KnoxNet

The First Dual-Domain Layer 1 Privacy Network with Homomorphic Encryption

Offline & Online Execution • Homomorphic Settlement • Connectivity-Agnostic Execution

Abstract

Privacy in digital payment systems remains fragile despite advances in cryptography. Even privacy-focused blockchains require continuous internet connectivity and global transaction broadcast, leaking metadata such as timing, frequency, and network-level correlations through infrastructure including ISPs, RPC providers, and network observers. As a result, transaction privacy in always-online systems is probabilistic and vulnerable to long-term surveillance.

We present **KnoxNet**, an offline-first digital payment network designed to enable private value transfer without continuous internet connectivity. KnoxNet decouples transaction execution from global settlement by allowing users to execute transactions locally using cryptographically bounded offline value, while deferring reconciliation to an online anchor ledger. Correctness is enforced through deterministic fraud proofs and escrow-based economic penalties rather than real-time global consensus.

The system supports lightweight peer-to-peer transaction execution in offline environments and a scalable reconciliation process that resolves conflicts and enforces global supply safety once connectivity is restored. To operate under adversarial conditions, KnoxNet incorporates cryptographic value bounding, optional witness attestations, and economic containment mechanisms that limit the impact of misbehavior without requiring continuous surveillance.

Finally, we characterize the properties of an ideal private digital payment system—emphasizing reduced observability, bounded risk, and eventual correctness—and use this model to motivate KnoxNet's architectural design. Our results demonstrate that durable digital privacy requires architectural separation of execution and settlement, complementing cryptographic privacy techniques rather than relying on them alone.

Table of Contents

1. Introduction

Digital money has become inseparable from global connectivity. Whether interacting with a bank, a blockchain, or a privacy-focused protocol, users implicitly accept that transferring value requires continuous internet access and immediate interaction with globally visible infrastructure. Transactions are expected to be broadcast, routed, validated, and recorded in real time.

Historically, money functioned very differently. Physical cash could be exchanged directly between participants, privately and locally. No third party observed the transaction, no timestamps were required, and no permanent global record was created. Privacy was not an added feature—it was a default property of how value moved.

Modern digital systems invert this model. Value transfer is no longer a local act between participants; it is a globally observable event. Even when transaction contents are encrypted, the act of transacting itself remains visible to networks, intermediaries, and infrastructure providers. As a result, privacy erosion occurs not only through what is revealed, but through the requirement that activity be continuously observable.

Many privacy-focused digital payment systems approach this problem primarily as one of cryptographic secrecy. Encryption, commitments, and zero-knowledge proofs are used to conceal senders, receivers, and amounts. However, privacy loss often occurs before cryptography is applied—at the network layer itself. Continuous connectivity, global broadcast, and real-time verification expose metadata such as timing, frequency, and routing behavior, which can be correlated over time to infer user activity without breaking any cryptographic guarantees.

KnoxNet begins from a different premise.

Privacy is not something to be added on top of the internet.
Privacy emerges when the internet is no longer a mandatory requirement for execution.

Rather than attempting to hide transactions within an always-online system, KnoxNet changes the conditions under which transactions occur. KnoxNet is the first blockchain system designed to **exist natively in two co-equal states**: an **execution state** and a **settlement state**.

In KnoxNet, transaction execution is **connectivity-agnostic**. Value can be transferred either offline or online between participants without requiring immediate interaction with global infrastructure or real-time consensus. Execution is a local act, validated at the moment it occurs using cryptographic authorization and value conservation rules. Transactions are real and effective at execution time, regardless of network connectivity.

Global settlement is handled separately in an online state. When connectivity is available, executed transactions are reconciled with an anchor ledger that enforces supply safety, resolves conflicts, and applies penalties for misbehavior. Importantly, settlement does not replay or re-execute transactions. Instead, correctness is enforced through encrypted settlement constraints and deterministic fraud proofs, minimizing disclosure while preserving global verifiability.

By separating execution from settlement as distinct ledger states, KnoxNet reduces observability at the moment value moves while retaining strong guarantees of correctness and accountability. Privacy is preserved not by hiding activity within a globally observable system, but by allowing digital value to exist and move meaningfully **outside** that system until global coordination is strictly necessary.

