A BLOCKCHAIN AND ZERO TRUST FRAMEWORK FOR SECURE ROAMING IN 5G NETWORKS

Presented by:

Rezazi Mohamed Abdessamed

In front of the juries:

Dr. MENACER D.

Dr. DAHMANI R.

Dr. KHENFOUCI Y.

Supervised by:

Dr Benabidallah R. (ESI)

Pr Ghamri Y. (L3I)

Dr Bouchiha A. (L3I)

Mr Bendada M. (L3I)

21 September, 2025

Problem Statement

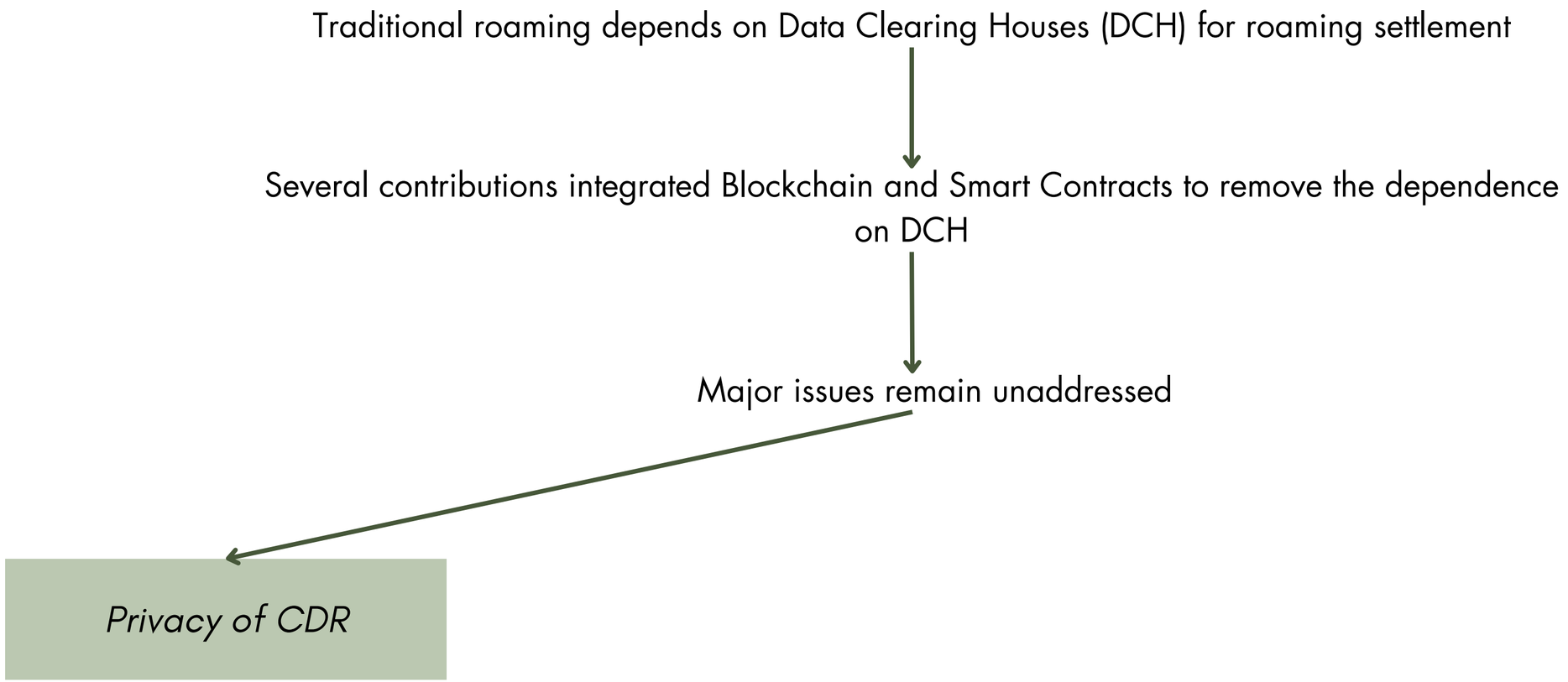
Traditional roaming depends on Data Clearing Houses (DCH) for roaming settlement

Problem Statement

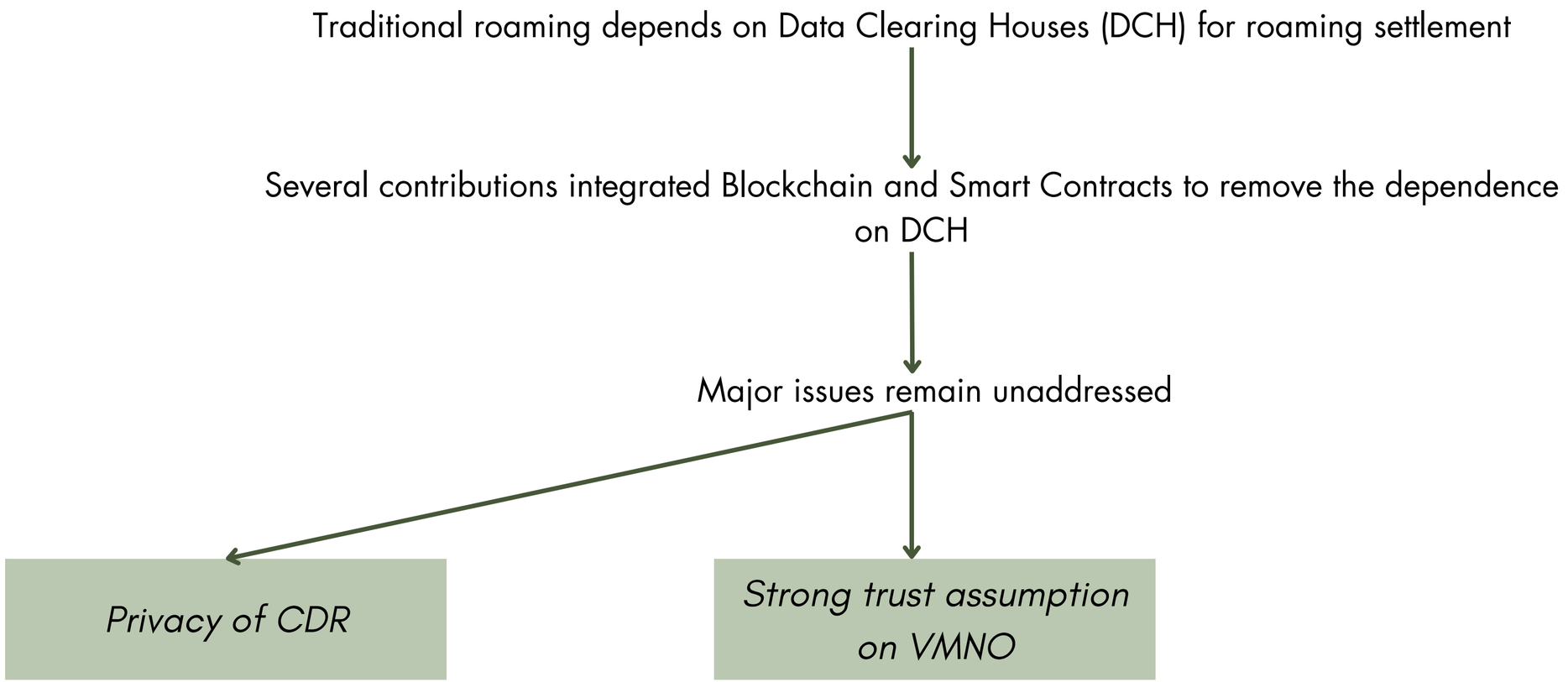
Traditional roaming depends on Data Clearing Houses (DCH) for roaming settlement

Several contributions integrated Blockchain and Smart Contracts to remove the dependence on DCH

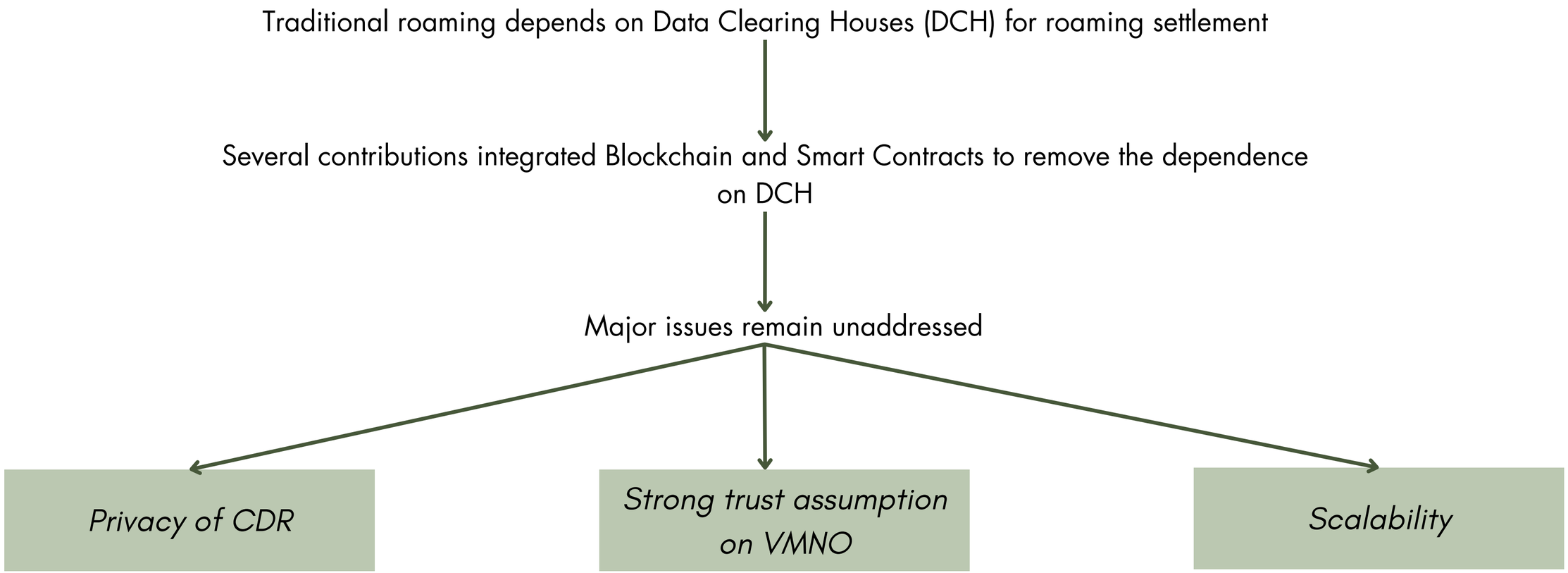
Problem Statement



Problem Statement



Problem Statement



Research Question

How to design a Blockchain-based 5G roaming settlement system that is scalable, trustless, and privacy-preserving?

