

Kósmos: The Trust Protocol - Project Blueprint

1. Executive Summary and Vision

Project Name: Kósmos: The Trust Protocol (Kósmos means 'order' or 'universe' in Greek, reflecting an ordered system of trust.)

Elevator Pitch: Kósmos is a privacy-centric, on-chain reputation and identity network built on Stellar and Soroban. It leverages Decentralized Identity (DID), Verifiable Credentials (VCs), and Zero-Knowledge Proofs (ZKPs) to establish a flexible, non-financial trust graph, enabling users to share verifiable claims (like credit scores or work history) without ever exposing the underlying data.

Core Vision: To create a global, non-financial trust layer on the Stellar network that is flexible, user-centric, and entirely privacy-preserving.

Category	Current Problem	Kósmos Solution
Identity	Centralized identity silos (credit bureaus, employer databases).	Self-Sovereign Identity (SSI) anchored by DIDs and VCs.
Privacy	Sharing verifiable data (like credit score) requires revealing the full data.	Zero-Knowledge Proofs (ZKPs) allow proving <i>possession</i> of a score <i>range</i> without revealing the score itself.
Trust	Trust is limited to financial stake (PoS) or computation (PoW).	Flexible, reputational trust graph based on the Stellar Consensus Protocol (SCP).

2. Foundational Pillars and Technology Stack

Kósmos is built upon three main technological pillars:

Pillar A: The Trust Layer (Stellar Consensus Protocol - SCP)

- **Role in Kósmos:** SCP enables the creation of a *reputation graph* distinct from the transaction ledger. Nodes (or specific smart contracts on Soroban) can define Quorum Slices based on which Identity Issuers (e.g., banks, universities) they trust.
- **Key Feature: Flexible Trust:** Unlike Proof-of-Work or Proof-of-Stake, SCP's Federated Byzantine Agreement (FBA) allows entities to define their own trust dependencies (Quorum Slices).
 - *Implementation Detail:* A Soroban Smart Contract will track the public key hashes of verified issuers. Validators on the Stellar network can include these Issuer/Reputation contracts in their Quorum Slices, implicitly validating the reputation framework.
- **Reputation Score:** A custom Kósmos Reputation Score (KRS) is derived from the verified VCs and the reputation of their issuers within the SCP trust network. This score is abstract and designed to be used as input for ZKPs.

Pillar B: Self-Sovereign Identity (SSI)

- **Technology Used:** W3C Decentralized Identifiers (DIDs) and Verifiable Credentials (VCs).
- **DID Implementation on Soroban:**
 - **Method:** A custom did:kosmos: method will be implemented as a Soroban smart contract.
 - **DID Document:** The DID Document will be anchored on Soroban, mapping the user's DID to their public keys, verification methods, and service endpoints (e.g., ZKP generation service).
 - **User Control:** Users retain the private key (off-chain) to control their DID, ensuring self-sovereignty.
- **Verifiable Credentials (VCs):**
 - **Issuance:** Trusted third parties (e.g., credit bureaus, employers, Kósmos network validators) act as Issuers, creating VCs (JSON-LD documents) cryptographically signed with their private keys.
 - **Storage:** VCs are stored securely in the user's personal digital wallet (off-chain, encrypted), *not* on the blockchain.
 - **Examples of VCs:** VC-Credit-Score, VC-Employment-History, VC-KYC-Verified.

Pillar C: Privacy via Zero-Knowledge Proofs (ZKPs)

- **Role:** ZKPs are the mechanism that allows the user (Prover) to prove a statement about their private VCs to a Verifier (e.g., a lender) without revealing the underlying data.
- **Implementation:** Using a non-interactive ZKP scheme (like zk-SNARKs or zk-STARKs) to prove specific predicates on the VC data.
 - **Credit Score Example:** A user has a VC-Credit-Score with an attribute score: 750.
 - **ZKP Statement (Proof):** The Prover (user) generates a proof that "I have a valid VC from a trusted Issuer, and the value of score in that VC is greater than 700."
 - **Verification on Soroban:** The Verifier submits the ZKP to a dedicated Soroban Smart Contract (the ZKP Verifier Contract). This contract executes the verification logic (the circuit) on-chain. If the proof is valid, the contract emits an event or sets a state variable confirming the predicate (Score_is_High: true), completing the transaction without ever knowing the score of 750.

3. Architecture and Soroban Integration

Component	Stellar/Soroban Integration	Function
DID Registry	Soroban Smart Contract (Rust)	Stores the DID Document hashes, public keys, and revocation status for all did:kosmos: identifiers.
Issuer Whitelist	Soroban Smart Contract (Rust)	Maintains a dynamic list of recognized and vetted VC issuers, whose reputation is tied to the SCP network.