---

2. Why Privacy Is a Myth in Always-Online Systems

Privacy failures in digital money systems rarely result from broken cryptography. Modern cryptographic primitives are mathematically robust and, when used correctly, can effectively conceal transaction contents such as senders, receivers, and amounts. In practice, privacy erosion arises elsewhere.

Always-online systems require transactions to be broadcast, routed, and validated across globally observable communication networks. As a consequence, every online transaction reveals information beyond its encrypted payload. At a minimum, it reveals that a transaction occurred, the approximate time of occurrence, and the frequency of a user's activity. Network propagation behavior further exposes how transactions move through the system, allowing correlations to be drawn between identity, connectivity, and geographic location.

This form of leakage is not accidental; it is intrinsic to globally observable systems. Even when transaction contents are protected using encryption or zero-knowledge proofs, the surrounding metadata remains visible to infrastructure outside the user's control, including internet service providers, RPC endpoints, relayers, and passive network observers. Over time, these signals can be aggregated to reconstruct behavioral patterns without violating any cryptographic guarantees.

Cryptography can hide *what* happened.
It cannot hide *that* something happened.

As long as transaction execution depends on continuous internet connectivity and real-time global verification, privacy remains probabilistic rather than absolute. It relies on assumptions

about adversarial limitations, network noise, and imperfect observation—none of which provide durable guarantees. In such systems, privacy can be weakened simply by observing the network, without breaking encryption or compromising cryptographic primitives.

KnoxNet: An Offline-First Approach to Private Digital Payments

The limitations described above suggest that privacy failures in digital money are not primarily caused by weak cryptography, but by architectural dependence on continuous connectivity and global observability. If transaction execution must always occur online and be immediately visible to shared infrastructure, then privacy can only ever be partial.

KnoxNet is designed in response to this observation. Rather than attempting to hide transactions within an always-online system, KnoxNet changes the conditions under which transactions occur. The core idea is simple: **remove the internet from the execution path of value transfer**, and minimize what must be revealed when global coordination becomes necessary.

At its core, KnoxNet is an **offline-first Layer 1 digital payment network**. Transaction execution does not require continuous internet access, interaction with global infrastructure, or real-time consensus. Users can transfer value locally using short-range or peer-to-peer communication, with transactions validated at the time of execution using cryptographic signatures and value conservation rules. Offline execution is treated as a first-class system primitive, not as a fallback mode or temporary exception.

Because offline execution cannot rely on immediate global state updates, KnoxNet separates **execution** from **settlement**. Transactions are executed locally and recorded as part of an offline history. Global correctness is enforced later, when connectivity is available, through reconciliation with an online anchor ledger. This ledger does not replay or inspect full transaction histories. Instead, it verifies that global invariants have been preserved and that no contradictions, such as double-spends, have occurred.

To further reduce disclosure during reconciliation, KnoxNet minimizes the information exposed at the settlement layer. Rather than processing plaintext transaction flows, the system enforces correctness over encrypted data. Homomorphic encryption is used to validate supply invariants, value conservation, issuance limits, and settlement deltas without revealing individual transaction amounts or sensitive accounting details. In this way, observability is reduced both during execution and during settlement.

By decoupling execution from settlement, KnoxNet limits continuous network-level observability at the moment value is transferred. By encrypting the settlement surface, it further restricts what can be learned when global verification occurs. Together, these choices allow KnoxNet to preserve global supply safety, bounded risk, and eventual correctness while addressing privacy loss at both the execution layer and the settlement layer.

---

3. Design Philosophy and Core Principles

KnoxNet is built around a **Dual-Domain Ledger Architecture**, a deliberate departure from the single, always-online execution models used by most blockchains. Rather than attempting to enforce execution, verification, and settlement within one globally observable system, KnoxNet separates these responsibilities across two tightly coupled domains: an offline execution domain and an encrypted online settlement domain.

This design reflects a fundamental observation: privacy and correctness place different and often conflicting demands on system architecture. Privacy is most vulnerable at the moment value is transferred, while correctness and supply safety require global coordination. KnoxNet addresses this tension by assigning each requirement to the domain best suited to enforce it.