Agenda Overview

01 Solution Design

02 Framework Modules

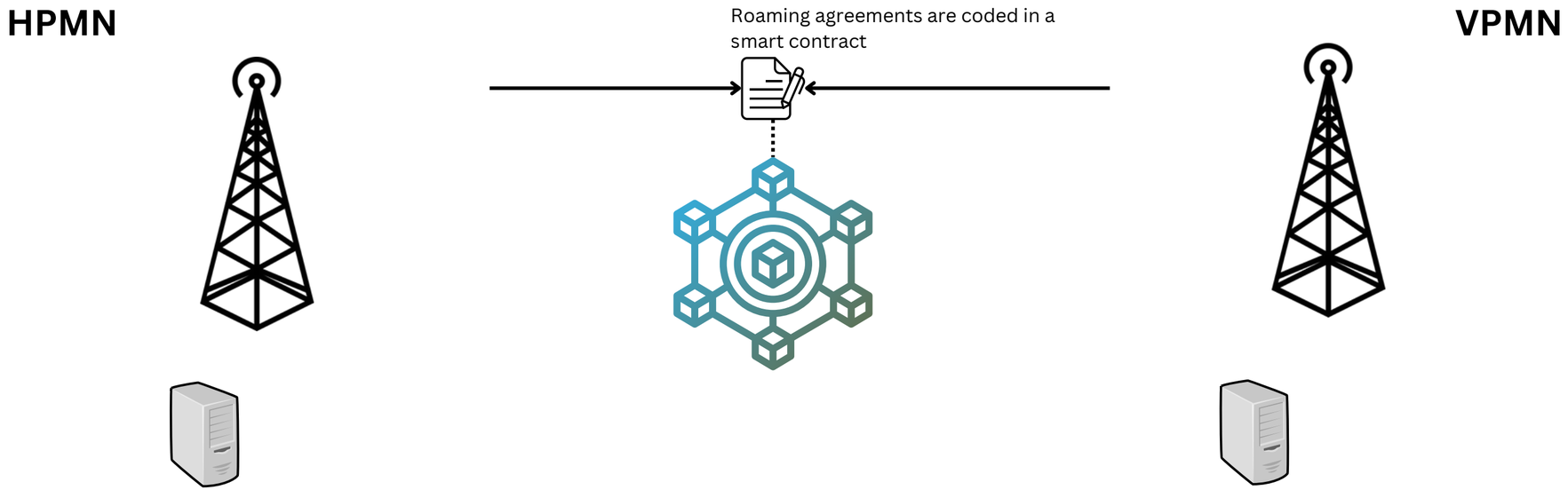
03 Tests & Results

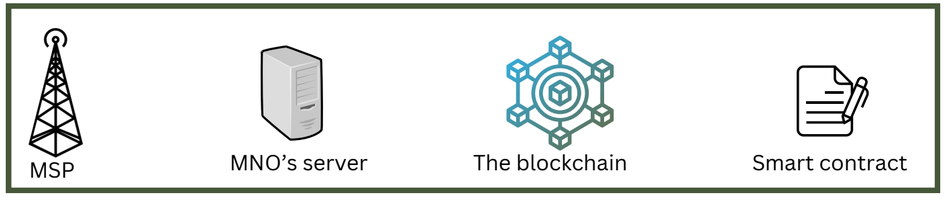
04 Perspectives

05 Conclusion

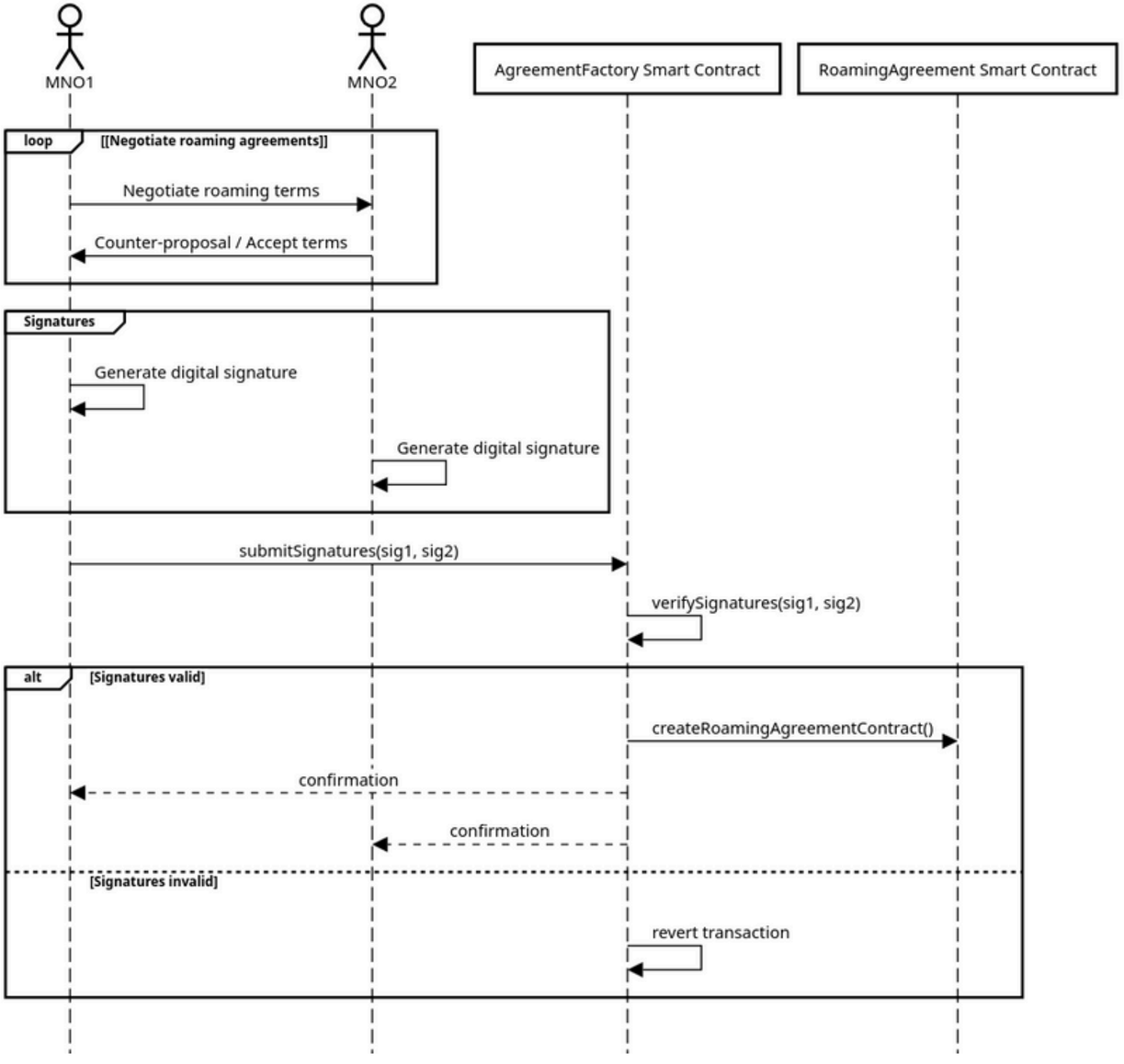
06 Demo

Solution Design

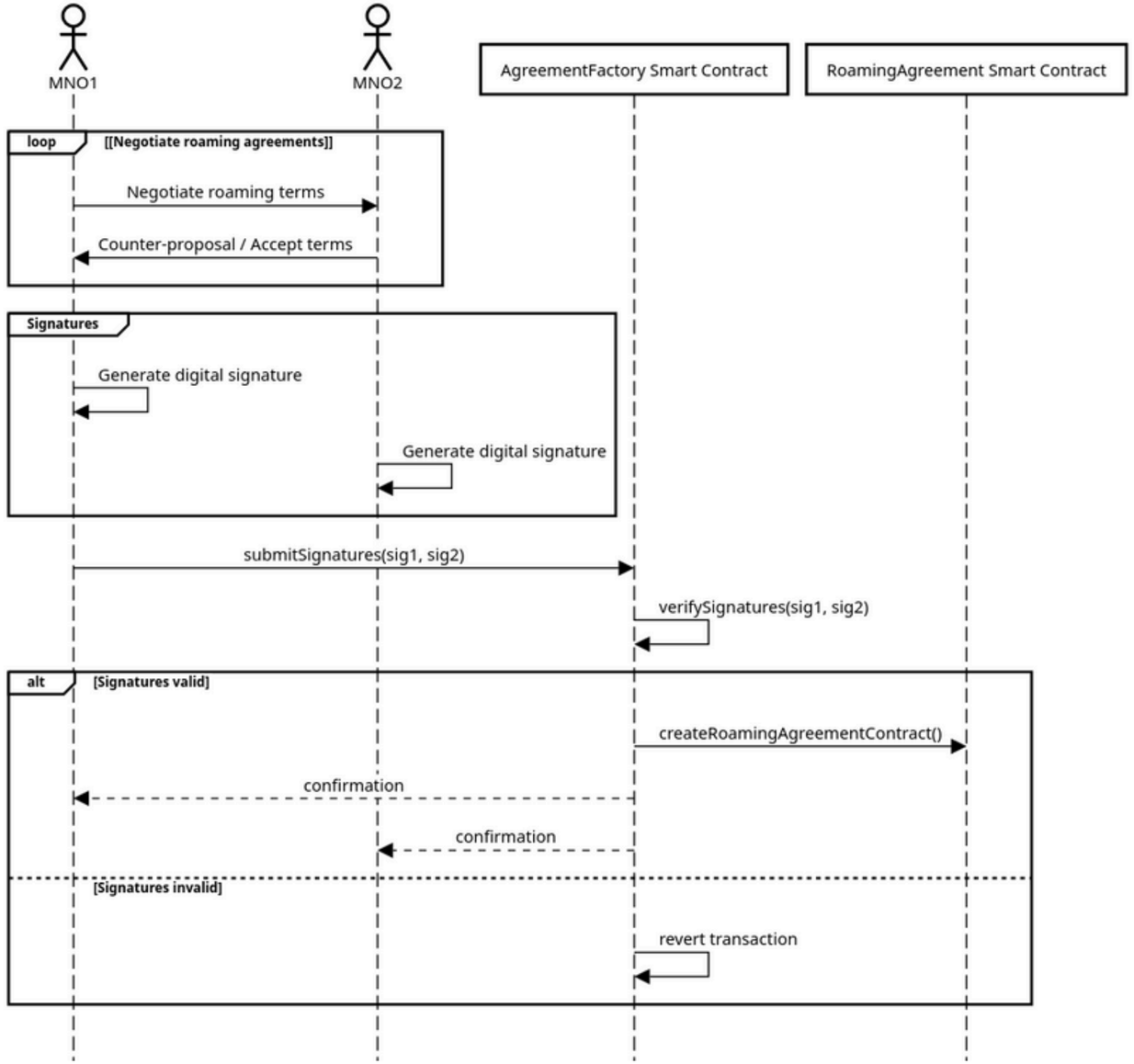




Solution Design



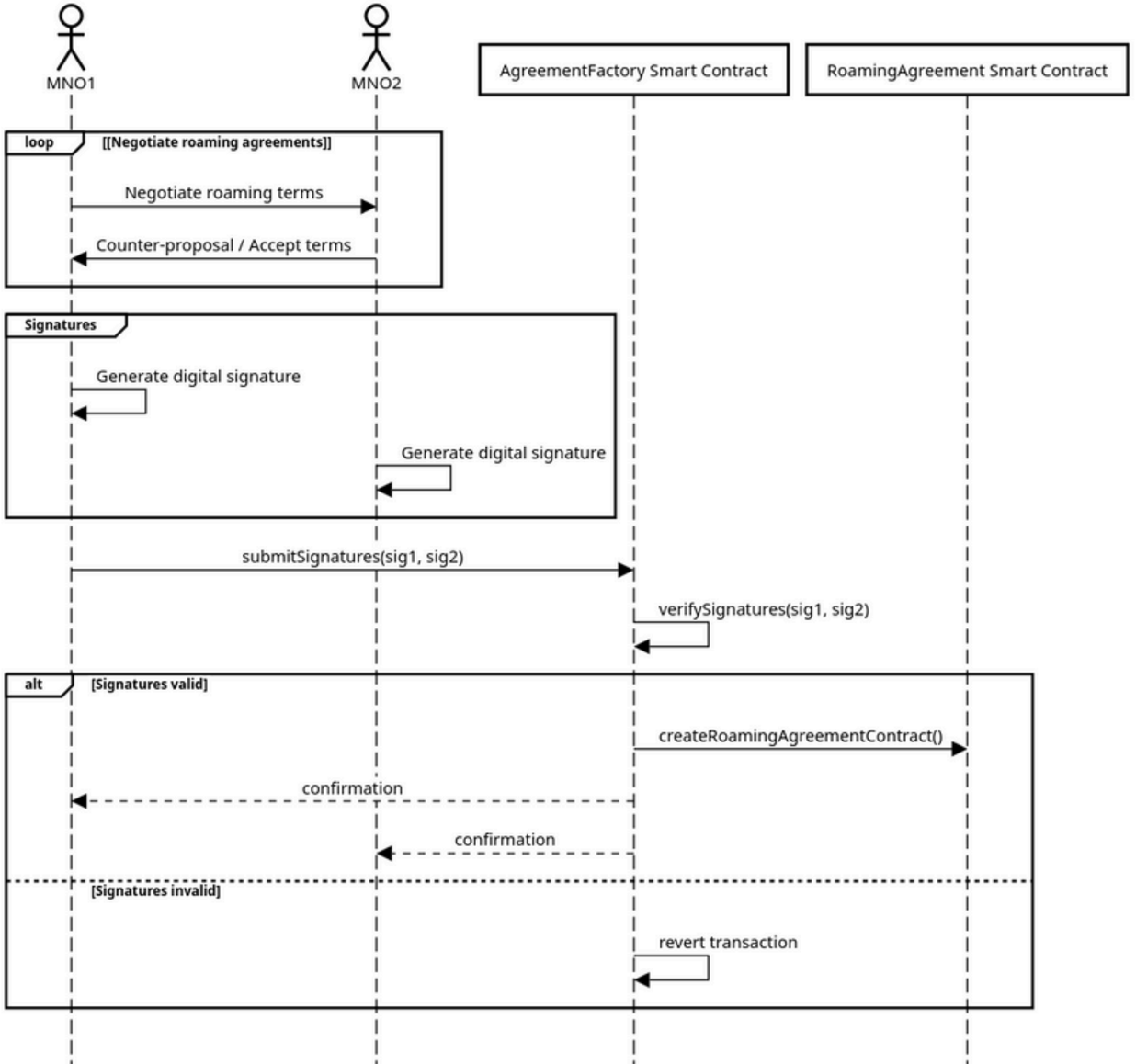
Solution Design



The roaming agreement encodes the following essential parameters

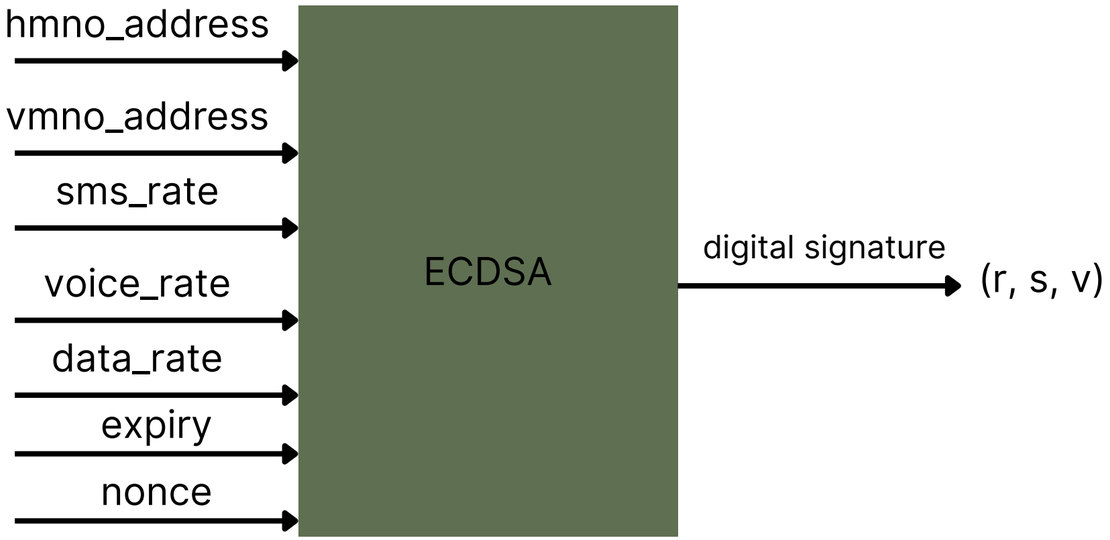
- Service rates
- HMNO and VMNO identifiers
- Agreement metadata

Solution Design

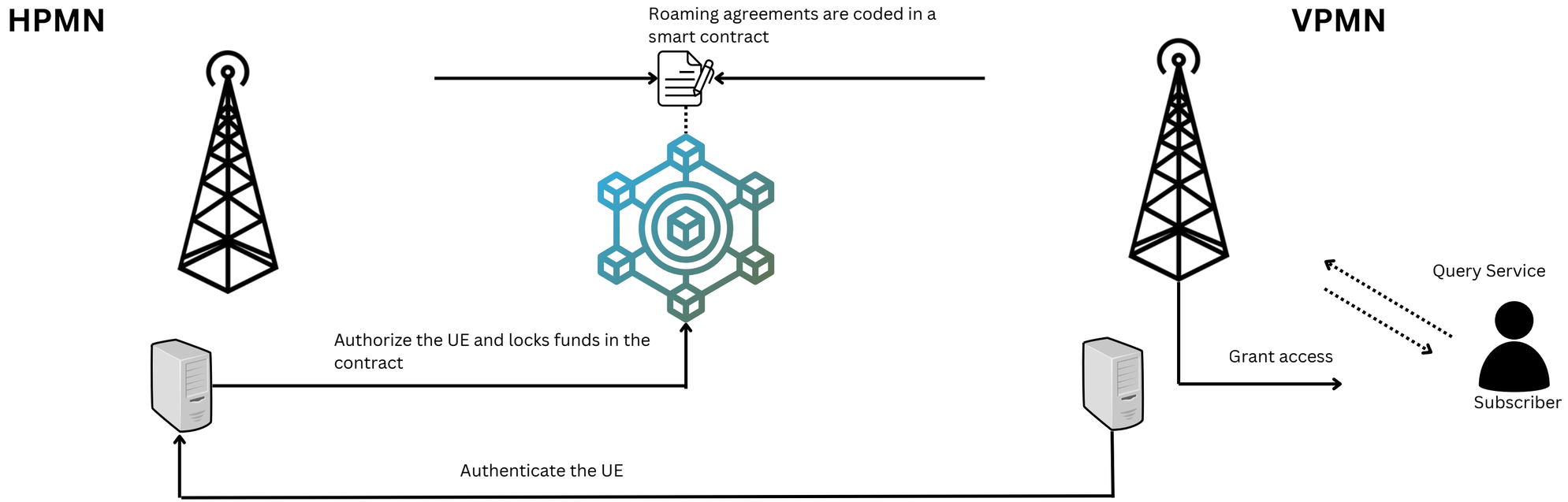


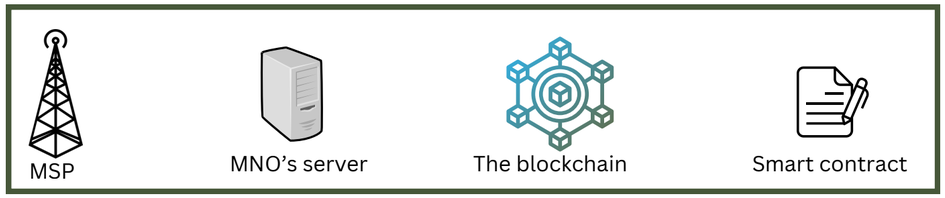
The roaming agreement encodes the following essential parameters

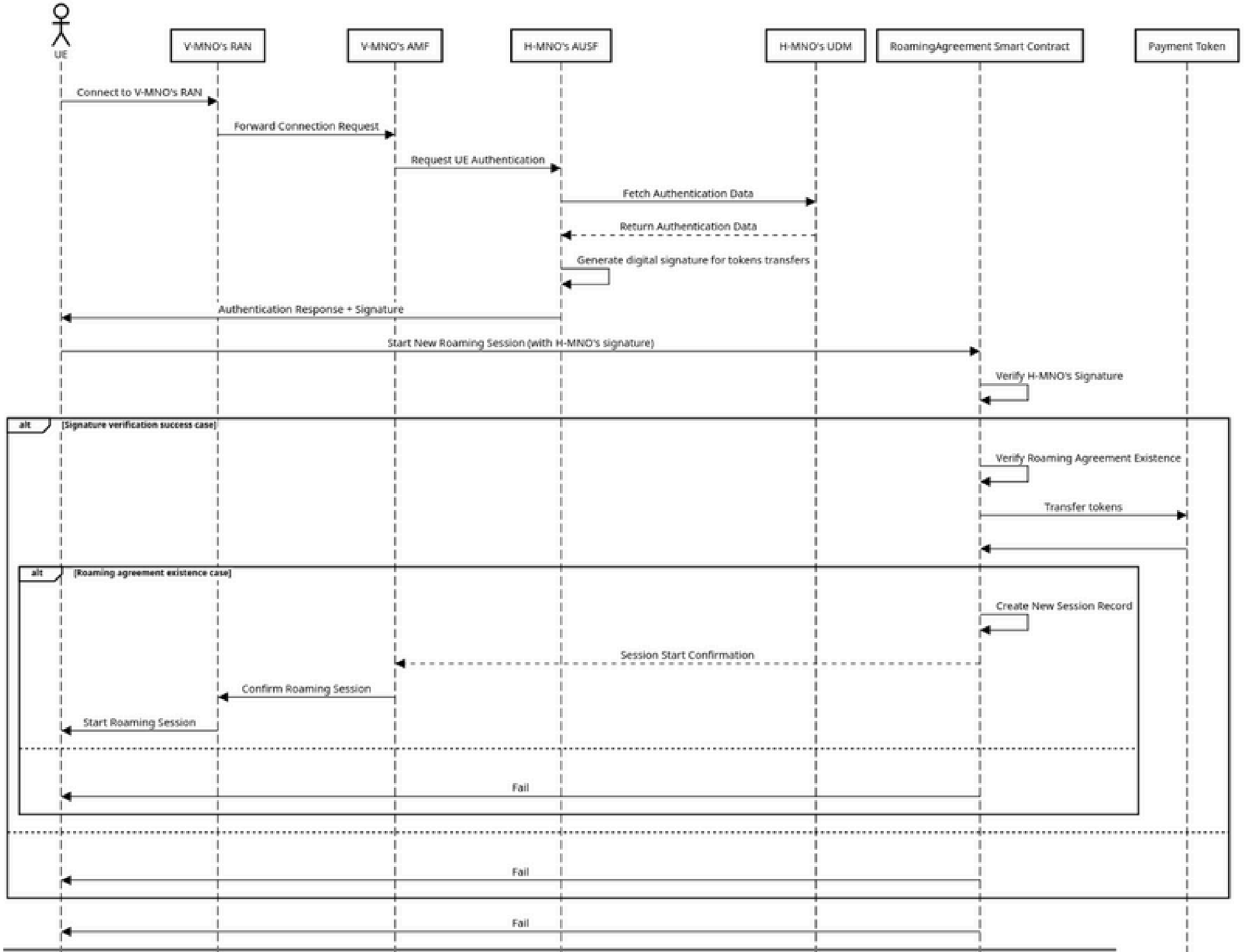
- Service rates
- HMNO and VMNO identifiers
- Agreement metadata



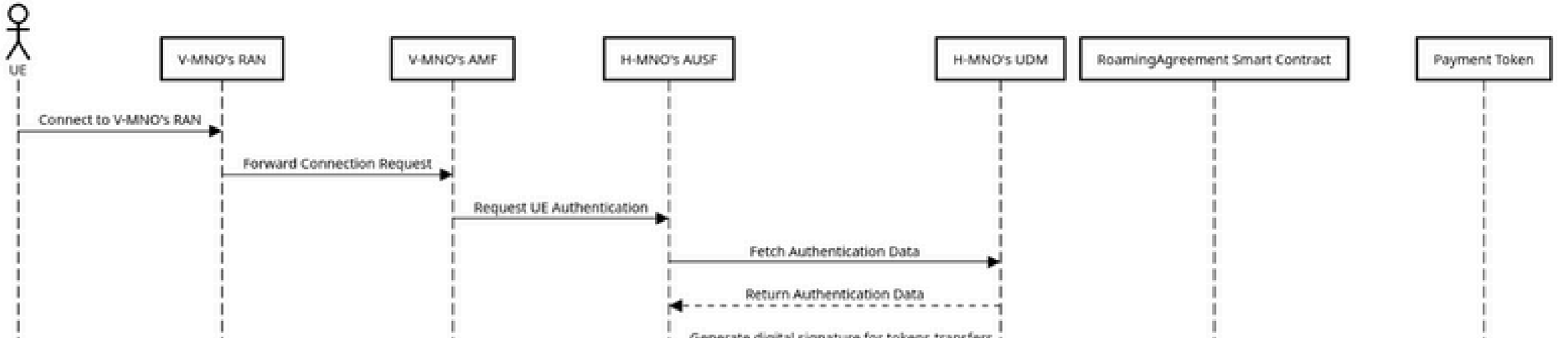
Solution Design



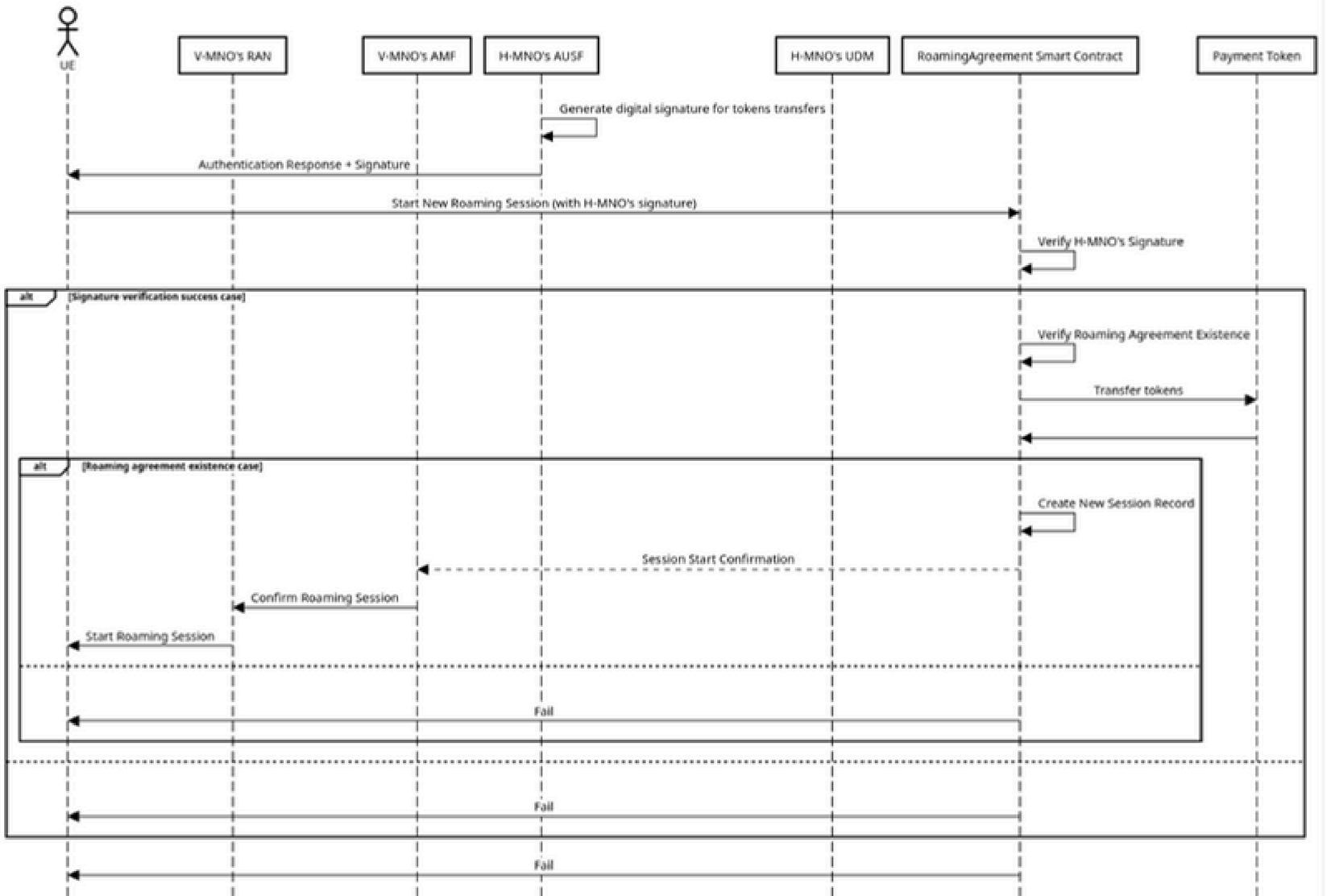



UE
V-MNO's RAN
V-MNO's AMF
H-MNO's AUSF
H-MNO's UDM
RoamingAgreement Smart Contract
Payment Token
Connect to V-MNO's RAN
Forward Connection Request
Request UE Authentication
Fetch Authentication Data
Return Authentication Data
Generate digital signature for tokens transfers
Authentication Response + Signature
Start New Roaming Session (with H-MNO's signature)
Verify H-MNO's Signature
alt
[Signature verification success case]
Verify Roaming Agreement Existence
Transfer tokens
alt
[Roaming agreement existence case]
Create New Session Record
Session Start Confirmation
Confirm Roaming Session
Start Roaming Session
Fail
Fail
Fail

