ethereum pre-compiles, which covers the needs of Aave.
- Scroll-zkEVM has rough equivalence with Ethereum london. Instead of being fully London equivalent, some London aips are disabled [EIP-1559, EIP-2930, EIP-3198] and some Shanghai aips are introduced [EIP-3651, EIP-3855, EIP-3860] for compatibility.
- Some opcodes (BLOCKHASH, COINBASE, DIFFICULTY, BASEFEE, SELFDESTRUCT) and precompiles (SHA-256, RIPEMD-160, blake2f, modexp, ecPairing) behave differently from L1. See https://docs.scroll.io/en/developers/ethereum-and-scroll-differences/#evm-opcodes
- Deep reorgs are very unlikely but not impossible. L2 blocks are only considered finalized (irreversible) when they have been committed to L1 and finalized through a zk proof.
- Fixed block time is not guaranteed on Scroll.

Additionally, we have a confirmation from the Scroll 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, 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, Scroll is transparently supported by the majority of all the chain-agnostic wallets, like Metamask, Ledger, or Frame.

It is possible that other types of wallets (e.g. based on smart contracts) don’t support Scroll, 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.

Scroll **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://safe.scroll.xyz/welcome.

The Scroll team has confirmed the maintainer team 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 Scroll with Foundry. Given its EVM compatibility, it should be perfectly doable with Hardhat too.

Currently, neither Tenderly nor Phalcon is integrating Scroll.

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.

Scroll is **not** supported on TheGraph (neither hosted nor decentralized).

Scroll is **not** supported by Dune.

Scroll is partially supported on Covalent.

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 Scroll, the state of these 2 types is as follows:

- **Assets.** sub-use case the generic messaging infrastructure supported by Scroll. For end users, it is possible to bridge assets via https://scroll.io/bridge.

  Regarding the security of the ERC20 smart contracts for bridged assets, we have checked the main tokens share the same implementation: an upgradable ERC20 with permit based on the OZ version, with burn/mint capabilities by an entity defined as "gateway".

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

In addition to the default bridging mechanism of Scroll, 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 Scroll are no exception.

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

- On the prevention side, Scroll has an Immunefi bug bounty campaign running.
- The Scroll Team runs a chain-monitor component which will automatically short-circuit finalization if any anomalies are detected.
- If any incident happens, Scroll Team 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.
- A private channel of communication will be kept between the Scroll team and the assigned technical team of the Aave community (e.g. BGD), for any necessary update concerning the network and consequently, Aave on Scroll. During our evaluation, the team has always been 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).

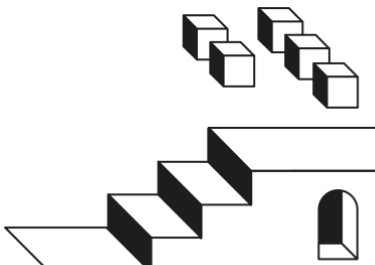
Regarding its morphology/type, Scroll is a zk roll-up, meaning that it leverages zk-knowledge technology (a combination of SNARK and STARK) to provide high scalability for blockchain transactions.

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

3.13.1. Transactions flow

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

1. A user interacting directly with Scroll 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 are also able to submit transactions to the Scroll bridge contract on Ethereum (batches), which will introduced in the Pool.*
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 the 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 Relayer component checks for and takes the batch proof from the stateDB, and submits a finalize Transaction to the rollup contract.

3.13.2. Data availability

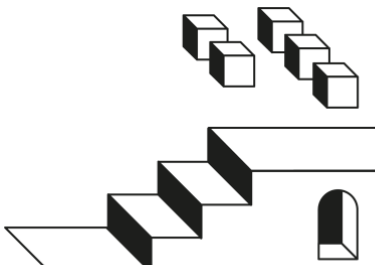
On Scroll, all the transaction data is submitted to Ethereum, so data availability boils down to Ethereum, which can be considered the highest standard at the moment.

3.13.3. Upgradeability and control model

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

Also, most proxies and verifiers can be upgraded by the ScrollMultisig (4/5)- in an emergency the SecurityCouncilMultisig can upgrade contracts without any timelock. You can find a breakdown of permissions HERE.

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



3.13.4. Security audits

The security audits of Scroll can be found HERE.

3.13.5. Network upgrade procedures

Contracts:

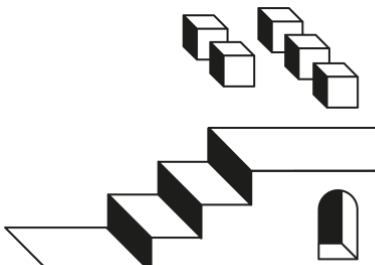
- Scheduled contract upgrades (new features, optimizations, etc.) are initiated by ScrollMultisig and are behind a 14-day timelock. We will publish these upgrades on our blog, and on social media, and will sync with affected partners individually.
- In case there is a vulnerability, the SecurityCouncilMultisig can upgrade the contracts with no time delay.

Sequencer:

- New releases will be published here: https://github.com/scroll-tech/go-ethereum/releases
- Node-running instructions are published here: https://bit.ly/scroll-l2geth
- Usually upgrades to a new version take just a few seconds. Any longer maintenance window will be announced on https://status.scroll.io/ a few days in advance.