3.1 Dual-Domain Ledger Architecture

The Dual-Domain Ledger Architecture divides the system into two interdependent domains:

- an **offline execution domain**, where transactions are executed locally between participants, and
- an **online settlement domain**, where executed transactions are reconciled and global invariants are enforced.

These domains are not alternative operating modes. Both are essential, and neither is sufficient on its own. Offline execution enables private, local value transfer without continuous connectivity, while online settlement ensures global correctness, supply safety, and conflict resolution. KnoxNet is designed to operate across both domains simultaneously, using each to compensate for the limitations of the other.

3.2 Offline-First Execution Domain

Transaction execution in KnoxNet is **offline-first**. Value transfers occur locally between participants using peer-to-peer communication, without requiring internet access, global broadcast, or real-time consensus. Execution involves validating cryptographic ownership, signatures, and value conservation, but does not update global state at the time the transaction occurs.

Offline execution is treated as a first-class system primitive rather than a fallback mechanism. By removing the internet from the execution path, KnoxNet minimizes network-level observability at the moment value is exchanged. Transactions that are never broadcast cannot be continuously monitored, logged, or correlated by global infrastructure.

3.3 Decoupling Execution from Settlement

A core principle of KnoxNet is the explicit decoupling of **execution** and **settlement**.

Execution refers to the act of transferring value between participants at the time a transaction occurs. Settlement refers to the later process of reconciling executed transactions with global state, enforcing system-wide constraints, and resolving conflicts.

By separating these processes, KnoxNet avoids forcing every transaction into a globally observable and contested environment. Execution can remain local and private, while settlement enforces correctness without requiring real-time visibility into transaction activity.

At the boundary between these domains, value exists in a dual state: locally executed yet globally unsettled. Only during reconciliation is this state resolved.

3.4 Encrypted Online Settlement Domain

The online settlement domain in KnoxNet is designed to enforce global correctness without reintroducing transparency. Unlike conventional blockchains that process plaintext transaction flows, KnoxNet verifies system-wide constraints over encrypted data.

Homomorphic encryption is used to validate global invariants such as value conservation, issuance limits, and settlement deltas without revealing individual transaction amounts or sensitive accounting details. This allows the system to maintain global verifiability while minimizing information leakage during settlement.

Encrypted settlement is not an auxiliary feature, but a core component of the Dual-Domain Ledger Architecture. It ensures that the privacy gains achieved during offline execution are not undone when global coordination becomes necessary.

3.5 Economic Enforcement Over Real-Time Prevention

KnoxNet does not attempt to prevent all misbehavior at the moment it occurs. Enforcing perfect real-time prevention would require continuous monitoring, trusted infrastructure, or invasive validation, all of which undermine privacy.

Instead, KnoxNet relies on **economic enforcement**. Misbehavior such as double-spending or contradictory transaction histories is inevitably detectable during settlement. When detected, penalties are applied automatically through escrow-based mechanisms, limiting the impact of abuse while preserving privacy during execution.

Correctness is enforced over time rather than instantaneously, allowing KnoxNet to balance strong privacy guarantees with global system integrity.

---

4. System Overview and Architecture

KnoxNet is structured as a **three-layer system** that implements the Dual-Domain Ledger Architecture. Each layer serves a distinct role in enabling private execution, encrypted global verification, and eventual correctness. The layers are designed to operate independently while enforcing shared system invariants.

At a high level, KnoxNet consists of:

1. an **Offline Execution Domain**, where transactions are executed locally without internet connectivity,

2. a **KnoxNet Layer 1 Ledger**, which enforces global correctness and supply safety, and

3. a **Reconciliation Layer**, which bridges offline execution with encrypted online settlement.

4.1 Offline Execution Domain ($\mathcal{D}$)

The **Offline Execution Domain ($\mathcal{D}$)** is where transaction execution occurs. Devices transact directly with one another using local or short-range communication mechanisms such as Bluetooth, Wi-Fi Direct, or local mesh networks. Transactions are validated at the time of execution using cryptographic signatures and value conservation rules, without interaction with global infrastructure.