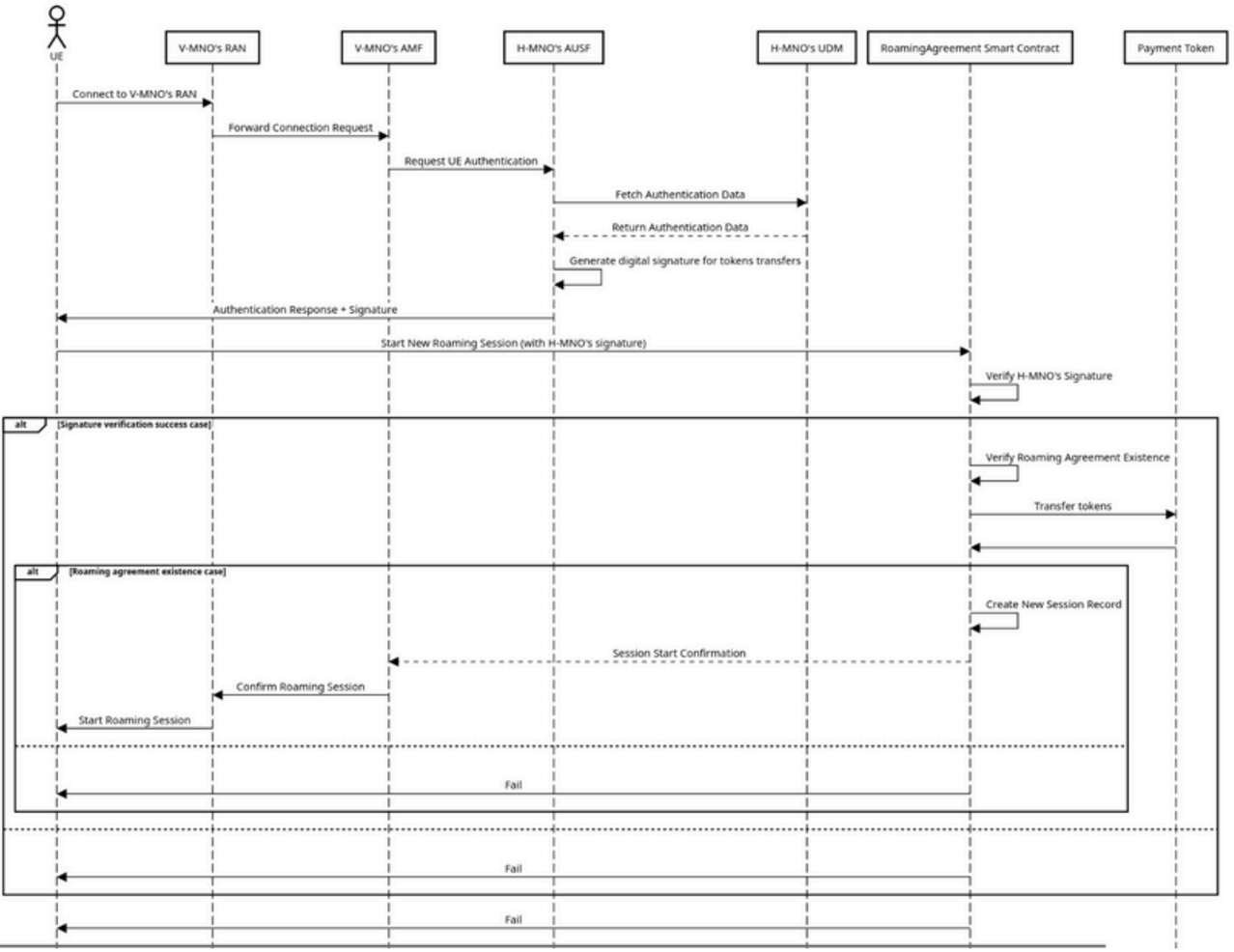
Solution Design



Solution Design



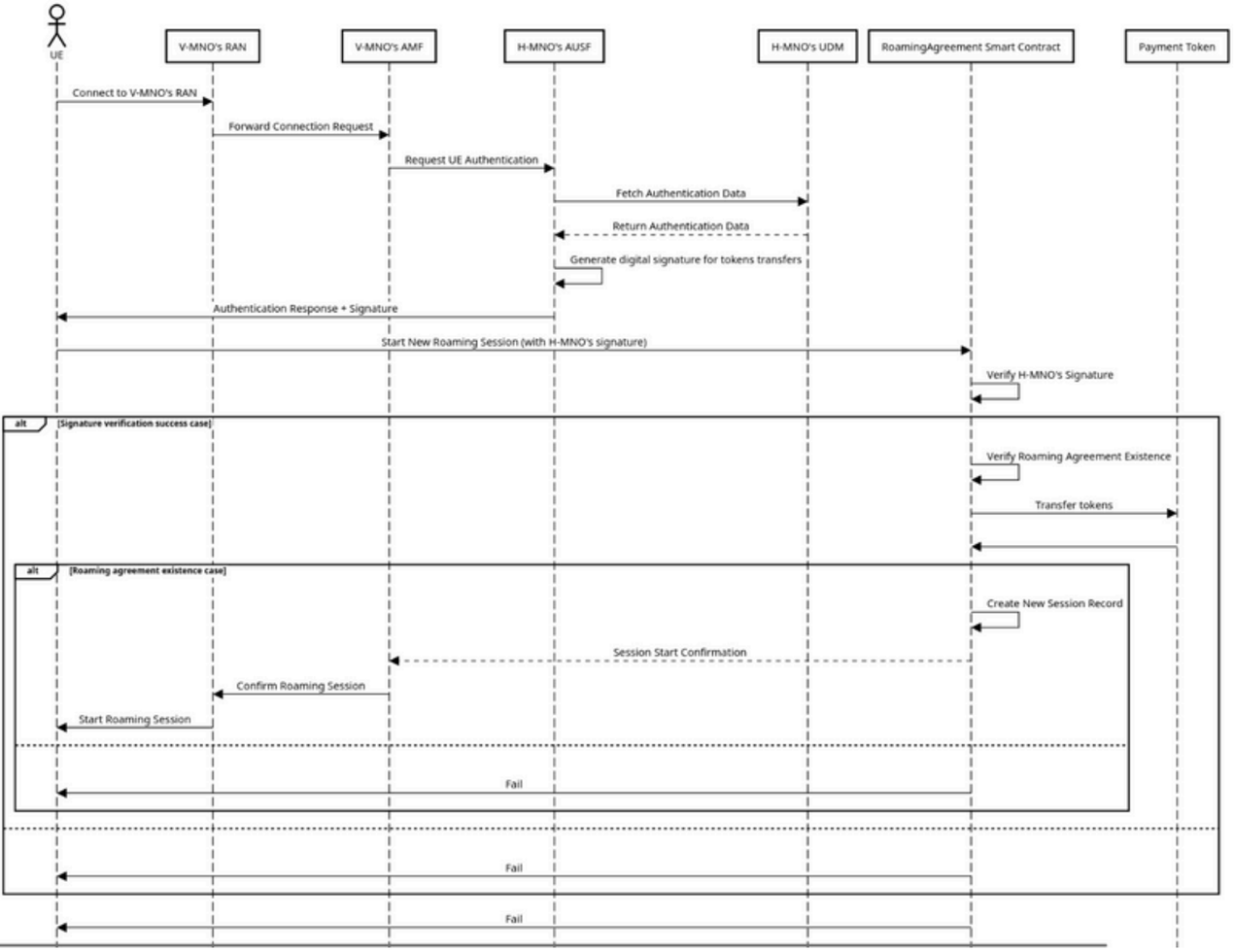
Solution Design



The signature encodes the following essential parameters:

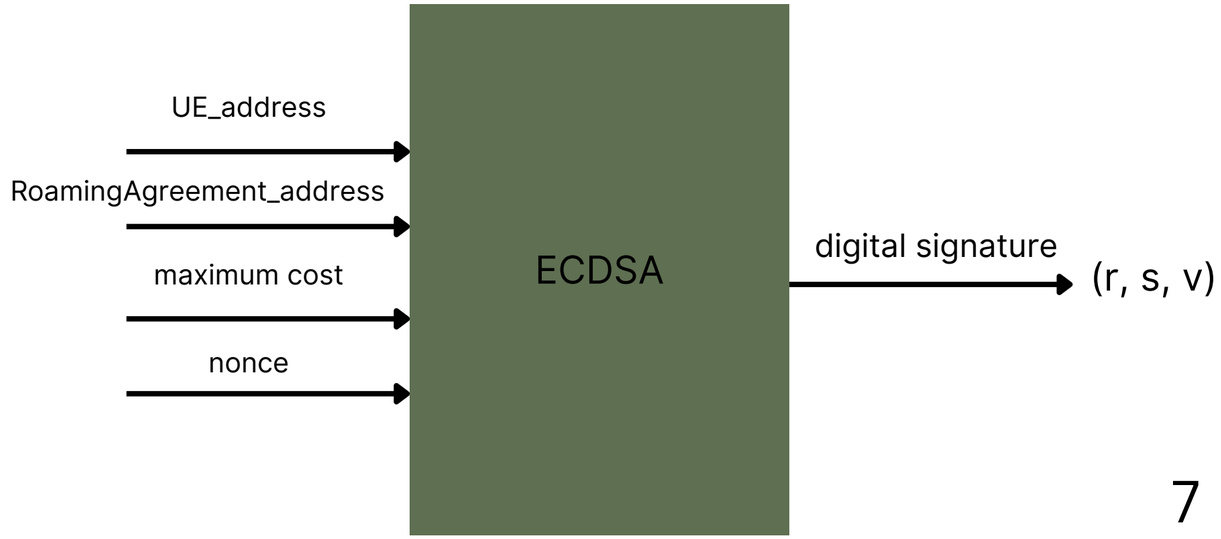
- The UE's unique identifier
- The address of the *RoamingAgreement* smart contract.
- The maximum cost of the session
- A unique session nonce

Solution Design

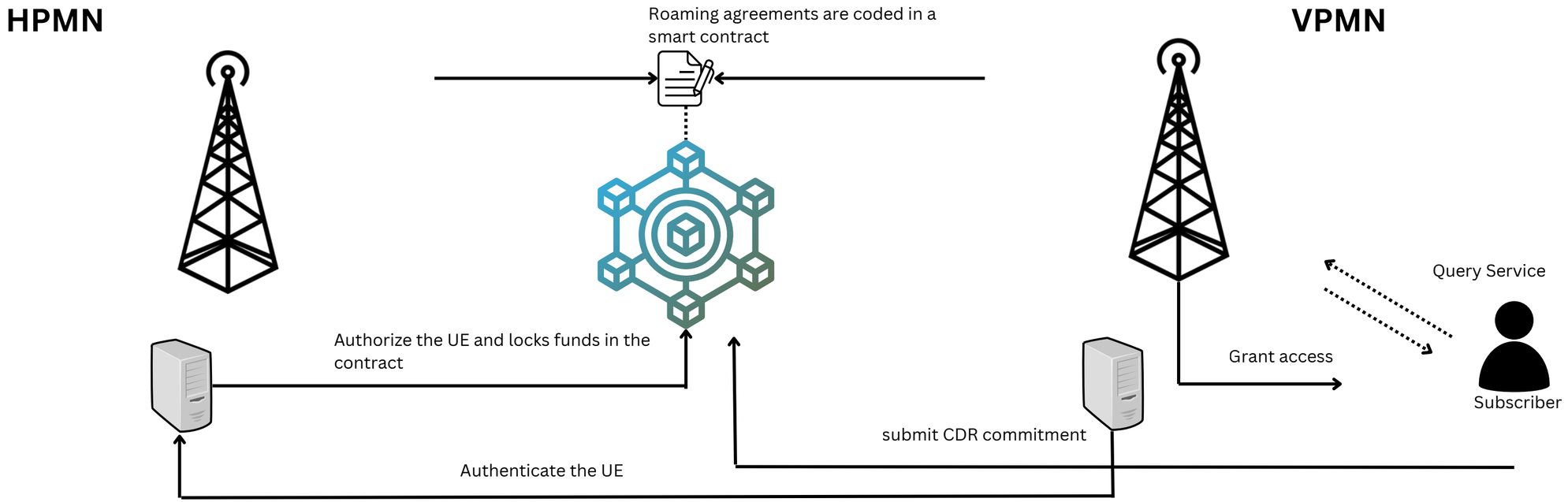


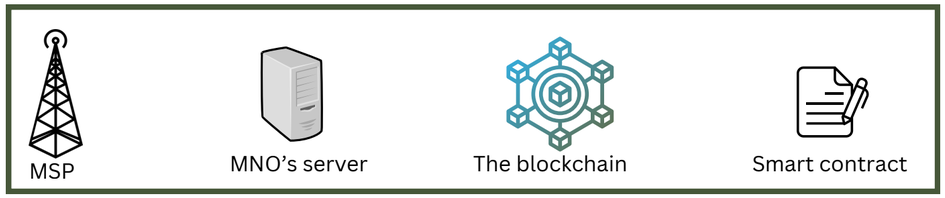
The signature encodes the following essential parameters:

- The UE's unique identifier
- The address of the *RoamingAgreement* smart contract.
- The maximum cost of the session
- A unique session nonce