Component	Stellar/Soroban Integration	Function
ZKP Verifier	Soroban Smart Contract (Rust)	Executes the cryptographic verification function for submitted Zero-Knowledge Proofs (e.g., verifying a proof that a score is within a range).
KRS Token/Asset	Stellar Native Asset / Soroban Token	An optional, non-transferable soul-bound token (SBT) representing the user's derived reputation within the ecosystem, used for governance or tiered service access.
Client Wallet	Off-Chain App	Securely stores the user's private keys, DIDs, and VCs. Generates ZKPs when requested by a Verifier.

4. Use Case Deep Dive: Privacy-Preserving Lending

Goal: Allow users to qualify for a loan based on their credit score *without* revealing the score itself to the lender.

1. **VC Issuance:** A credit bureau (Trusted Issuer) generates a VC-Credit-Score containing the user's private credit score (Score: 780) and signs it. The user receives this VC in their Kósmos Wallet.
2. **Lender Request (Verifier):** A DeFi lender on Soroban requires a loan applicant to prove they have a score over 700. The lender's smart contract sends a ZKP request to the user's wallet.
3. **Proof Generation (Prover):** The user's Kósmos Wallet software constructs a Zero-Knowledge Proof (ZKP) based on the secret statement: Credit_Score > 700. The wallet uses the signed VC as the input "witness" to generate the proof.
4. **On-Chain Verification:** The user submits the ZKP (a small, public cryptographic proof) via a Stellar transaction that calls the **ZKP Verifier Smart Contract** on Soroban.
5. **Smart Contract Outcome:** The Verifier Contract executes the proof checking function. It verifies:
 - The proof is mathematically sound.
 - The underlying VC was signed by a whitelisted Issuer.
 - The public statement (Score > 700) is confirmed as TRUE.
 - The contract's state is updated to Loan_Qualified: true.
6. **Result:** The loan application proceeds based on verified eligibility, and the lender *never* saw the private score of 780.

5. Market Potential and Business Model

Target Market

1. **DeFi/Lending Platforms:** Soroban-based lending protocols needing private, reliable

- eligibility checks.
2. **Global Payments/Remittance:** Providing verifiable KYC/AML status without sharing ID documents for cross-border transactions (a core strength of Stellar).
 3. **Decentralized HR/Employment:** Proving work history, certifications, or academic degrees to employers without sharing the full documents.

Business Model (Kósmos Protocol)

Stream	Mechanism	Description
Proof-as-a-Service (PaaS)	Transaction Fees (XLM/KRS)	Small fee charged for every ZKP submitted to the ZKP Verifier Smart Contract on Soroban. This is the primary revenue source.
Issuer Certification	Annual Licensing Fee	Fees charged to institutions (banks, credit bureaus, universities) to be included and maintained on the Issuer Whitelist Contract .
Premium API Access	Subscription/Tiered Access	Offering dedicated SDKs and resolution services for enterprise verifiers and wallets.
Staking/Bonding	Utility Token (KRS)	Requiring Issuers and certain Verifiers to stake the KRS token to prove commitment and integrity. Slashing mechanism for malicious behavior.

6. Implementation & Roadmap

Phase 1: Foundation (6 Months)

- **Smart Contract Development:** Launch DID Registry and Issuer Whitelist contracts on Soroban Testnet.
- **ZKP Core Integration:** Select and integrate a non-interactive ZKP library (e.g., built-in Rust/Wasm capability or established library).
- **Wallet Prototype:** Develop a functional mobile wallet prototype (off-chain) to store VCs and generate ZKPs.

Phase 2: Pilot & MVP (6 Months)

- **MVP Launch:** Deploy full stack on Soroban Mainnet.
- **Pilot Program:** Onboard 1-3 anchor institutions (e.g., a micro-lender, a university) as initial Trusted Issuers.
- **SCP Integration:** Secure public validation from key Stellar ecosystem validators for the Kósmos reputation model.

Phase 3: Scaling & Governance (Ongoing)

- **Governance:** Transition the Issuer Whitelist and protocol upgrade decisions to a decentralized governance model (DAO).
- **Network Effect:** Expand globally, focusing on emerging markets where traditional credit systems are inadequate or non-existent, leveraging Stellar's low-cost infrastructure.
- **Advanced ZKP:** Implement more complex ZKP circuits (e.g., proving identity attributes across multiple VCs).