No global coordination is required during execution. Transactions are not broadcast, routed through public networks, or submitted to shared ledgers at the moment value is transferred. As

a result, execution does not generate continuous network-level metadata and is resistant to global surveillance.

The offline domain is designed to be permissive with respect to connectivity and environment. Devices may operate in fully offline settings, intermittently connected settings, or constrained network environments. Execution correctness is enforced locally, while global correctness is deferred.

4.2 KnoxNet Layer 1 Ledger ($\mathcal{L}_1$)

The **KnoxNet Layer 1 Ledger ($\mathcal{L}_1$)** is a custom blockchain optimized for **encrypted verification and economic enforcement**, rather than for replaying every transaction in plaintext. It serves as the system's global coordination and enforcement layer.

The Layer 1 ledger is responsible for:

- escrowing value and issuing offline spendable units,
- registering and enforcing uniqueness of notes or value identifiers,
- verifying reconciliation submissions,
- detecting contradictions such as double-spends through deterministic fraud proofs,
- applying penalties and slashing when misbehavior is detected, and
- enforcing global supply and issuance invariants.

Unlike conventional blockchains, $\mathcal{L}_1$ does not process full transaction histories or observe execution in real time. Its primary role is to verify that system-wide constraints have been preserved once offline activity is reconciled. To minimize information leakage, the ledger is designed to validate encrypted settlement constraints rather than plaintext transaction flows.

4.3 Reconciliation Layer ($\mathcal{R}$)

The **Reconciliation Layer ($\mathcal{R}$)** connects the offline execution domain with the KnoxNet Layer 1 ledger. It is responsible for transforming offline transaction histories into settlement updates that can be verified by the ledger.

When connectivity is available, devices submit reconciliation payloads derived from offline activity. These payloads do not reveal full transaction histories. Instead, they encode the information necessary to:

- demonstrate value conservation,
- update ownership state,
- prove uniqueness of consumed value units, and
- expose contradictions if conflicting histories exist.

The reconciliation layer enforces correctness by ensuring that all submitted updates satisfy global invariants. If contradictions are detected—such as the same value unit appearing in multiple histories—fraud proofs are generated deterministically and penalties are applied by the Layer 1 ledger.

By separating reconciliation from both execution and ledger enforcement, KnoxNet avoids forcing offline activity into immediate global visibility while still guaranteeing eventual correctness and accountability.

5. Core Objects and Cryptographic Primitives

This section defines the minimal set of identities and cryptographic keys used by the KnoxNet protocol. These primitives underpin authorization, offline value issuance, and encrypted settlement across both execution and settlement domains.

5.1 Identities

Each user uuu is associated with a signing keypair:

(sku,pku)(sk_u, pk_u)(sku,pku)

which is used to authorize ownership and transfer of offline notes, as well as reconciliation submissions.

KnoxNet optionally supports **device-scoped keypairs**:

(skdev,pkdev)(sk_{dev}, pk_{dev})(skdev,pkdev)

used to enforce device-level policies such as issuance caps, exposure limits, and recovery mechanisms. Device identities are subordinate to user identities and do not represent independent economic actors.

5.2 Anchor Ledger Keys

The KnoxNet Layer 1 ledger maintains protocol-level cryptographic keys.

A signing key:

skL1sk_{\mathcal{L}_1}skL1

is used to authorize the issuance of offline notes. Notes lacking a valid signature under this key are invalid.

For encrypted settlement, the ledger maintains a homomorphic encryption public key:

pkHEpk_{HE}pkHE

used to encrypt settlement data and verify global constraints without revealing plaintext values.

Decryption capability may optionally be distributed across a validator committee using threshold decryption to reduce trust concentration.

5.3 Cryptographic Assumptions

KnoxNet assumes the standard security properties of the cryptographic primitives it employs, including the unforgeability of digital signatures and the semantic security of encryption schemes. The protocol does not rely on trusted hardware or continuous connectivity.

Cryptographic primitives are combined with architectural design—offline execution and encrypted settlement—to reduce observability rather than relying on cryptography alone for privacy.