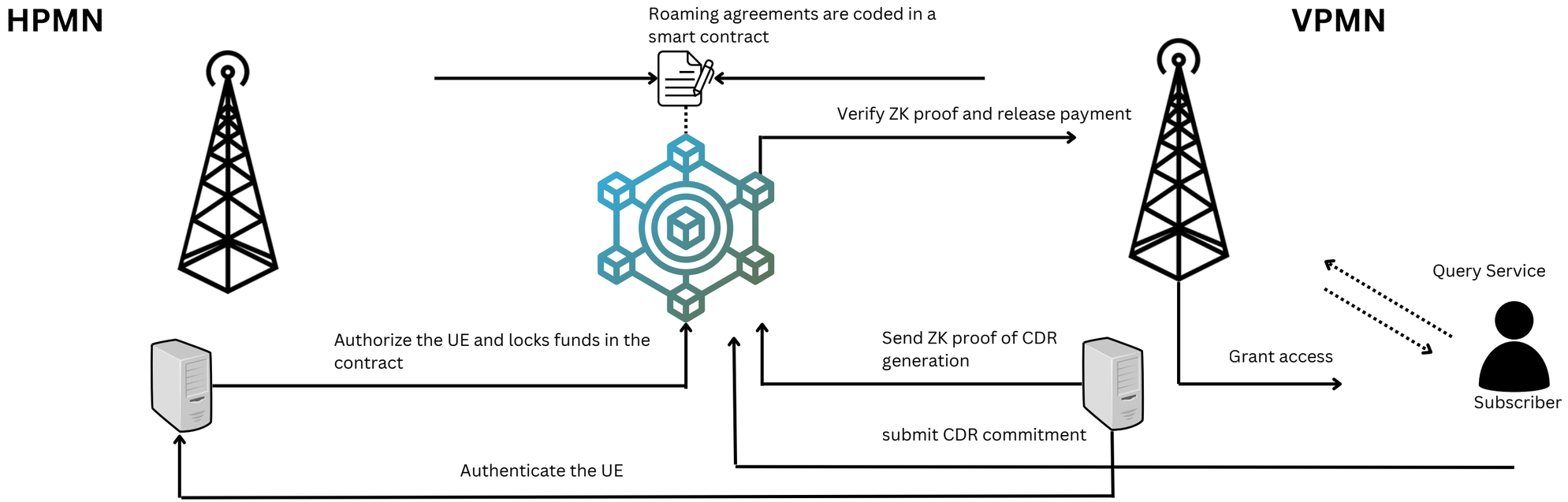


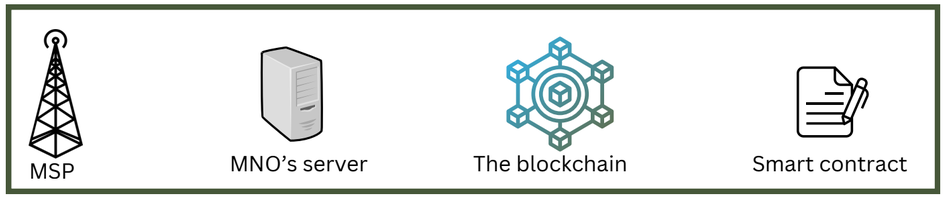
Solution Design



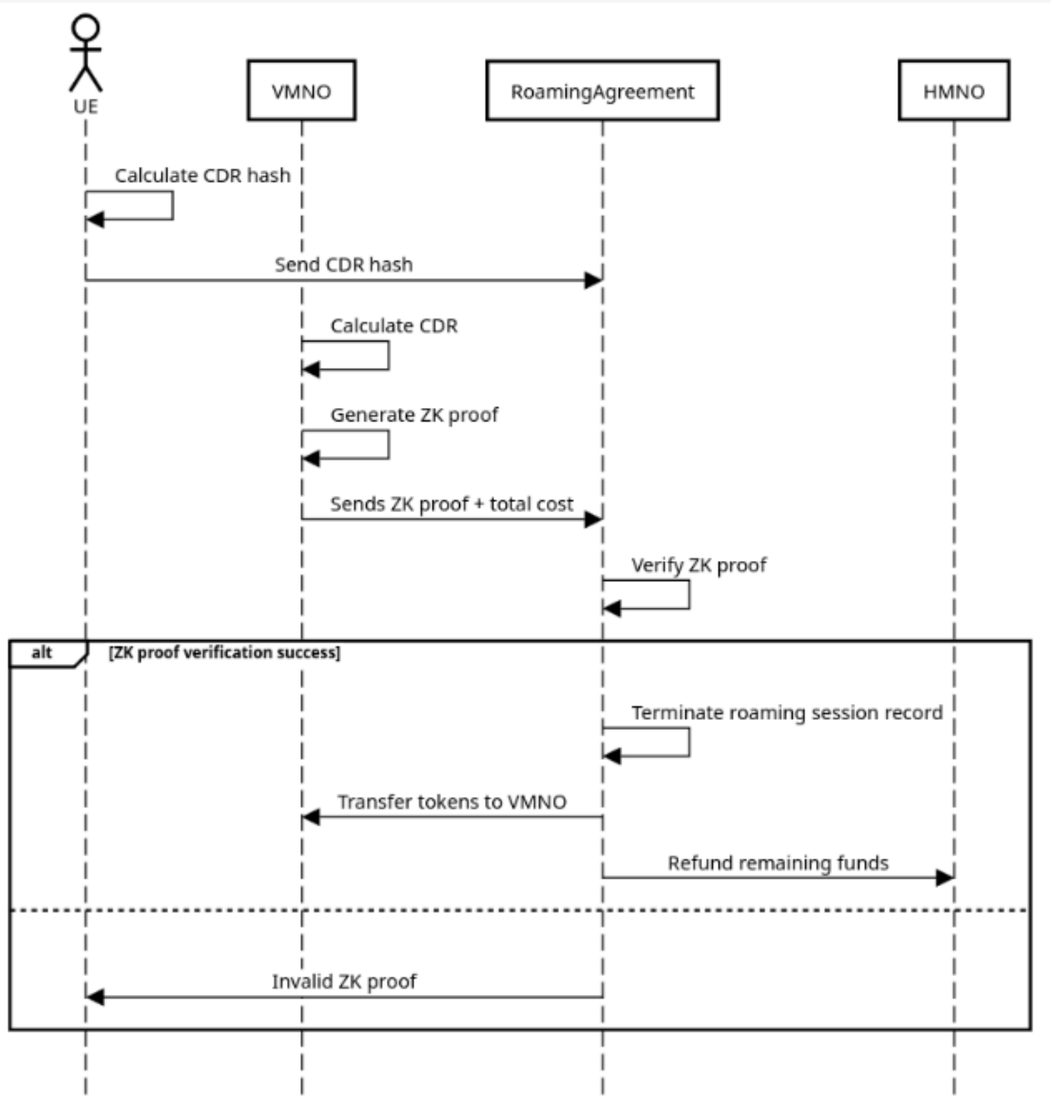


Solution Design

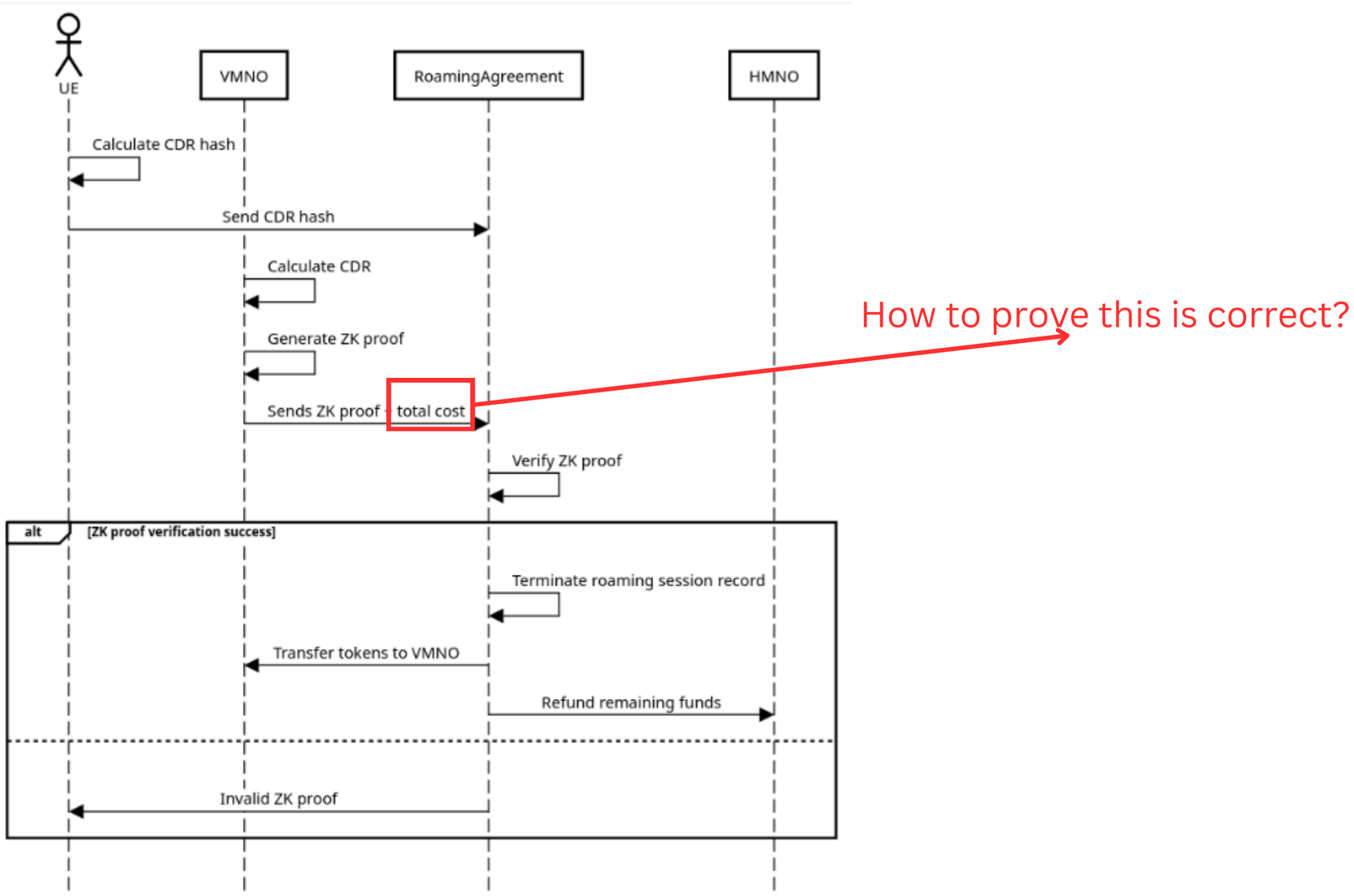




Solution Design

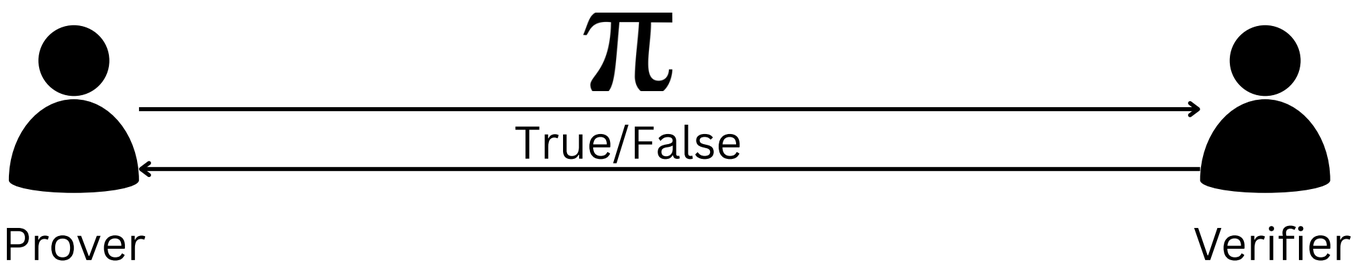


Solution Design



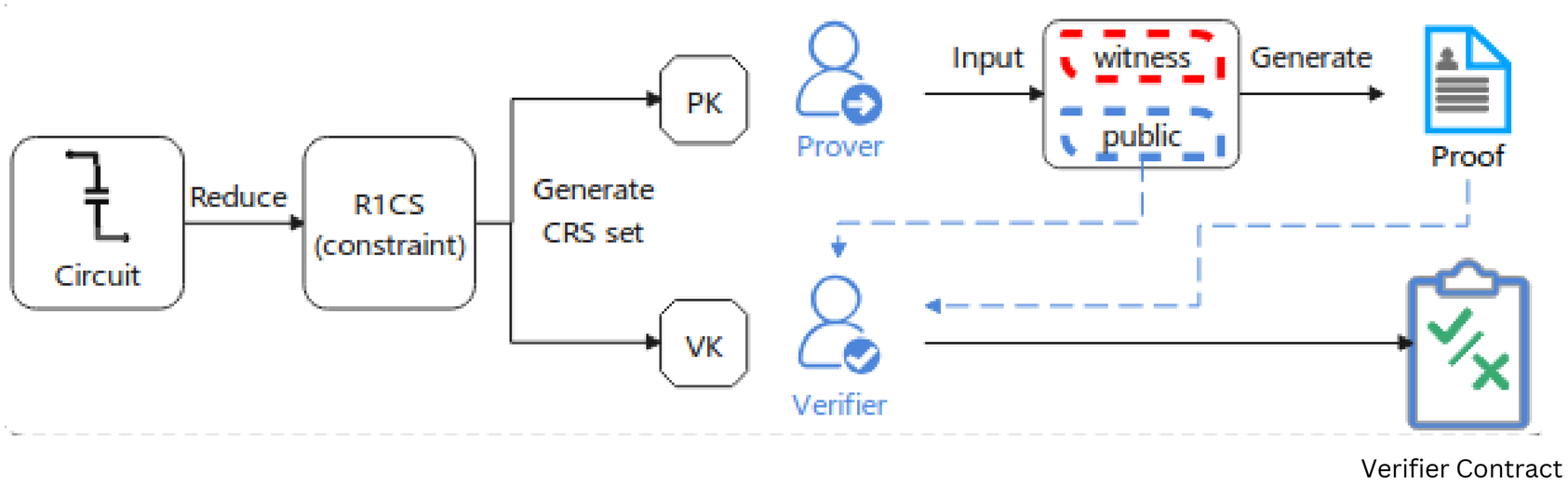
Solution Design

Zero-Knowledge Proofs is a cryptography technique that allows a party (prover) to prove a statement to another party (verifier) without revealing anything else but the fact that the statement is correct



$\pi = C(x, w)$

Solution Design



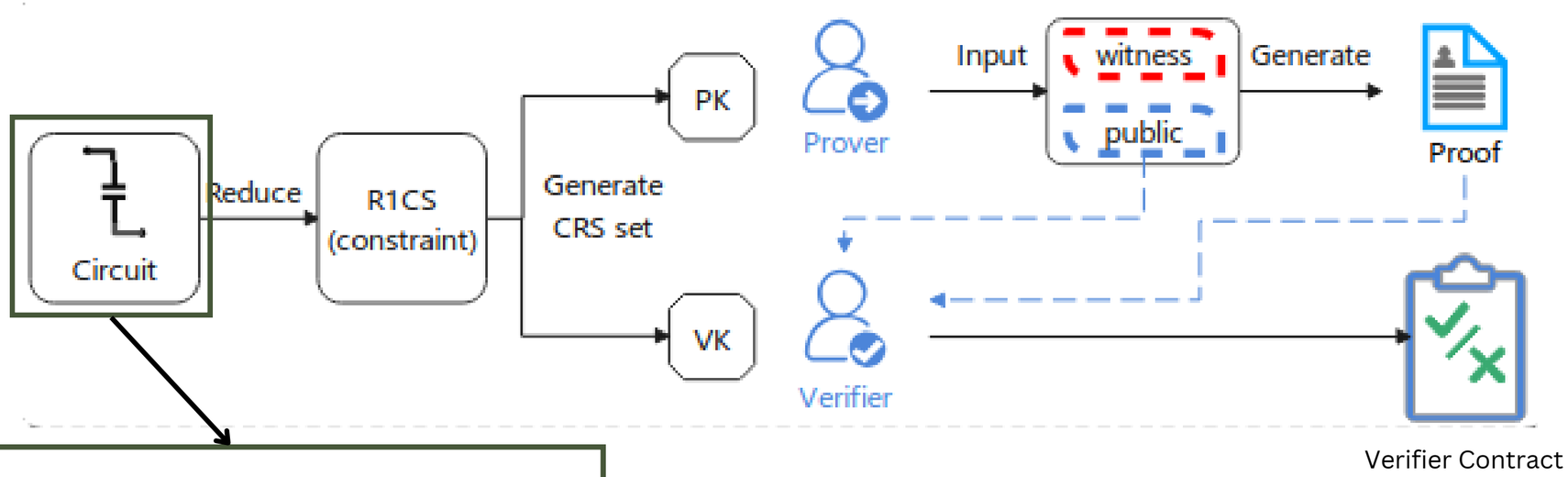
Solution Design

How to prove the correct computation of the total cost?

$$T = < RU > . < RA >$$

$$T = \begin{bmatrix} n_{mb} \\ n_{sms} \\ n_{voice} \end{bmatrix} \times \begin{bmatrix} r_{mb} \\ r_{sms} \\ r_{voice} \end{bmatrix}$$

Solution Design



$$T_{sms} = n_{sms} \times r_{sms}$$

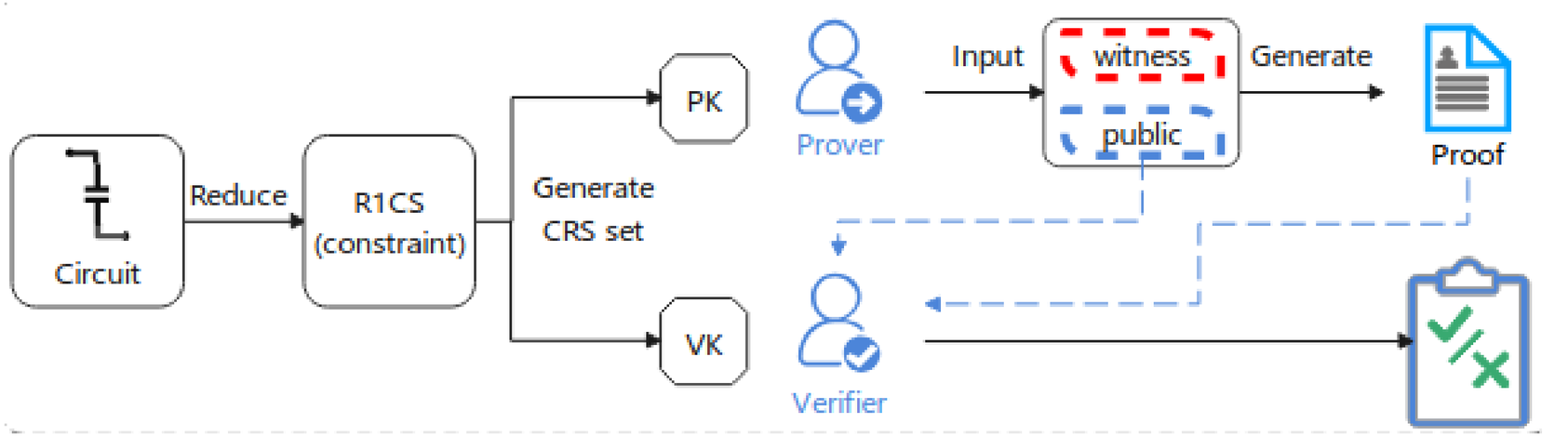
$$T_{mb} = n_{mb} \times r_{mb}$$

$$T_{voice} = n_{voice} \times r_{voice}$$

$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$

$$T_{computed} == T$$

Solution Design



$$T_{sms} = n_{sms} \times r_{sms}$$

$$T_{mb} = n_{mb} \times r_{mb}$$

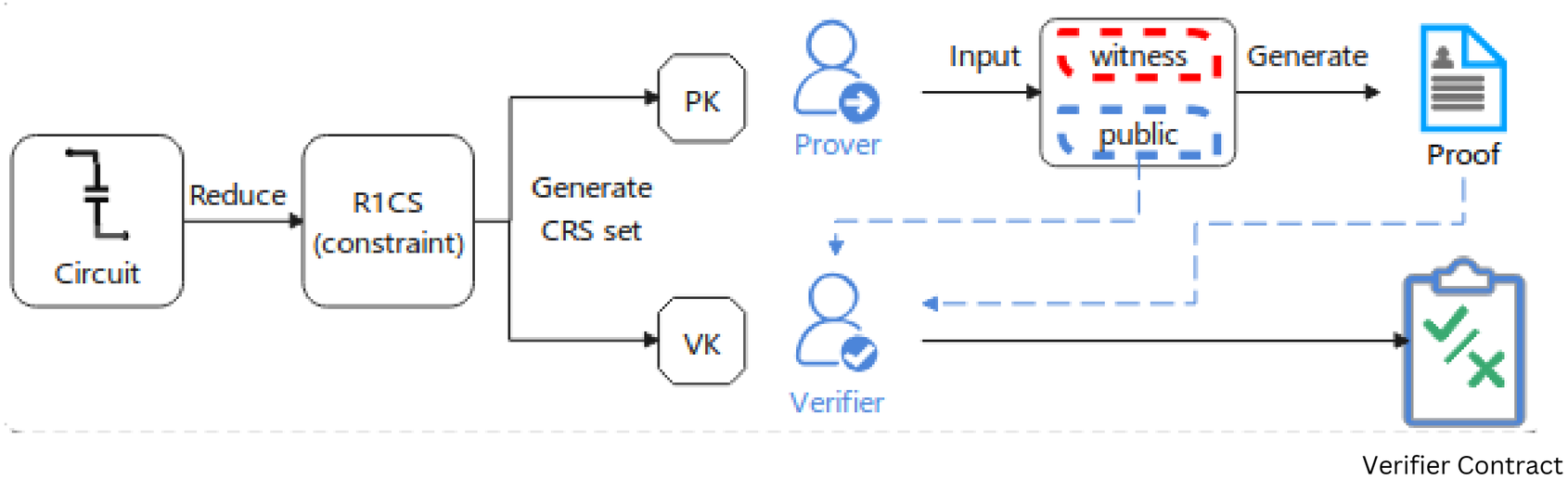
$$T_{voice} = n_{voice} \times r_{voice}$$

$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$

$$T_{computed} == T$$

Circuit inputs

Solution Design



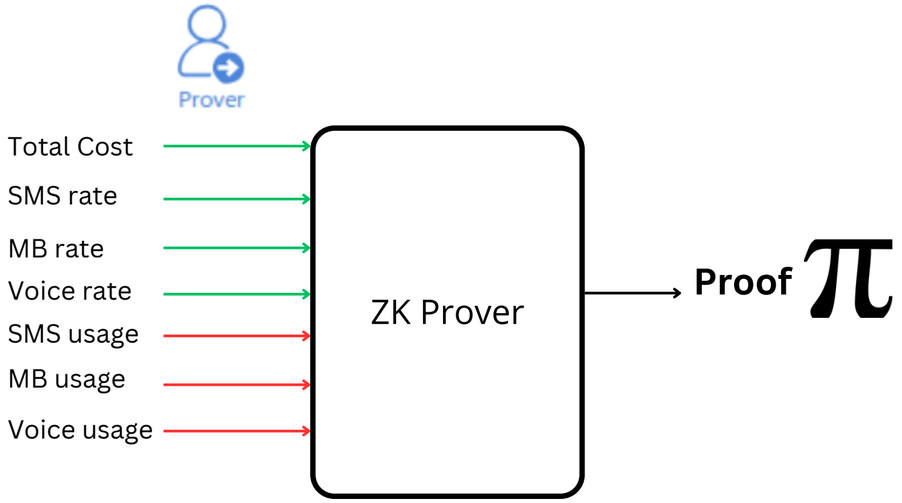
$$T_{sms} = n_{sms} \times r_{sms}$$

$$T_{mb} = n_{mb} \times r_{mb}$$

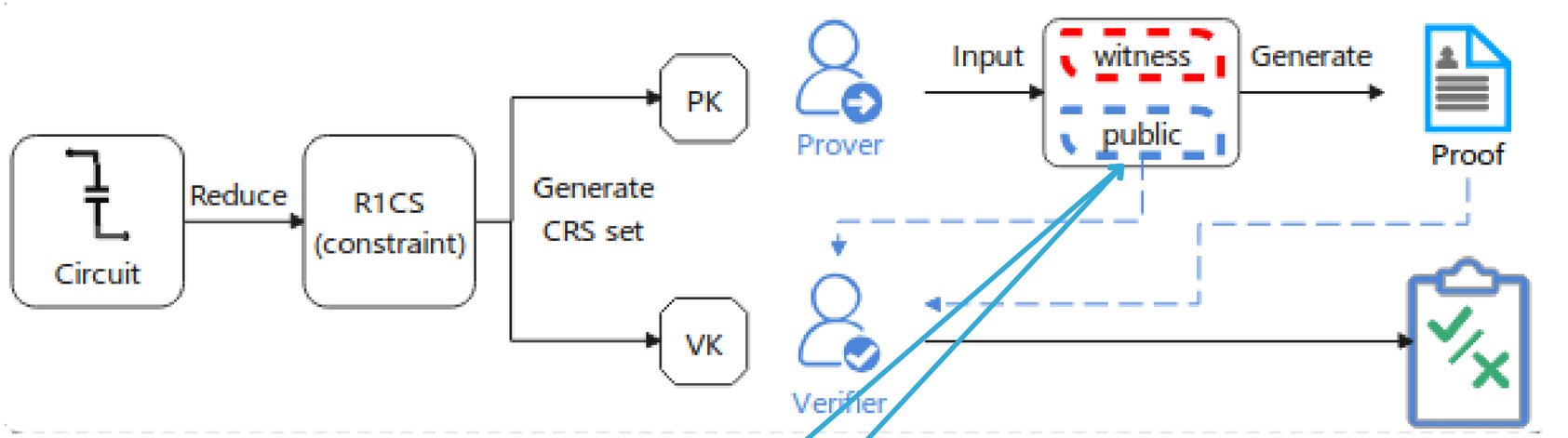
$$T_{voice} = n_{voice} \times r_{voice}$$