Provers:

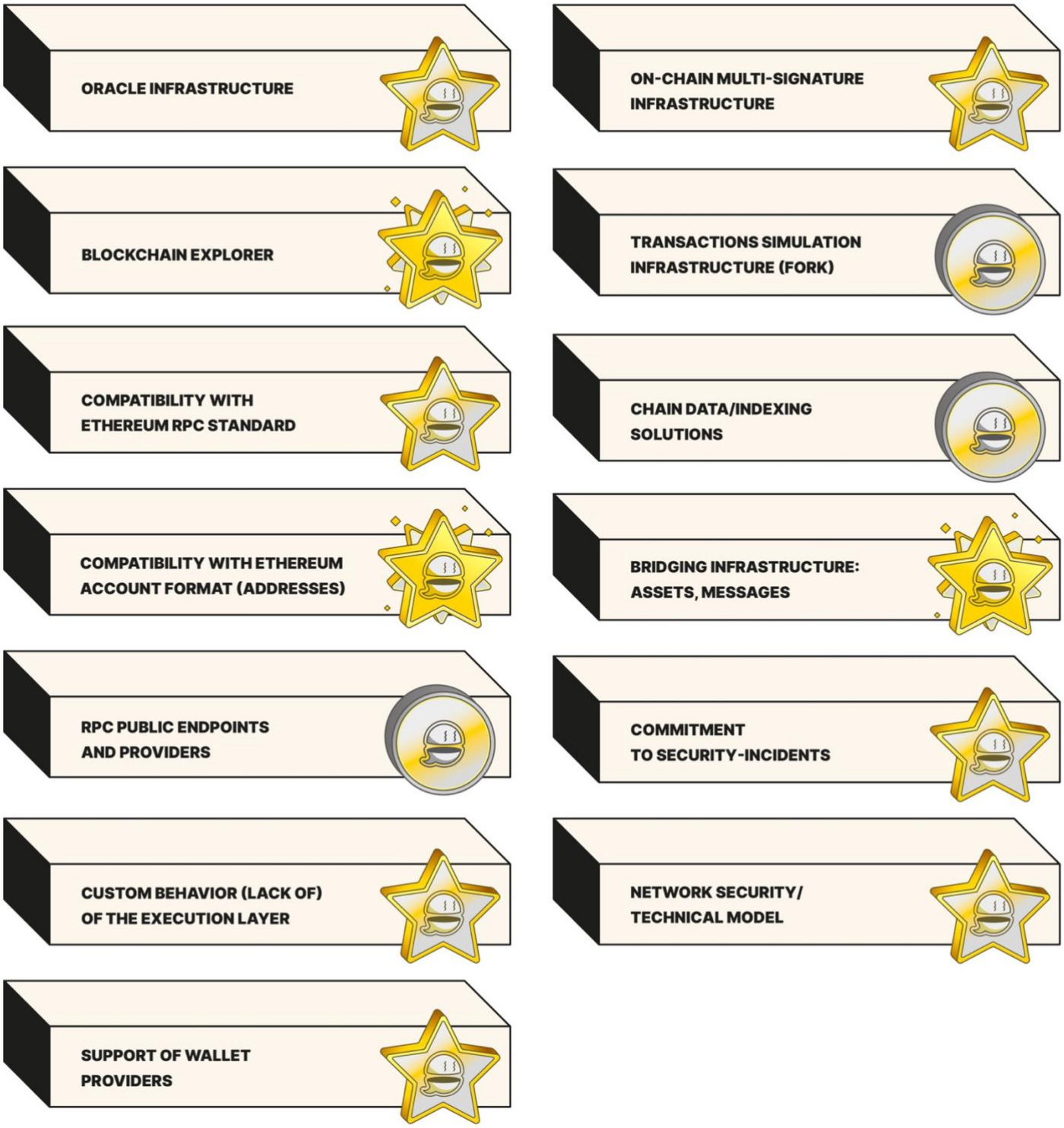
- Prover upgrades are mostly transparent to users, but they might result in increased finalization delay.
- Scroll team will announce planned prover maintenance windows on https://status.scroll.io/.



Future strategy of alignment with Scroll/Ethereum upgrades:

- Scroll team will evaluate this on a case-by-case basis, considering usefulness, feasibility, and effort.
- They aim to adopt any new opcodes and precompiles when it makes sense. They have already adopted PUSH0 and aim to support transient storage opcodes (TLOAD, TSTORE).
- They might adopt any rollup standards defined through the "RollCall/RIP" coordination mechanism.

Summary


ORACLE INFRASTRUCTURE
ON-CHAIN MULTI-SIGNATURE INFRASTRUCTURE
BLOCKCHAIN EXPLORER
TRANSACTIONS SIMULATION INFRASTRUCTURE (FORK)
COMPATIBILITY WITH ETHEREUM RPC STANDARD
CHAIN DATA/INDEXING SOLUTIONS
COMPATIBILITY WITH ETHEREUM ACCOUNT FORMAT (ADDRESSES)
BRIDGING INFRASTRUCTURE: ASSETS, MESSAGES
RPC PUBLIC ENDPOINTS AND PROVIDERS
COMMITMENT TO SECURITY-INCIDENTS
CUSTOM BEHAVIOR (LACK OF) OF THE EXECUTION LAYER
NETWORK SECURITY/ TECHNICAL MODEL
SUPPORT OF WALLET PROVIDERS

From our analysis, we conclude that Scroll 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 Scroll is a pretty strong aspect to consider.