5. Offline Value Model

The core challenge of offline execution is enabling value transfer without continuous access to global state, while preserving global supply safety and preventing unbounded loss. KnoxNet addresses this challenge through a constrained offline value model in which all spendable offline value is derived from, and strictly bounded by, value escrowed on the Layer 1 ledger.

KnoxNet uses a native asset, denoted **$KNX**, as the unit of value within the system. All escrowed balances, offline value units, transactions, and settlement constraints are denominated in $KNX. No other asset can circulate within the KnoxNet execution or settlement domains.

Offline value in KnoxNet is not arbitrary and cannot be created locally. Every unit of $KNX that circulates in the offline execution domain originates from an on-chain escrow and is cryptographically authorized by the KnoxNet Layer 1 ledger. This design ensures that offline execution does not introduce inflation or violate global supply invariants.

5.1 Escrowed $KNX and Offline Issuance

To obtain spendable offline value, a user first escrows $KNX on the KnoxNet Layer 1 ledger. Escrowed $KNX is locked and cannot be spent, transferred, or otherwise used on-chain while corresponding offline value remains in circulation.

From this escrow, the ledger issues a set of **offline notes**, each representing a fixed denomination of $KNX. The total value of issued notes is strictly bounded by the amount of $KNX escrowed. At no point can the aggregate value of offline notes exceed the escrow backing them.

Once issued, offline notes may circulate freely within the offline execution domain without requiring further interaction with the ledger until reconciliation occurs.

5.2 Offline Notes

An **offline note** is a cryptographically authenticated object representing a fixed amount of $KNX that is spendable within the offline execution domain. Notes serve as the fundamental unit of offline value transfer.

Each offline note contains:

- a globally unique identifier,
- a denomination expressed in $KNX,
- an owner public key, and
- a cryptographic signature from the KnoxNet Layer 1 ledger authorizing its existence.

Notes are non-forgeable and cannot be created or modified by users. While they may be transferred offline, their global validity ultimately depends on reconciliation with the ledger, which enforces uniqueness and supply constraints.

Offline notes function as controlled bearer instruments: possession and authorization determine spendability during execution, while correctness is enforced later at settlement.

5.3 Offline Transfers and Local Value Conservation

Offline transactions consume one or more existing notes and produce a new set of notes whose total value equals the total value of the inputs. This enforces **local value conservation** at execution time.

During offline execution, participants verify:

- the authenticity of the ledger's signature on each input note,
- authorization by the current owner of the note, and
- conservation of $KNX within the transaction.

Global uniqueness of note identifiers is intentionally **not enforced** during offline execution. This design choice avoids requiring global coordination, trusted hardware, or continuous connectivity at the moment of value transfer.

5.4 Bounded Risk and Loss Containment

Because global uniqueness cannot be enforced offline, KnoxNet does not attempt to prevent all forms of double-spending at execution time. Instead, it bounds the potential impact of misbehavior.

The maximum value that can be misused offline is strictly limited by the amount of $KNX escrowed on the Layer 1 ledger. Any attempt to spend the same offline note in multiple conflicting histories is deterministically detectable during reconciliation.

When such contradictions are detected, penalties are applied by slashing the associated escrowed $KNX. This ensures that offline misbehavior is economically irrational beyond the value already locked and cannot lead to unbounded systemic loss.

This model transforms offline double-spending from a catastrophic failure into a bounded, punishable event.

5.5 Relationship to Reconciliation and Settlement

Offline notes and offline transactions are not final until they are reconciled with the KnoxNet Layer 1 ledger. During reconciliation, the ledger verifies that:

- each offline note identifier is consumed at most once,
- total $KNX value is conserved across all reconciled histories, and
- no issuance or supply limits have been violated.

If multiple reconciliation submissions attempt to redeem the same note identifier, a contradiction is exposed and a deterministic fraud proof is generated. The ledger resolves the conflict and applies penalties according to protocol rules.

Through this process, the offline value model preserves the privacy and flexibility of local execution while maintaining global correctness, supply safety, and economic accountability.

6. Offline Transaction Execution Protocol

This section describes how $KNX is transferred within the offline execution domain. The offline transaction execution protocol defines transaction structure, local validation rules, and optional witness attestations. The protocol is designed to enable private, local value transfer without continuous connectivity, while ensuring that all executed transactions remain reconcilable with the global ledger.