$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$

$$T_{computed} == T$$



Solution Design



$$T_{sms} = n_{sms} \times r_{sms}$$

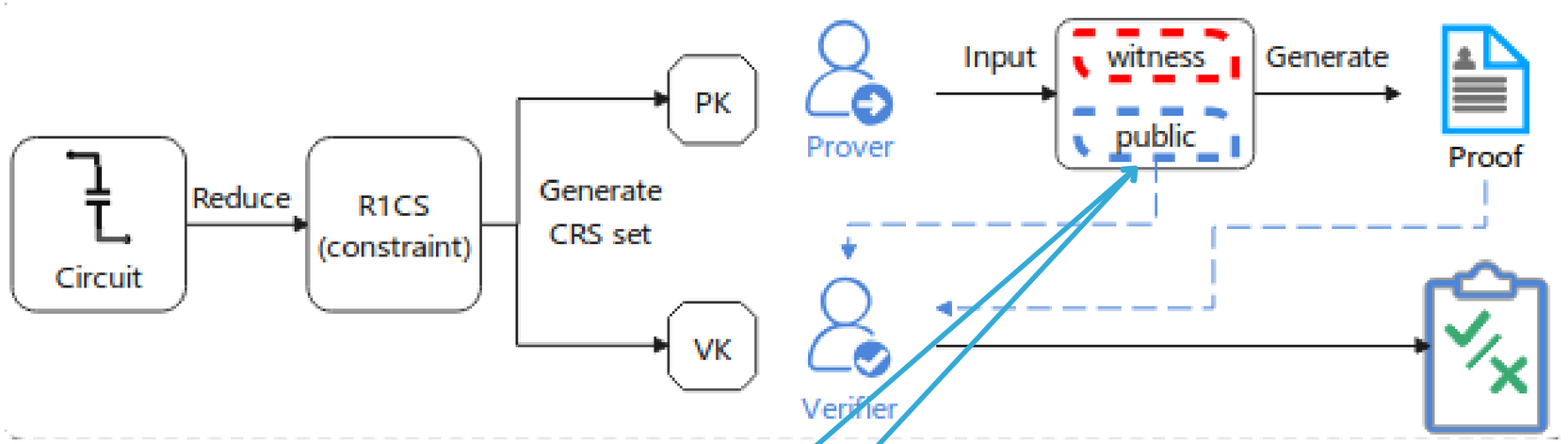
$$T_{mb} = n_{mb} \times r_{mb}$$

$$T_{voice} = n_{voice} \times r_{voice}$$

$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$

$$T_{computed} == T$$

Solution Design



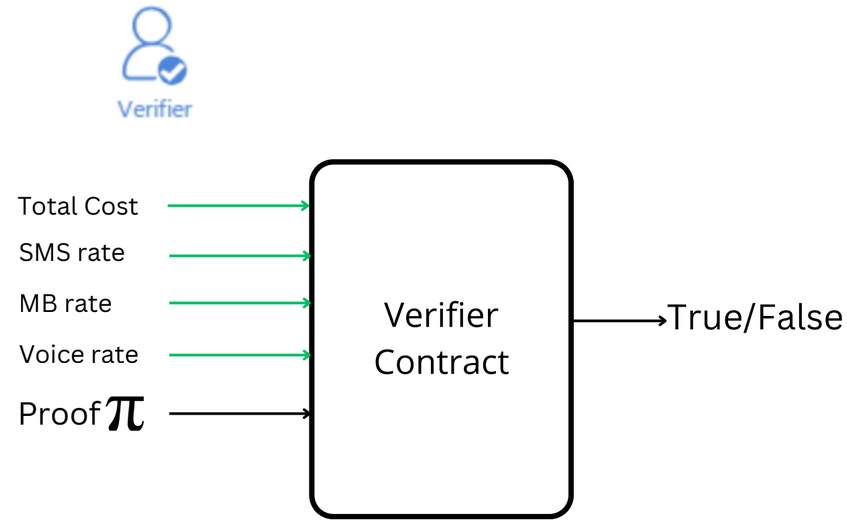
$$T_{sms} = n_{sms} \times r_{sms}$$

$$T_{mb} = n_{mb} \times r_{mb}$$

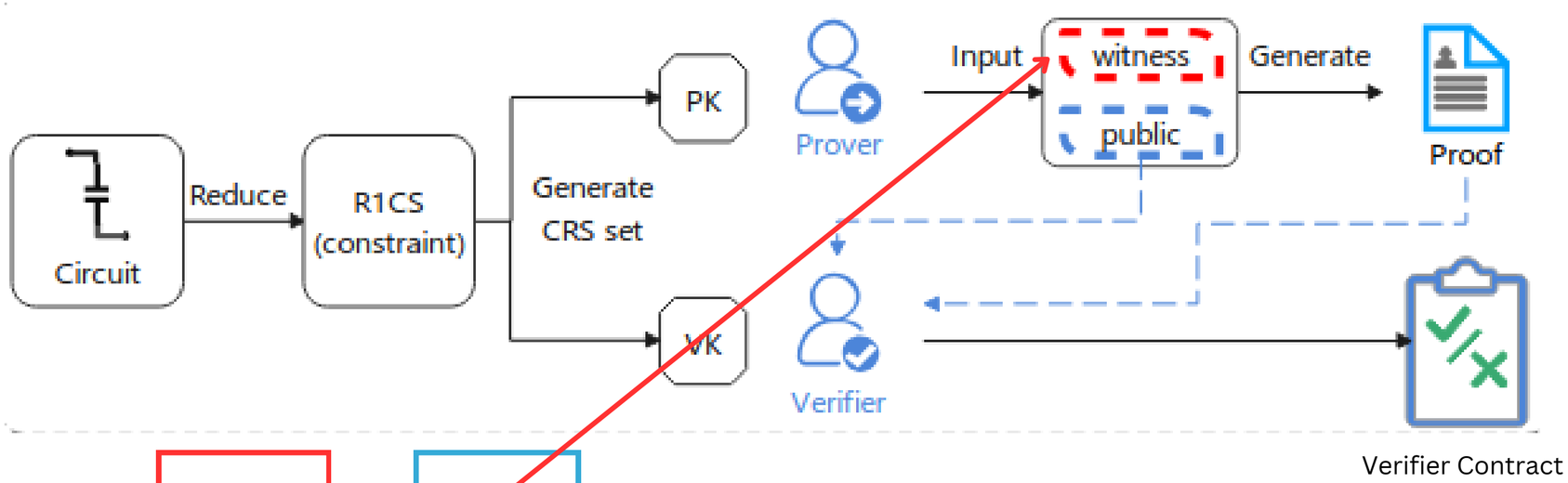
$$T_{voice} = n_{voice} \times r_{voice}$$

$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$

$$T_{computed} == T$$



Solution Design



$$T_{sms} = n_{sms} \times r_{sms}$$

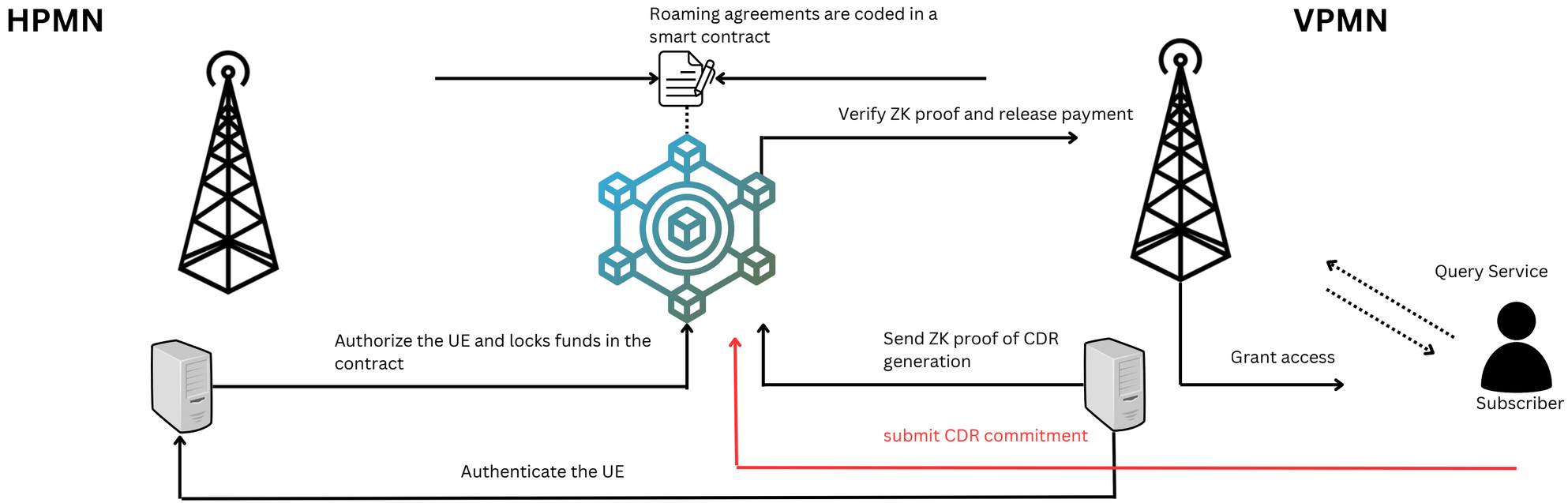
$$T_{mb} = n_{mb} \times r_{mb}$$

$$T_{voice} = n_{voice} \times r_{voice}$$

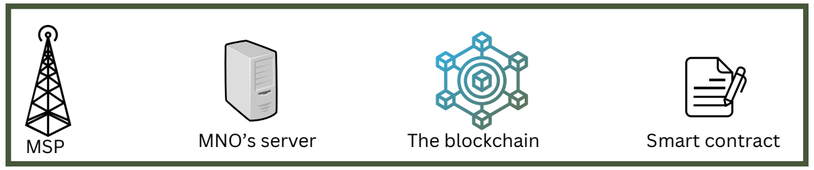
$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$

$$T_{computed} == T$$

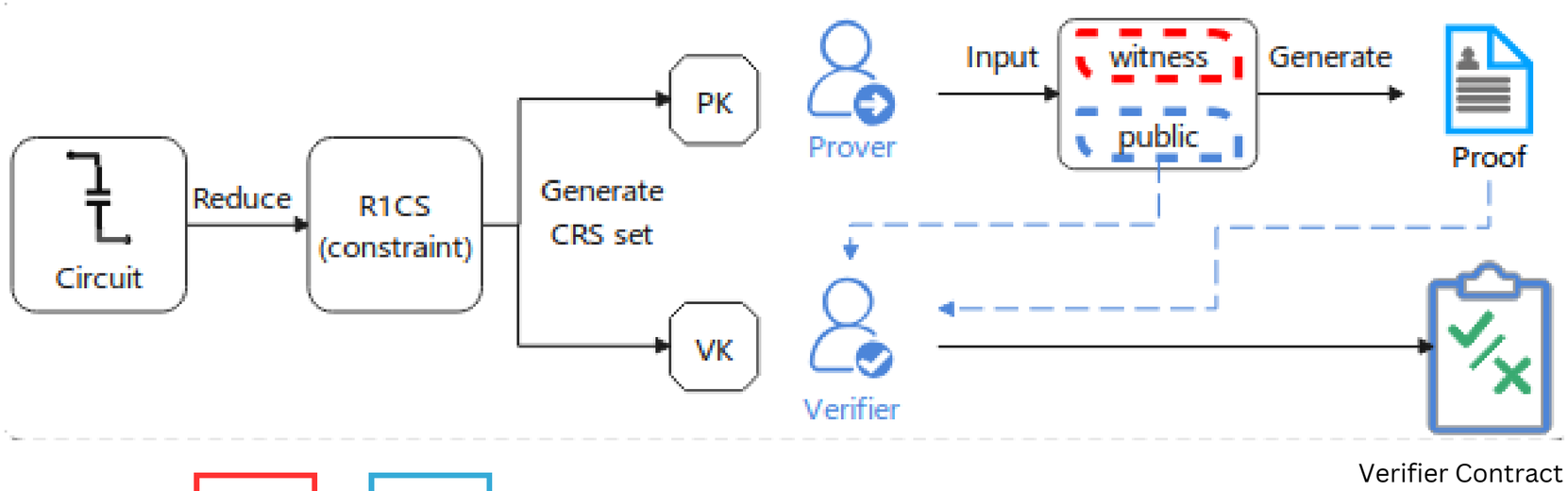
Solution Design



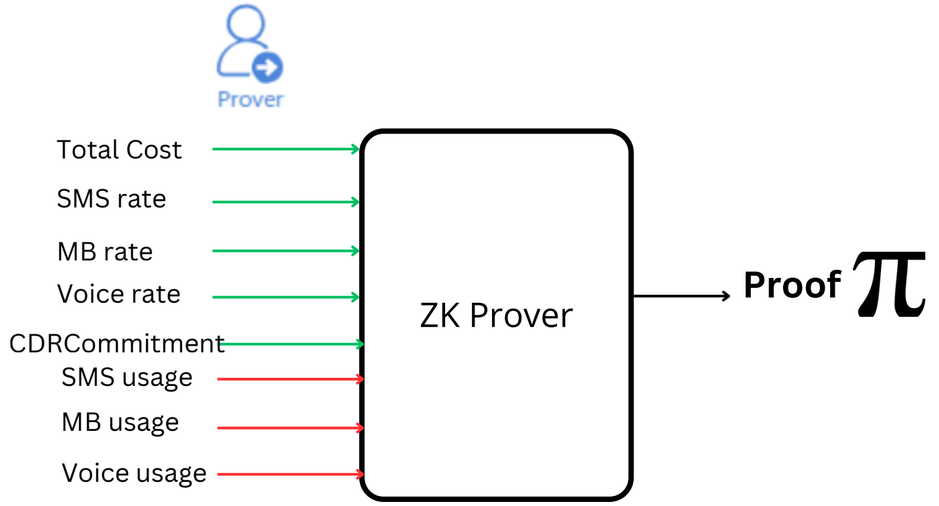
commitment = Poseidon(sms_count, voice_minutes, data_megabytes)



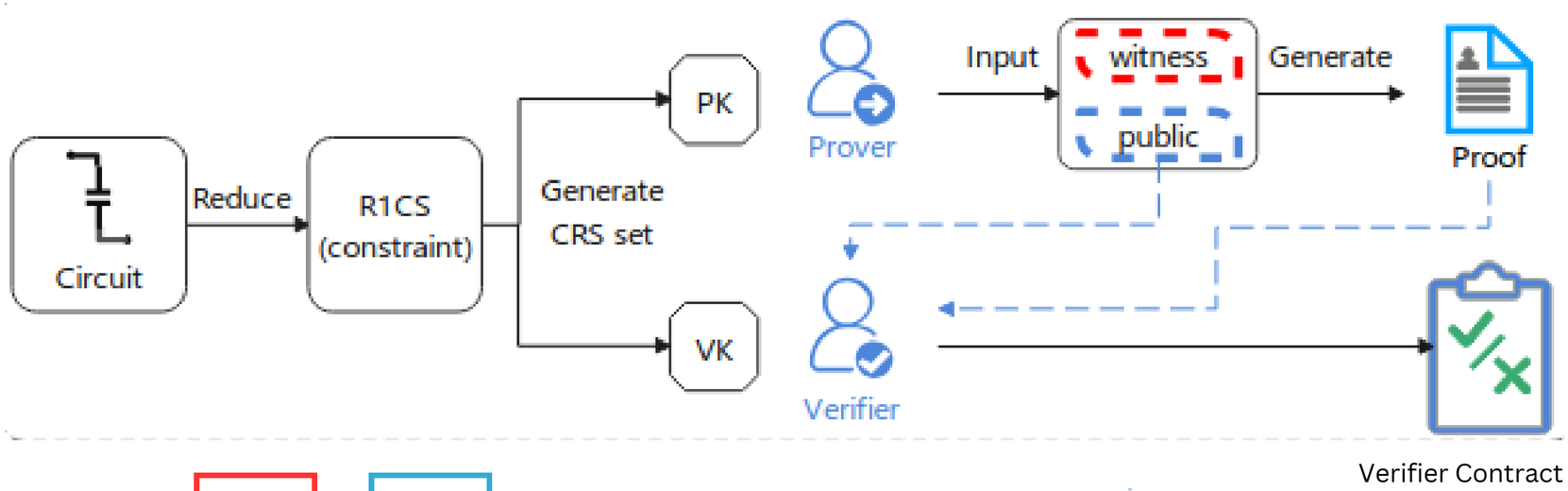
Solution Design



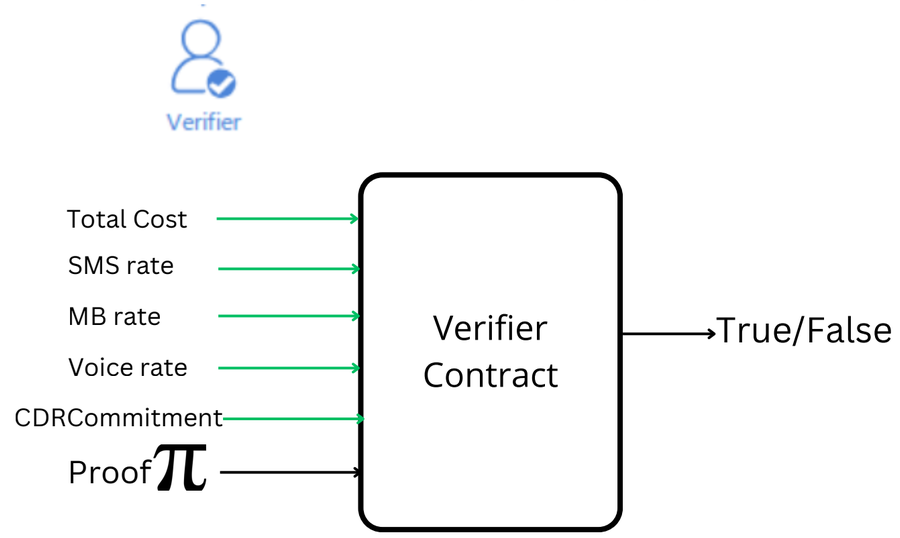
$$T_{sms} = n_{sms} \times r_{sms}$$
$$T_{mb} = n_{mb} \times r_{mb}$$
$$T_{voice} = n_{voice} \times r_{voice}$$
$$H = poseidon\,(n_{sms}, n_{mb}, n_{voice})$$
$$H == CDR_{commitment}$$
$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$
$$T_{computed} == T$$



Solution Design

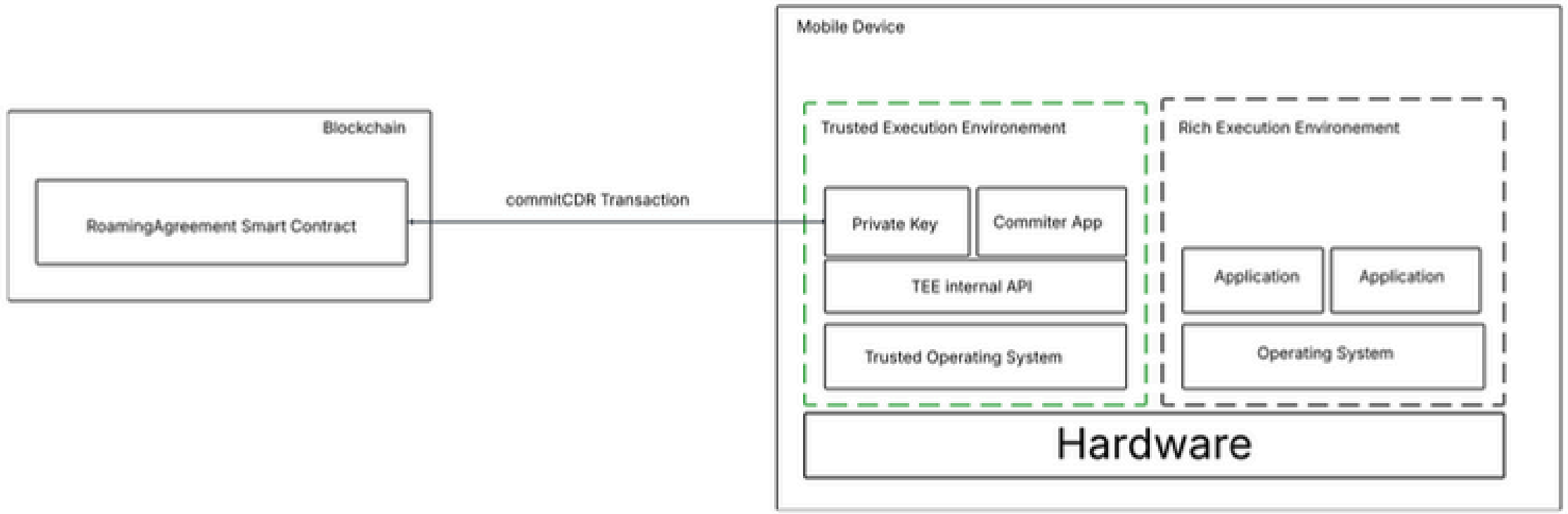


$$T_{sms} = n_{sms} \times r_{sms}$$
$$T_{mb} = n_{mb} \times r_{mb}$$
$$T_{voice} = n_{voice} \times r_{voice}$$
$$H = poseidon\,(n_{sms}, n_{mb}, n_{voice})$$
$$H == CDR_{commitment}$$
$$T_{computed} = T_{sms} + T_{mb} + T_{voice}$$
$$T_{computed} == T$$



Solution Design

We assume the UE is equiped with a TEE to submit his CDR



Solution Design

ZK Proof Security Properties

We denote $R(x, w)$ the relation that expresses the zk verification algorithm

- Completeness

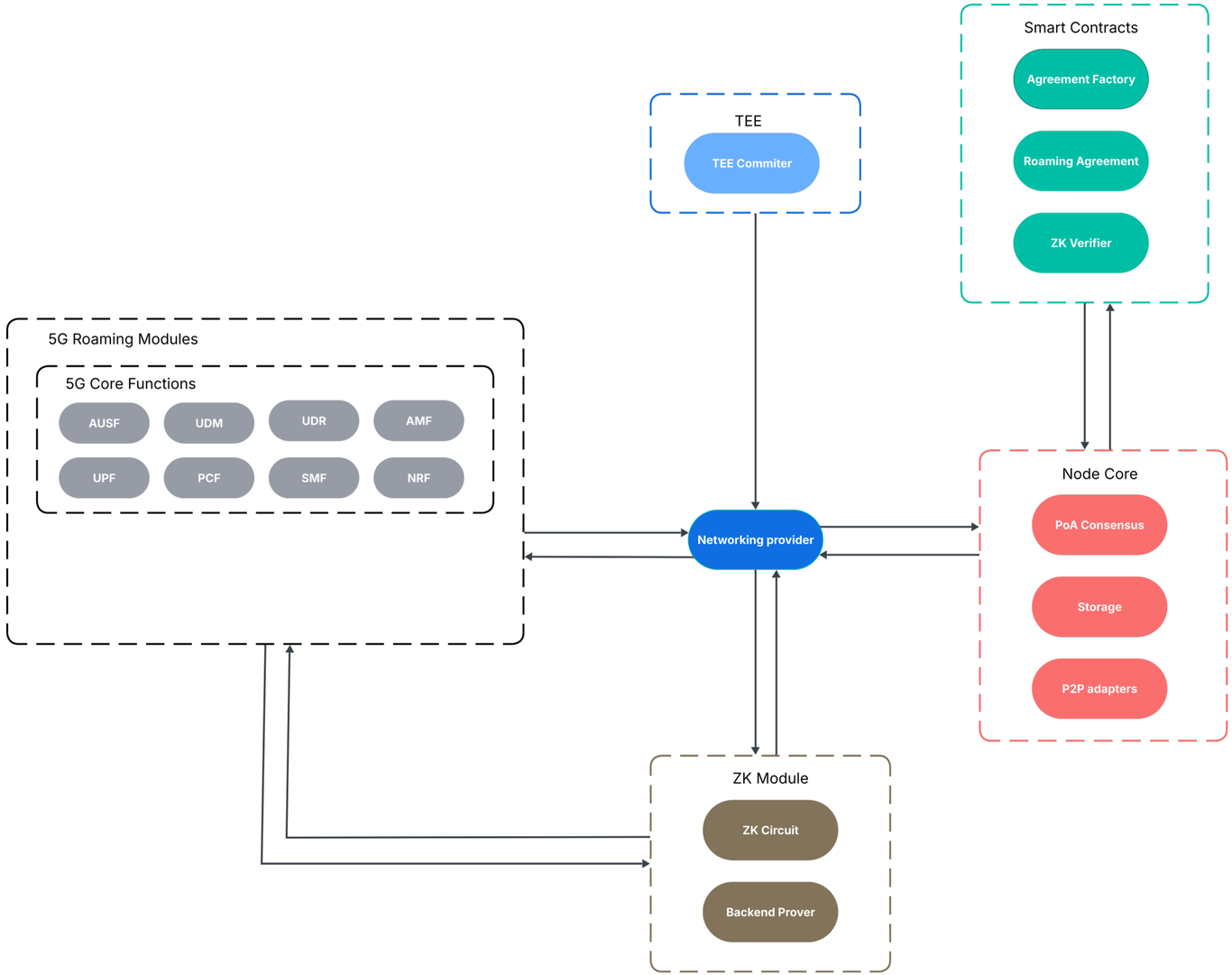
$$\forall (x, w) : R(x, w) = 1 \implies \Pr\left[\text{Verify}(x, \text{Prove}(x, w)) = 1\right] \approx 1$$

- Soundness

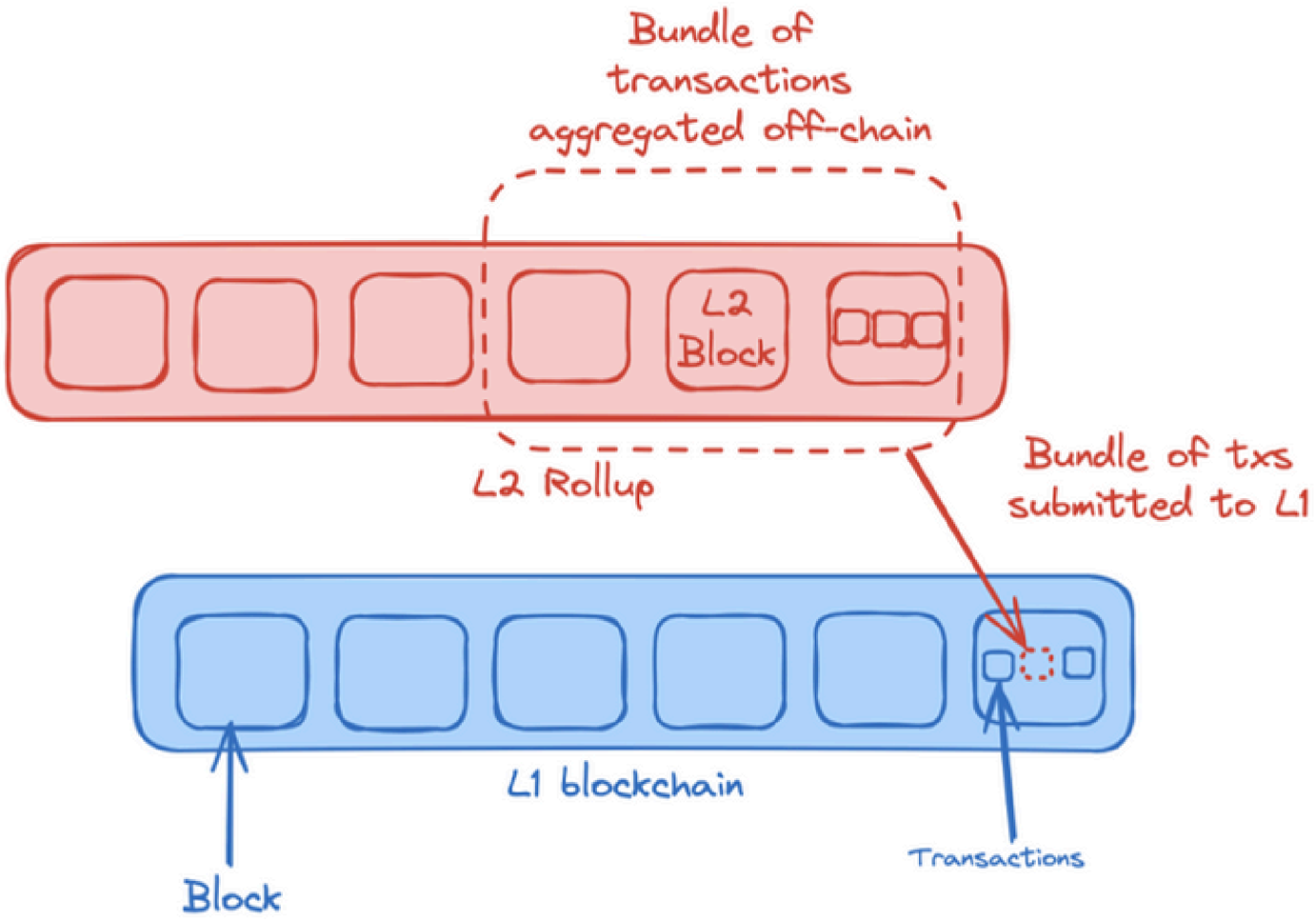
$$\Pr\left[\text{Verify}(x, \pi) = 1 \ \wedge \ \nexists w : R(x, w) = 1\right] \leq \epsilon$$

- Zero-Knowledge

Framework Modules



Zk-Rollup L2


Bundle of
transactions
aggregated off-chain
L2
Block
L2 Rollup
Bundle of txs
submitted to L1
L1 blockchain
Block
Transactions

Tests & Results

ZK proof benchmarks

- Proof Generation Overhead: Memory and Time Analysis
- On-Chain Proofs Verification Overhead: Gas Consumption

Tests & Results

ZK proof benchmarks

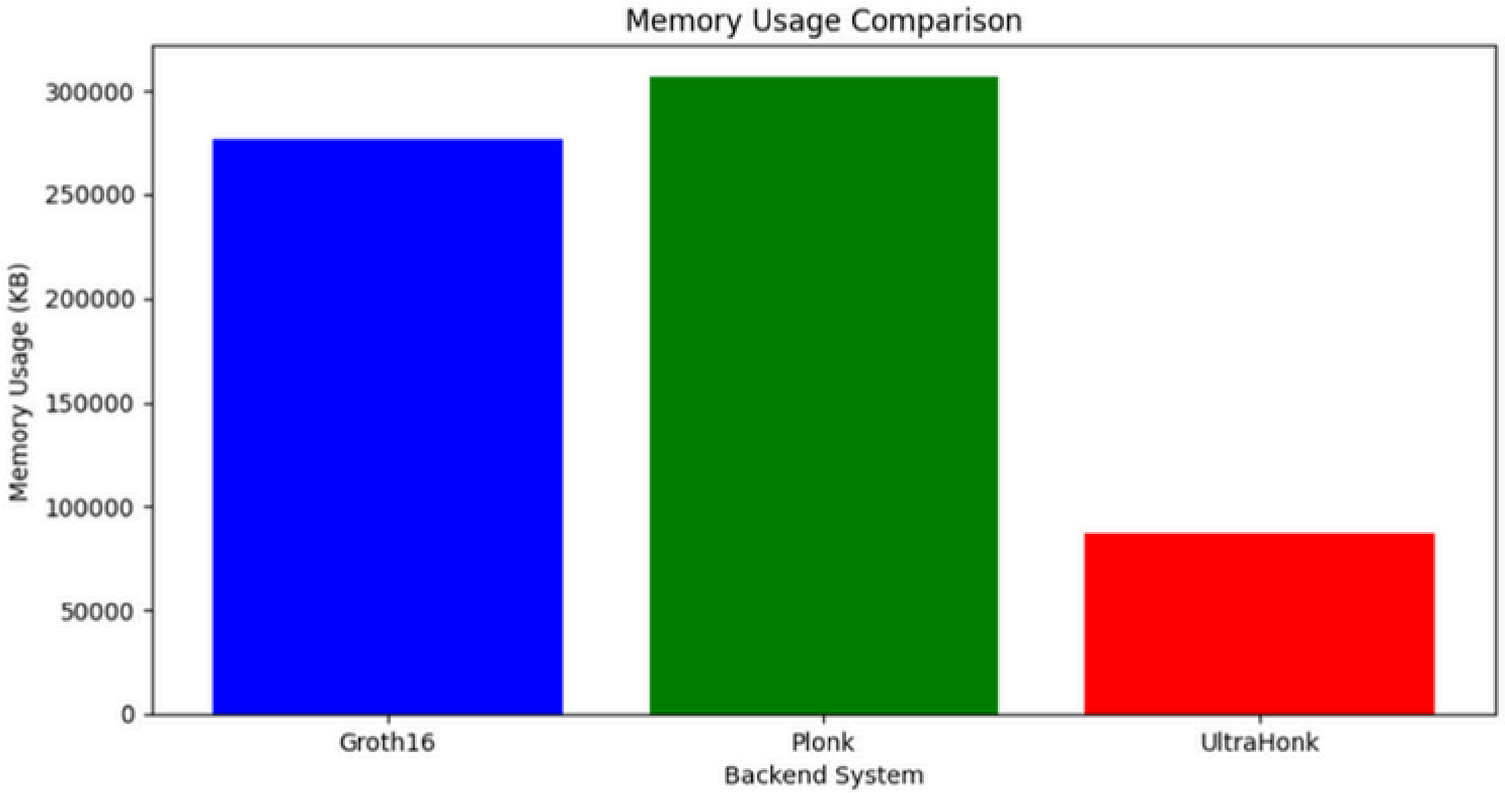
- Proof Generation Overhead: Memory and Time Analysis
- On-Chain Proofs Verification Overhead: Gas Consumption

System flows’ throughput benchmark

- Create a roaming agreement
- Starting a roaming session
- Setteling a roaming session

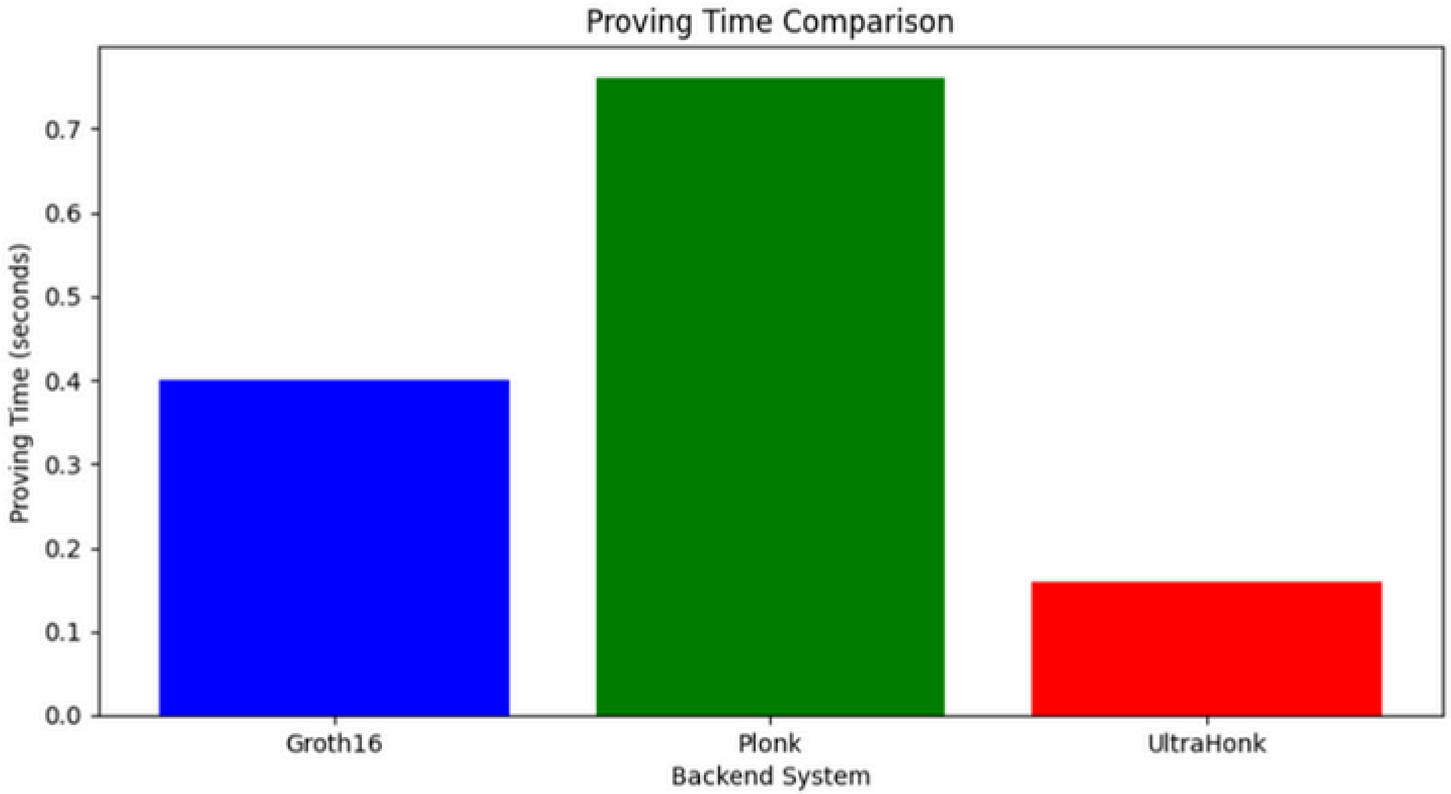
Tests & Results

ZK Proof Generation Overhead: Memory and Time Analysis



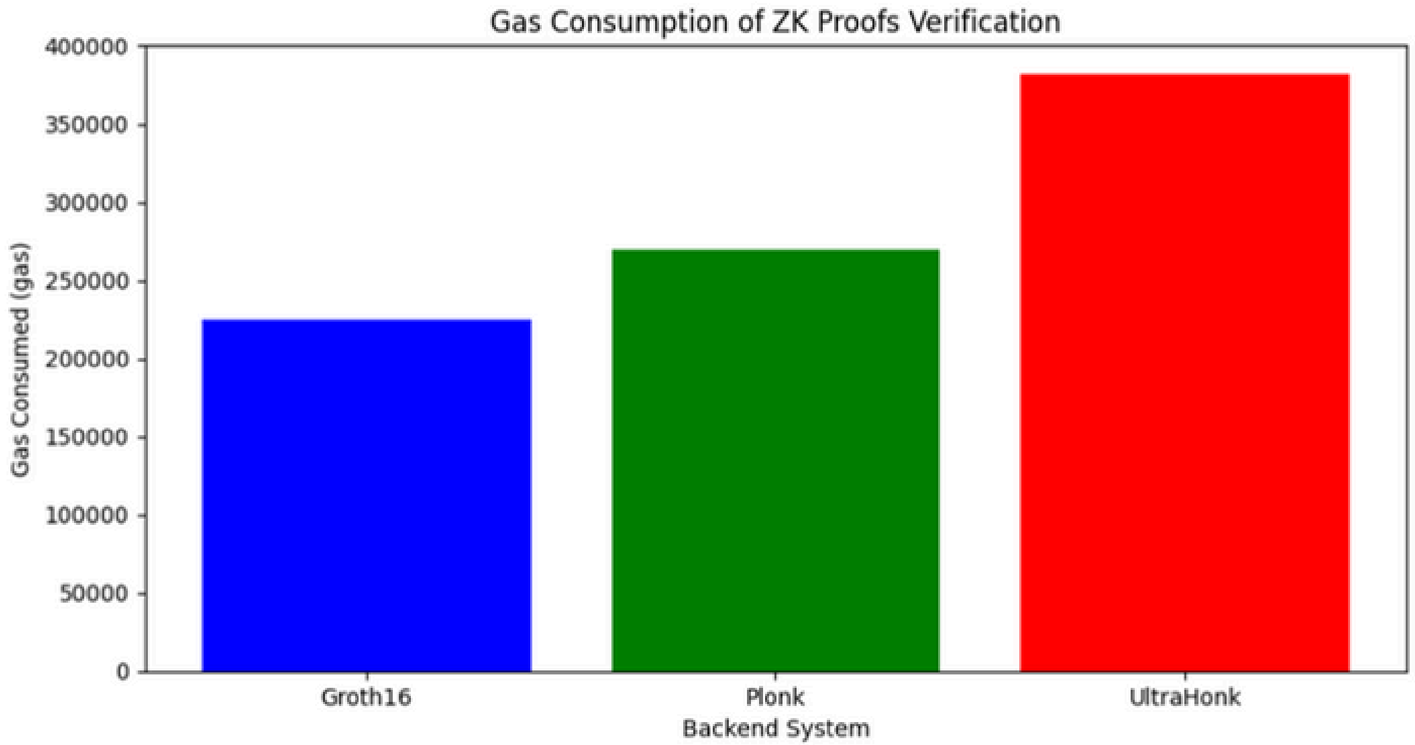
Tests & Results

ZK Proof Generation Overhead: Memory and Time Analysis



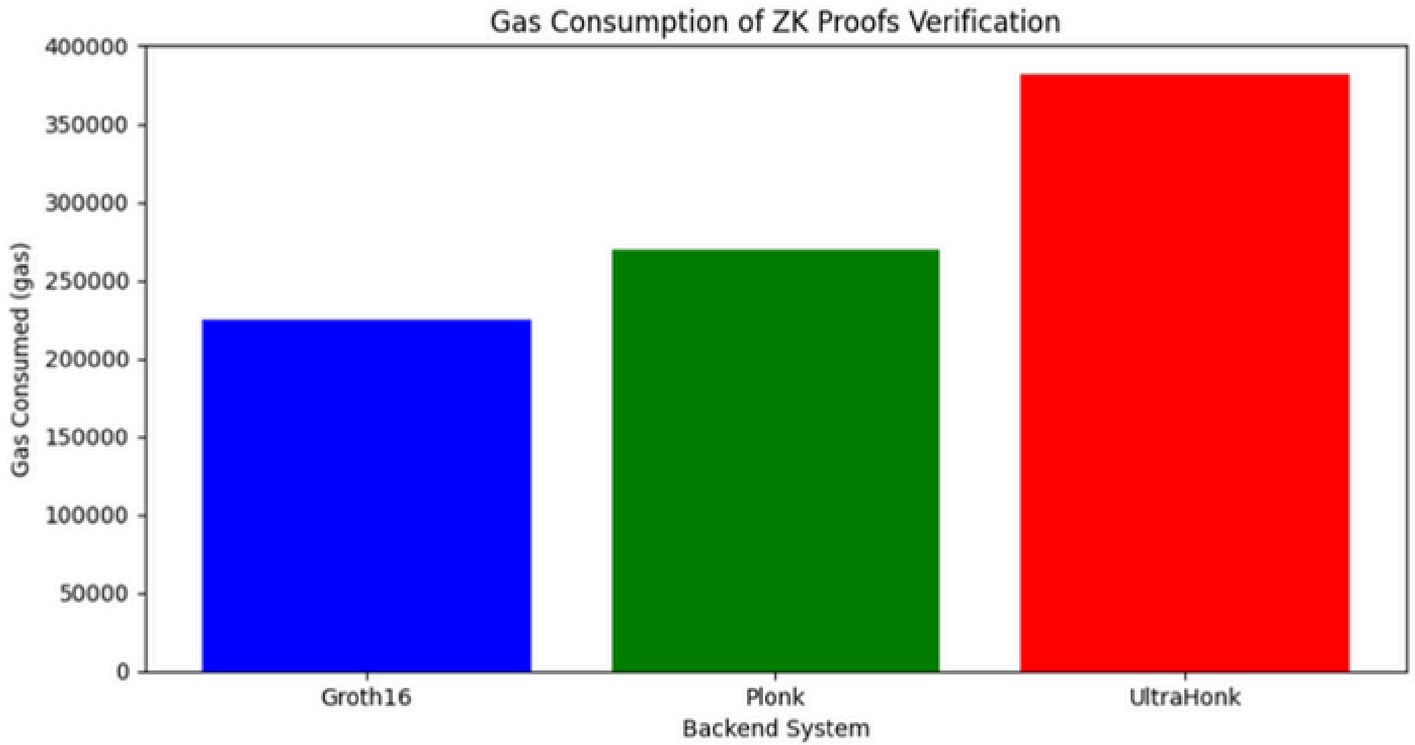
Tests & Results

On-Chain Proofs Verification Overhead: Gas Consumption



Tests & Results

On-Chain Proofs Verification Overhead: Gas Consumption



Even with Groth16; on Ethereum, one proof verification is about 0.6% of the total block gas limit, e.g. 154 simultaneous proof verification hits the block gas limit!!!

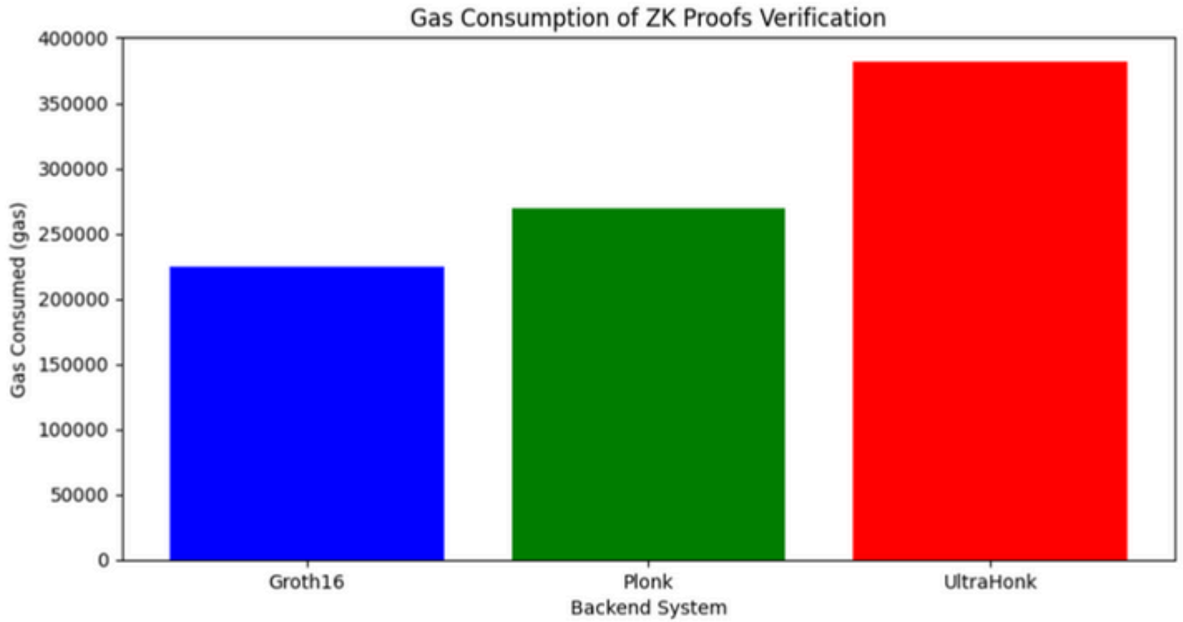
Tests & Results

On-Chain Proofs Verification Overhead: Gas Consumption

	Groth16	Plonk	UltraHonk
Transaction Cost	$2.28	$3.12	$4.34

Considering that:

- Gas price = 2.5 Gwei
- ETH price = $4468



Tests & Results

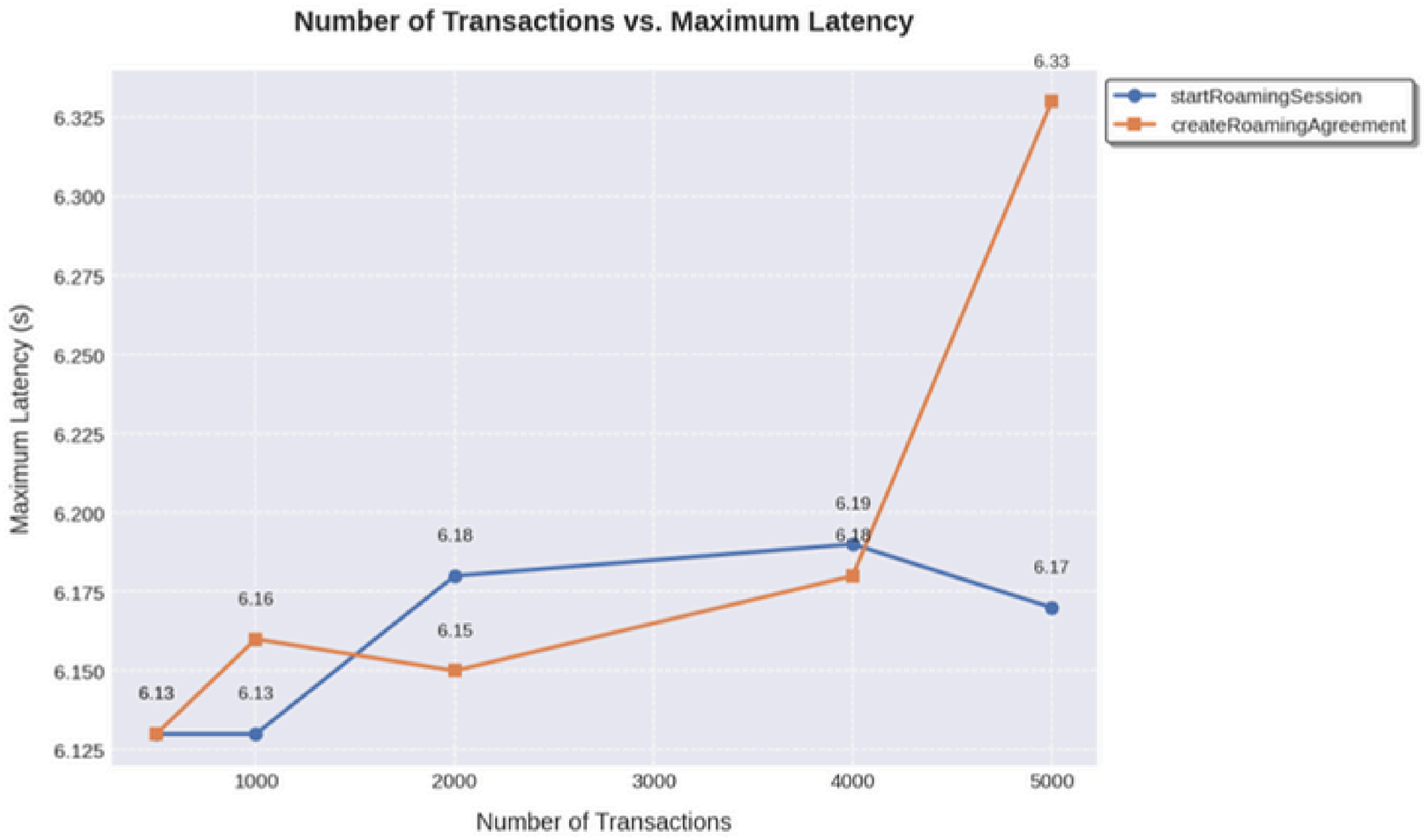
System flows benchmark

- All benchmarks for the B5GRoam protocol were performed on a Dell XPS 15 9530 laptop equipped with a 12th-generation Intel Core i7 processor (14 cores) and 16GB of RAM. The system operates under Manjaro Arc, an Arch-based Linux distribution.

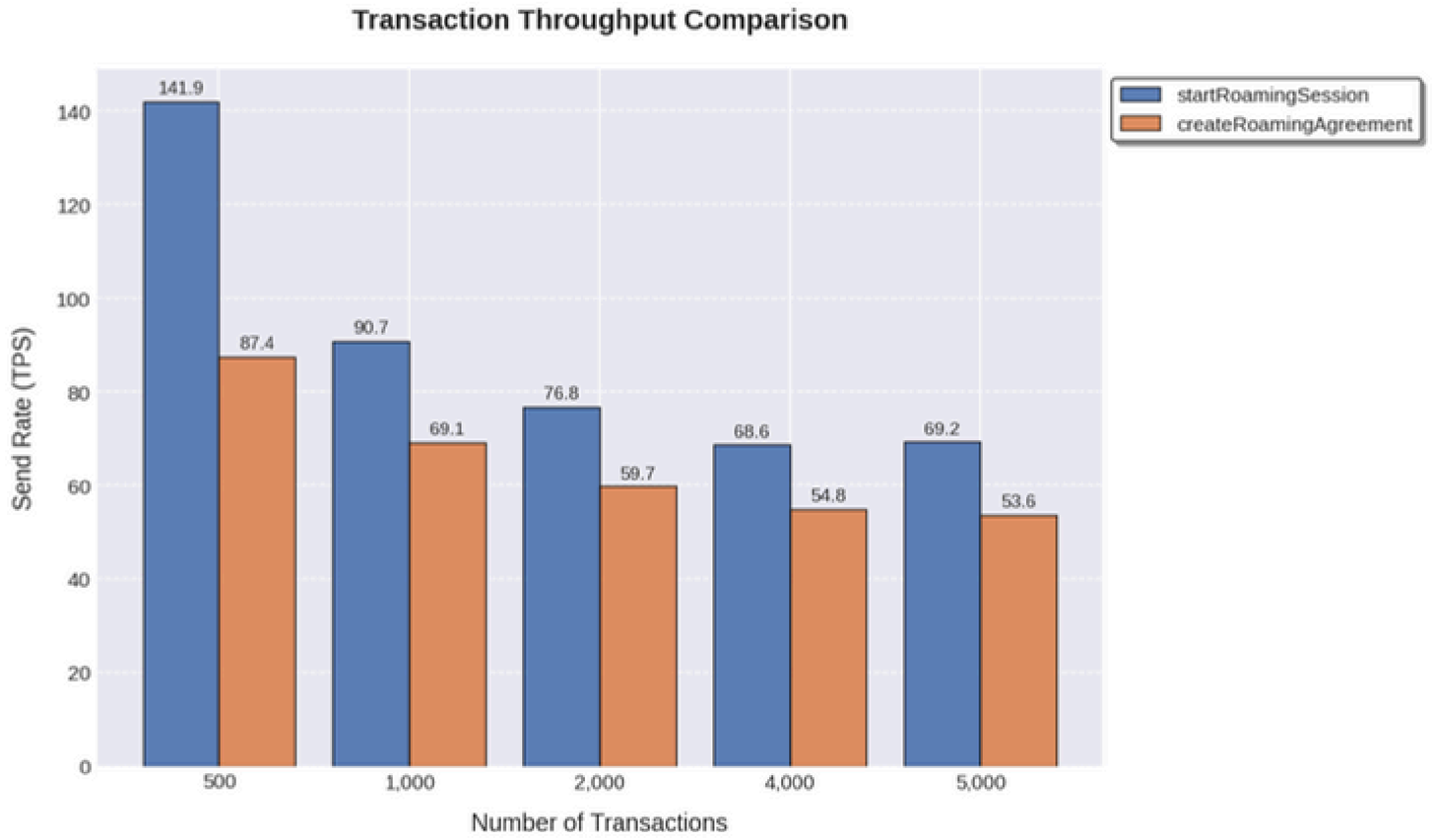
- We used Geth to bootstrap a local blockchain, consisting of 4 nodes.

- We used Caliper for blockchain benchmark

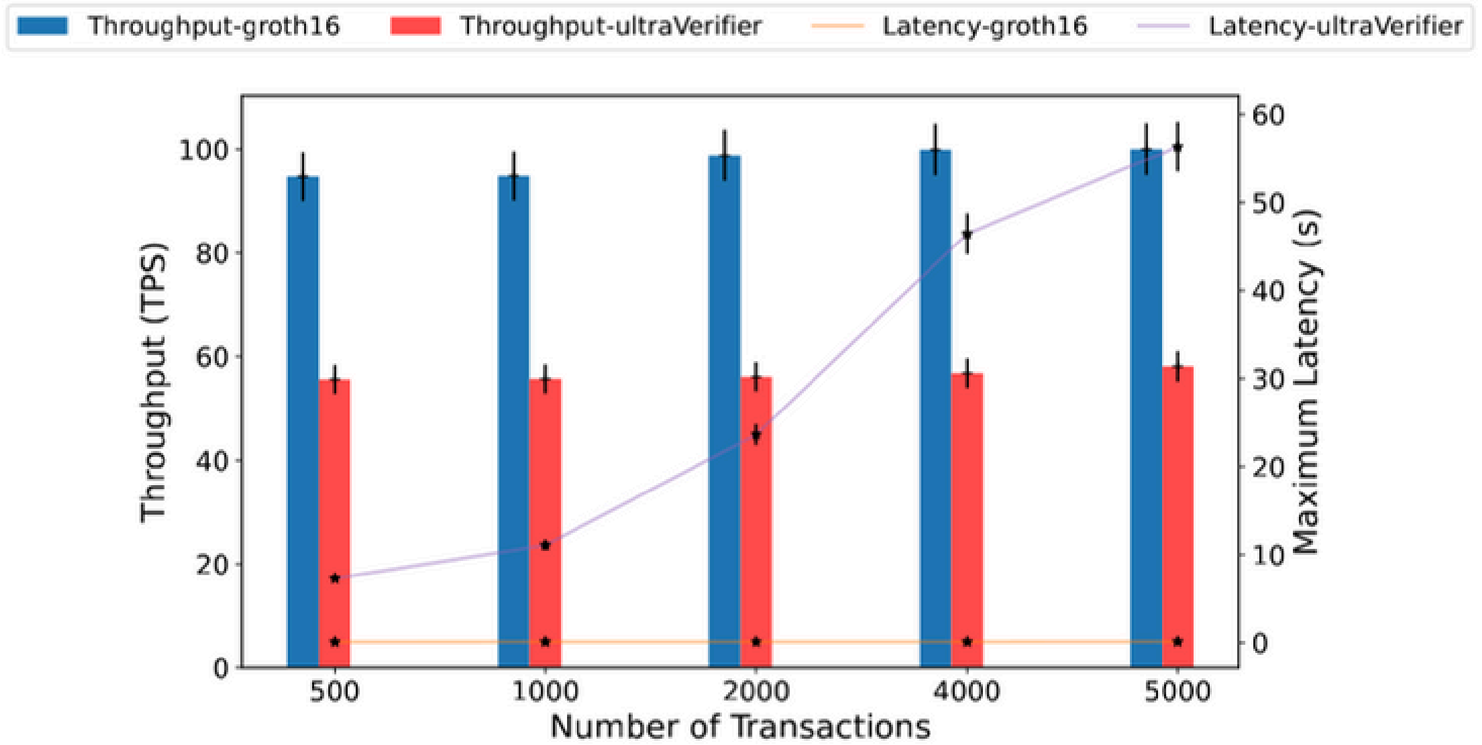
Tests & Results



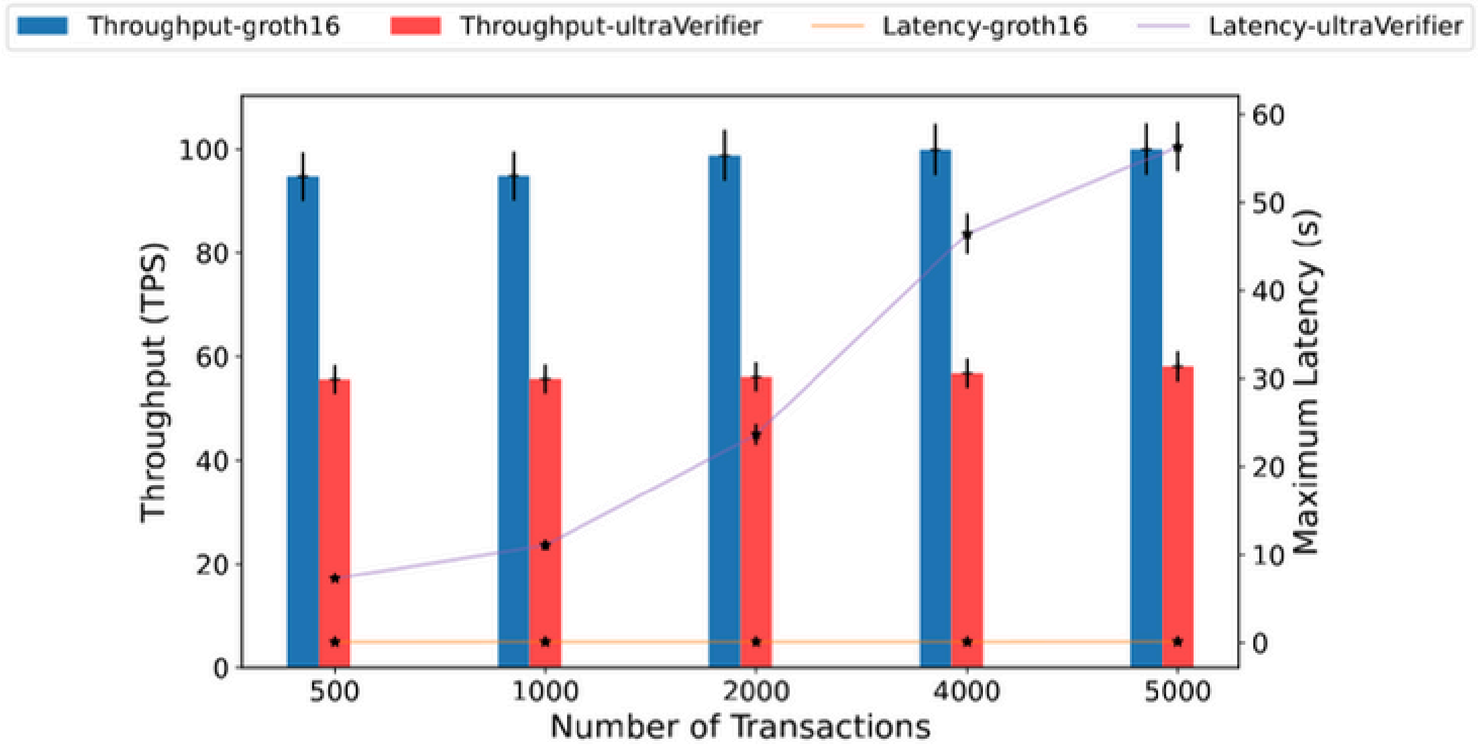
Tests & Results



Tests & Results



Tests & Results



Groth16 is the optimal zk proof system to use

Tests & Results

Benchmark results comparing Dual Layer and Single Layer transaction costs of setteleing roaming session

Total TXs	Dual Layer 2						Single Layer (L1)
	Batches	Batch size	Commit	Prove	Execute	Total	
60	1	60	205,907	83,676	99,166	388,749	15,636,180
100	2	50	411,802	167,328	182,794	761,924	26,060,300
200	3	67	617,625	250,980	297,486	1,166,091	52,120,600
500	7	72	1,458,233	585,588	694,162	2,737,983	130,301,500

Tests & Results

Benchmark results comparing Dual Layer and Single Layer transaction costs of setteleing roaming session

Total TXs	Dual Layer 2						Single Layer (L1)
	Batches	Batch size	Commit	Prove	Execute	Total	
60	1	60	205,907	83,676	99,166	388,749	15,636,180
100	2	50	411,802	167,328	182,794	761,924	26,060,300
200	3	67	617,625	250,980	297,486	1,166,091	52,120,600
500	7	72	1,458,233	585,588	694,162	2,737,983	130,301,500

97% reduction!

TPS = 8640 tx/s!

Conclusion

- Completely decentralized & Trustless system
- Backward-compatible with existing 5G architecture
- Correct roaming cost calculation proofs without disclosing any sensitive data
- Scalable to keep up with the high load of 5G environment

Tests & Results

B5GRoam: A Zero Trust Framework for Secure and Efficient On-Chain B5G Roaming

Mohamed Abdessamed Rezazi[1,2], Mouhamed Amine Bouchiha[1], Ahmed Mounsf Rafik Bendada[1], and Yacine Ghamri-Doudane[1]

[1]*L3i, La Rochelle University, La Rochelle, France*
[2]*Ecole Nationale Supérieure d'Informatique, Algiers, Algeria*

***Abstract*—Roaming settlement in 5G and beyond networks demands secure, efficient, and trustworthy mechanisms for billing reconciliation between mobile operators. While blockchain promises decentralization and auditability, existing solutions suffer from critical limitations—namely, data privacy risks, assumptions of mutual trust, and scalability bottlenecks. To address these challenges, we present B5GRoam, a novel on-chain and zero-trust framework for secure, privacy-preserving, and scalable roaming settlements. B5GRoam introduces a cryptographically verifiable call detail record (CDR) submission protocol, enabling smart contracts to authenticate usage claims without exposing sensitive data. To preserve privacy, we integrate non-interactive zero-knowledge proofs (zkSNARKs) that allow on-chain verification of roaming activity without revealing user or network details. To meet the high-throughput demands of 5G environments, B5GRoam leverages Layer 2 zk-Rollups, significantly reducing**

latency and cost overhead and also represent single points of failure and trust. Moreover, the lack of transparency and real-time responsiveness impedes fraud detection and resolution. In this evolving landscape, decentralizing roaming services presents a compelling alternative. This aims to enable automated, transparent, and tamper-resistant coordination between MNOs and other stakeholders without the need for centralized oversight [3].

Blockchain technology has emerged as a powerful tool for enabling decentralization, offering a tamper-proof and transparent method for recording transactions [4], [5]. In the context of 5G roaming, blockchain has the potential to remove intermediaries, reduce fraud, and improve settlement

The paper was submitted and accepted at **the 2025 IEEE Global Communication Conference: Communication & Information Systems Security,** to be presented at Taiwan this December.

Perspectives

- Aggregating proofs to settle multiple roaming sessions at once

Perspectives

- Aggregating proofs to settle multiple roaming sessions at once

- Currently, we suppose that the UE is equipped with a TEE that calculates the CDR commitment, we can develop that or at least have an MVP

Perspectives

- Aggregating proofs to settle multiple roaming sessions at once
- Currently, we suppose that the UE is equipped with a TEE that calculates the CDR commitment, we can develop that or at least have an MVP
- Currently, the HMNO is required to lock a maximum amount of money for the roaming session before it even begins. ML models can be developed to predict the precise amount.

DEMO

21 September, 2025

Demo plan

- Running the Groth16 trusted setup (generating prover & verifier keys)
- Bootstraping a local blockchain with 4 nodes, using geth
- End-to-end on-chain flow simulation

    1. Creating a new roaming agreement
    2. Starting a new roaming session
    3. Submitting the CDR by the UE
    4. Generating a proof of the total cost calculation
    5. Setteling the roaming session by submitting the total & ZK proof by VMNO

Appendix - ZKP Generation

$$
\begin{aligned}
T_{sms} &= n_{sms} \times r_{sms} \\
T_{mb} &= n_{mb} \times r_{mb} \\
T_{voice} &= n_{voice} \times r_{voice} \\
T_{computed} &= T_{sms} + T_{mb} + T_{voice} \\
T_{computed} &== T
\end{aligned}
$$

$$a = [1, \quad r_{sms}, \quad r_{mb}, \quad r_{voice}, \quad T, \quad n_{sms}, \quad n_{mb}, \quad n_{voice}, \quad T_{sms}, \quad T_{mb}, \quad T_{voice}, \quad T_{computed}]$$

$$La \times Ra = Oa$$

$$\mathbf{La} \rightarrow \sum_{i=1}^{m} a_i u_i(x) = u(x)$$

$$\mathbf{Ra} \rightarrow \sum_{i=1}^{m} a_i v_i(x) = v(x)$$

$$\mathbf{Oa} \rightarrow \sum_{i=1}^{m} a_i w_i(x) = w(x)$$

Appendix - ZKP Generation

$$\sum_{i=1}^{m} a_i u_i(x) \sum_{i=1}^{m} a_i v_i(x) = \sum_{i=1}^{m} a_i w_i(x) + h(x)t(x)$$

where

$$t(x) = (x-1)(x-2)\dots(x-n)$$

$$\begin{aligned}
[\Omega_{n-1}, \Omega_{n-2}, \dots, \Omega_2, \Omega_1, G_1] &= [\tau^n G_1, \tau^{n-1} G_1, \dots, \tau G_1, G_1] \\
[\Theta_{n-1}, \Theta_{n-2}, \dots, \Theta_2, \Theta_2, G_2] &= [\tau^n G_2, \tau^{n-1} G_2, \dots, \tau G_2, G_2] \\
[\Upsilon_{n-2}, \Upsilon_{n-3}, \dots, \Upsilon_1, \Upsilon_0] &= [\tau^{n-2} t(\tau) G_1, \tau^{n-3} t(\tau) G_1, \dots, \tau t(\tau) G_1, t(\tau) G_1]
\end{aligned}$$

$$f(\tau) = \sum_{i=1}^{d} f_i \Omega_i = \langle [f_d, f_{d-1}, \dots, f_1, f_0], [\Omega_d, \Omega_{d-1}, \dots, G_1] \rangle$$

Appendix - ZKP Generation

$$\underbrace{([\alpha]_1 + \sum_{i=1}^{m} a_i u_i(\tau))([\beta]_2 + \sum_{i=1}^{m} a_i v_i(\tau))}_{[A]_1 \quad [B]_2} = [\alpha]_1 \bullet [\beta]_2 + \underbrace{(\alpha \sum_{i=1}^{m} a_i v_i(\tau) + \beta \sum_{i=1}^{m} a_i u_i(\tau) + \sum_{i=1}^{m} a_i w_i(\tau) + h(\tau)t(\tau))}_{[C]_1} \bullet G_2$$

Appendix - ZKP Generation

$$\underbrace{([\alpha]_1 + \sum_{i=1}^{m} a_i u_i(\tau))([\beta]_2 + \sum_{i=1}^{m} a_i v_i(\tau))}_{[A]_1 \quad [B]_2} = [\alpha]_1 \bullet [\beta]_2 + \underbrace{(\alpha \sum_{i=1}^{m} a_i v_i(\tau) + \beta \sum_{i=1}^{m} a_i u_i(\tau) + \sum_{i=1}^{m} a_i w_i(\tau) + h(\tau)t(\tau))}_{[C]_1} \bullet G_2$$

Trusted setup calculation

$$\begin{aligned}
[\Psi_1]_1 &= (\alpha v_1(\tau) + \beta u_1(\tau) + w_1(\tau))G_1 \\
[\Psi_2]_1 &= (\alpha v_2(\tau) + \beta u_2(\tau) + w_2(\tau))G_1 \\
&\vdots \\
[\Psi_m]_1 &= (\alpha v_m(\tau) + \beta u_m(\tau) + w_m(\tau))G_1
\end{aligned}$$

Appendix - ZKP Generation

$$
\begin{aligned}
\alpha, \beta, \tau, \gamma, \delta &\leftarrow \text{random scalars} \\
[\tau^{n-1}G_1, \tau^{n-2}G_1, \ldots, \tau G_1, G_1] &\leftarrow \text{srs for } \mathbb{G}_1 \\
[\tau^{n-1}G_2, \tau^{n-2}G_2, \ldots, \tau G_2, G_2] &\leftarrow \text{srs for } \mathbb{G}_2 \\
[\frac{\tau^{n-2}t(\tau)}{\delta}, \frac{\tau^{n-3}t(\tau)}{\delta}, \ldots, \frac{\tau t(\tau)}{\delta}, \frac{t(\tau)}{\delta}] &\leftarrow \text{srs for } h(\tau)t(\tau)
\end{aligned}
$$

public portion of the witness

$$
\begin{aligned}
[\Psi_1]_1 &= \frac{\alpha v_1(\tau) + \beta u_1(\tau) + w_1(\tau)}{\gamma} G_1 \\
[\Psi_2]_1 &= \frac{\alpha v_2(\tau) + \beta u_2(\tau) + w_2(\tau)}{\gamma} G_1 \\
&\vdots \\
[\Psi_\ell]_1 &= \frac{\alpha v_\ell(\tau) + \beta u_\ell(\tau) + w_\ell(\tau)}{\gamma} G_1
\end{aligned}
$$

private portion of the witness

$$
\begin{aligned}
[\Psi_{\ell+1}]_1 &= \frac{\alpha v_{\ell+1}(\tau) + \beta u_{\ell+1}(\tau) + w_{\ell+1}(\tau)}{\delta} G_1 \\
[\Psi_{\ell+2}]_1 &= \frac{\alpha v_{\ell+2}(\tau) + \beta u_{\ell+2}(\tau) + w_{\ell+2}(\tau)}{\delta} G_1 \\
&\vdots \\
[\Psi_m]_1 &= \frac{\alpha v_m(\tau) + \beta u_m(\tau) + w_m(\tau)}{\delta} G_1
\end{aligned}
$$

The trusted setup publishes

$$
([\alpha]_1, [\beta]_2, [\gamma]_2, [\delta]_2, \text{srs}_{G_1}, \text{srs}_{G_2}, [\Psi_1]_1, [\Psi_2]_1, \ldots, [\Psi_m]_1)
$$

Appendix - ZKP Generation

$$\alpha, \beta, \tau, \gamma, \delta \leftarrow \text{random scalars}$$

$$[\tau^{n-1}G_1, \tau^{n-2}G_1, \ldots, \tau G_1, G_1] \leftarrow \text{srs for } \mathbb{G}_1$$

$$[\tau^{n-1}G_2, \tau^{n-2}G_2, \ldots, \tau G_2, G_2] \leftarrow \text{srs for } \mathbb{G}_2$$

$$[\frac{\tau^{n-2}t(\tau)}{\delta}, \frac{\tau^{n-3}t(\tau)}{\delta}, \ldots, \frac{\tau t(\tau)}{\delta}, \frac{t(\tau)}{\delta}] \leftarrow \text{srs for } h(\tau)t(\tau)$$

public portion of the witness

$$[\Psi_1]_1 = \frac{\alpha v_1(\tau) + \beta u_1(\tau) + w_1(\tau)}{\gamma} G_1$$

$$[\Psi_2]_1 = \frac{\alpha v_2(\tau) + \beta u_2(\tau) + w_2(\tau)}{\gamma} G_1$$

$$\vdots$$

$$[\Psi_\ell]_1 = \frac{\alpha v_\ell(\tau) + \beta u_\ell(\tau) + w_\ell(\tau)}{\gamma} G_1$$

private portion of the witness

$$[\Psi_{\ell+1}]_1 = \frac{\alpha v_{\ell+1}(\tau) + \beta u_{\ell+1}(\tau) + w_{\ell+1}(\tau)}{\delta} G_1$$

$$[\Psi_{\ell+2}]_1 = \frac{\alpha v_{\ell+2}(\tau) + \beta u_{\ell+2}(\tau) + w_{\ell+2}(\tau)}{\delta} G_1$$

$$\vdots$$

$$[\Psi_m]_1 = \frac{\alpha v_m(\tau) + \beta u_m(\tau) + w_m(\tau)}{\delta} G_1$$

Prover key

The trusted setup publishes

$$([\alpha]_1, [\beta]_2, [\gamma]_2, [\delta]_2, \text{srs}_{G_1}, \text{srs}_{G_2}, [\Psi_1]_1, [\Psi_2]_1, \ldots, [\Psi_m]_1)$$

Appendix - ZKP Generation

$$\alpha, \beta, \tau, \gamma, \delta \leftarrow \text{random scalars}$$
$$[\tau^{n-1}G_1, \tau^{n-2}G_1, \ldots, \tau G_1, G_1] \leftarrow \text{srs for } \mathbb{G}_1$$
$$[\tau^{n-1}G_2, \tau^{n-2}G_2, \ldots, \tau G_2, G_2] \leftarrow \text{srs for } \mathbb{G}_2$$
$$[\frac{\tau^{n-2}t(\tau)}{\delta}, \frac{\tau^{n-3}t(\tau)}{\delta}, \ldots, \frac{\tau t(\tau)}{\delta}, \frac{t(\tau)}{\delta}] \leftarrow \text{srs for } h(\tau)t(\tau)$$

public portion of the witness

$$[\Psi_1]_1 = \frac{\alpha v_1(\tau) + \beta u_1(\tau) + w_1(\tau)}{\gamma} G_1$$
$$[\Psi_2]_1 = \frac{\alpha v_2(\tau) + \beta u_2(\tau) + w_2(\tau)}{\gamma} G_1$$
$$\vdots$$
$$[\Psi_\ell]_1 = \frac{\alpha v_\ell(\tau) + \beta u_\ell(\tau) + w_\ell(\tau)}{\gamma} G_1$$

private portion of the witness

$$[\Psi_{\ell+1}]_1 = \frac{\alpha v_{\ell+1}(\tau) + \beta u_{\ell+1}(\tau) + w_{\ell+1}(\tau)}{\delta} G_1$$
$$[\Psi_{\ell+2}]_1 = \frac{\alpha v_{\ell+2}(\tau) + \beta u_{\ell+2}(\tau) + w_{\ell+2}(\tau)}{\delta} G_1$$
$$\vdots$$
$$[\Psi_m]_1 = \frac{\alpha v_m(\tau) + \beta u_m(\tau) + w_m(\tau)}{\delta} G_1$$

Verifier key

The trusted setup publishes

$$([\alpha]_1, [\beta]_2, [\gamma]_2, [\delta]_2, \text{srs}_{G_1}, \text{srs}_{G_2}, [\Psi_1]_1, [\Psi_2]_1, \ldots, [\Psi_m]_1)$$

Appendix - ZKP Generation

$$\alpha, \beta, \tau, \gamma, \delta \leftarrow \text{random scalars}$$
$$[\tau^{n-1}G_1, \tau^{n-2}G_1, \ldots, \tau G_1, G_1] \leftarrow \text{srs for } \mathbb{G}_1$$
$$[\tau^{n-1}G_2, \tau^{n-2}G_2, \ldots, \tau G_2, G_2] \leftarrow \text{srs for } \mathbb{G}_2$$
$$[\frac{\tau^{n-2}t(\tau)}{\delta}, \frac{\tau^{n-3}t(\tau)}{\delta}, \ldots, \frac{\tau t(\tau)}{\delta}, \frac{t(\tau)}{\delta}] \leftarrow \text{srs for } h(\tau)t(\tau)$$

public portion of the witness

$$[\Psi_1]_1 = \frac{\alpha v_1(\tau) + \beta u_1(\tau) + w_1(\tau)}{\gamma} G_1$$
$$[\Psi_2]_1 = \frac{\alpha v_2(\tau) + \beta u_2(\tau) + w_2(\tau)}{\gamma} G_1$$
$$\vdots$$
$$[\Psi_\ell]_1 = \frac{\alpha v_\ell(\tau) + \beta u_\ell(\tau) + w_\ell(\tau)}{\gamma} G_1$$

private portion of the witness

$$[\Psi_{\ell+1}]_1 = \frac{\alpha v_{\ell+1}(\tau) + \beta u_{\ell+1}(\tau) + w_{\ell+1}(\tau)}{\delta} G_1$$
$$[\Psi_{\ell+2}]_1 = \frac{\alpha v_{\ell+2}(\tau) + \beta u_{\ell+2}(\tau) + w_{\ell+2}(\tau)}{\delta} G_1$$
$$\vdots$$
$$[\Psi_m]_1 = \frac{\alpha v_m(\tau) + \beta u_m(\tau) + w_m(\tau)}{\delta} G_1$$

The trusted setup publishes

$$([\alpha]_1, [\beta]_2, [\gamma]_2, [\delta]_2, \text{srs}_{G_1}, \text{srs}_{G_2}, [\Psi_1]_1, [\Psi_2]_1, \ldots, [\Psi_m]_1)$$

The prover (VMNO) calculates:

$$[A]_1 = [\alpha]_1 + \sum_{i=1}^{m} a_i u_i(\tau)$$
$$[B]_2 = [\beta]_2 + \sum_{i=1}^{m} a_i v_i(\tau)$$
$$[C]_1 = \sum_{i=\ell+1}^{m} a_i [\Psi_i]_1 + h(\tau)t(\tau)$$

Appendix - ZKP Generation

$$\alpha, \beta, \tau, \gamma, \delta \leftarrow \text{random scalars}$$
$$[\tau^{n-1}G_1, \tau^{n-2}G_1, \ldots, \tau G_1, G_1] \leftarrow \text{srs for } \mathbb{G}_1$$
$$[\tau^{n-1}G_2, \tau^{n-2}G_2, \ldots, \tau G_2, G_2] \leftarrow \text{srs for } \mathbb{G}_2$$
$$[\frac{\tau^{n-2}t(\tau)}{\delta}, \frac{\tau^{n-3}t(\tau)}{\delta}, \ldots, \frac{\tau t(\tau)}{\delta}, \frac{t(\tau)}{\delta}] \leftarrow \text{srs for } h(\tau)t(\tau)$$

public portion of the witness

$$[\Psi_1]_1 = \frac{\alpha v_1(\tau) + \beta u_1(\tau) + w_1(\tau)}{\gamma} G_1$$
$$[\Psi_2]_1 = \frac{\alpha v_2(\tau) + \beta u_2(\tau) + w_2(\tau)}{\gamma} G_1$$
$$\vdots$$
$$[\Psi_\ell]_1 = \frac{\alpha v_\ell(\tau) + \beta u_\ell(\tau) + w_\ell(\tau)}{\gamma} G_1$$

private portion of the witness

$$[\Psi_{\ell+1}]_1 = \frac{\alpha v_{\ell+1}(\tau) + \beta u_{\ell+1}(\tau) + w_{\ell+1}(\tau)}{\delta} G_1$$
$$[\Psi_{\ell+2}]_1 = \frac{\alpha v_{\ell+2}(\tau) + \beta u_{\ell+2}(\tau) + w_{\ell+2}(\tau)}{\delta} G_1$$
$$\vdots$$
$$[\Psi_m]_1 = \frac{\alpha v_m(\tau) + \beta u_m(\tau) + w_m(\tau)}{\delta} G_1$$

The trusted setup publishes

$$([\alpha]_1, [\beta]_2, [\gamma]_2, [\delta]_2, \text{srs}_{G_1}, \text{srs}_{G_2}, [\Psi_1]_1, [\Psi_2]_1, \ldots, [\Psi_m]_1)$$

The prover (VMNO) calculates:

$$[A]_1 = [\alpha]_1 + \sum_{i=1}^{m} a_i u_i(\tau)$$
$$[B]_2 = [\beta]_2 + \sum_{i=1}^{m} a_i v_i(\tau)$$
$$[C]_1 = \sum_{i=\ell+1}^{m} a_i [\Psi_i]_1 + h(\tau)t(\tau)$$

The verifier (smart contract) computes the public portion by itself:

$$[X]_1 = \sum_{i=1}^{\ell} a_i \Psi_i$$

The verifier finaly verifies that:

$$[A]_1 \bullet [B]_2 \stackrel{?}{=} [\alpha]_1 \bullet [\beta]_2 + [X]_1 \bullet [\gamma]_2 + [C]_1 \bullet [\delta]_2$$