6.1 Transaction Structure

An offline transaction represents the transfer of ownership of one or more offline notes. Each transaction consumes a set of existing notes and produces a new set of notes whose total value is conserved.

An offline transaction consists of:

- a set of **input notes**,
- a set of **output notes**,
- a transaction nonce or identifier, and

- a cryptographic signature from the sender authorizing the transfer.

Each output note specifies a new owner public key and a fixed denomination of $KNX. Output notes are newly created objects that replace the consumed input notes.

6.2 Local Validation Rules

Before accepting an offline transaction, participants perform local validation to ensure basic correctness at execution time. These checks do not require access to global state and can be performed entirely offline.

Local validation includes verifying:

- the authenticity of the Layer 1 ledger's signature on each input note,
- that the transaction is signed by the current owner of each input note,
- that no input note appears more than once within the same transaction, and
- that the total value of input notes equals the total value of output notes.

If any of these checks fail, the transaction is rejected locally and not executed.

6.3 Ownership Transfer and Authorization

Ownership of an offline note is transferred by cryptographic authorization. The current owner signs the transaction that consumes the note, thereby authorizing the creation of new notes assigned to the recipients.

Because notes function as controlled bearer instruments, possession alone is insufficient for transfer. Authorization must be explicitly provided by the owner's private key. This prevents unauthorized reuse or reassignment of notes during offline execution.

6.4 Optional Witness Attestations

Offline transactions may optionally include **witness attestations** from nearby devices. A witness attests that it observed the execution of a specific transaction at a specific time and location context.

Witnesses do not validate correctness, enforce rules, or produce consensus. Their role is purely evidentiary. Witness attestations can strengthen accountability during reconciliation by providing additional context if conflicting histories are submitted.

To discourage false attestations, witnesses may be required to stake $KNX, which can be slashed if contradictory attestations are proven during settlement. Participation as a witness is optional and incentive-driven.

6.5 Absence of Global Uniqueness Enforcement

The offline execution protocol does not enforce global uniqueness of note identifiers at execution time. This means that, in adversarial scenarios, a malicious participant could attempt to execute multiple conflicting transactions using the same input notes.

This behavior is **intentionally not prevented** during offline execution. Enforcing global uniqueness would require continuous connectivity, trusted infrastructure, or invasive monitoring, undermining the privacy goals of the system.

Instead, KnoxNet relies on deferred enforcement. Any conflicting uses of the same note identifiers are deterministically detectable during reconciliation, where penalties are applied according to protocol rules.

7. Offline Transaction Execution Protocol

This section explains how transactions operate within KnoxNet when users are offline. The objective is to enable local value transfer without requiring internet connectivity, while still ensuring that the system remains correct and enforceable once devices reconnect.

KnoxNet explicitly separates **transaction execution** from **global settlement**. Execution occurs locally between devices, while settlement occurs later through reconciliation with the global ledger. As a result, offline transactions can be valid at the moment of exchange, but they only become globally final after reconciliation.

7.1 Transaction Structure

An offline transaction transfers ownership of one or more offline notes.

Each transaction consists of the following components:

Inputs

- The notes being spent (existing notes)

Outputs

- The new notes being created and assigned to the receiver

Authorization

- A cryptographic signature from the sender proving approval of the transfer

A valid transaction must always conserve value:

Total value in = Total value out

For example, if a user spends 10 $KNX offline, the outputs must sum to exactly 10 $KNX.

7.2 Local Validation Rules

Before accepting an offline transaction, the receiver performs a set of local validation checks. These checks are lightweight and do not require internet connectivity.

The receiver verifies the following:

1. **Note Authenticity**
   The receiver verifies that each input note was originally issued by KnoxNet and is not forged.

2. **Ownership Authorization**
   The receiver verifies that the transaction is signed by the correct private key, proving that the sender is the legitimate owner of the notes being spent.

3. **Value Conservation**
   The receiver verifies that the transaction does not create or destroy value. The total value of inputs must equal the total value of outputs.

4. **Basic Integrity Checks**
   The receiver also verifies that:

    - The same note is not referenced multiple times within the same transaction

- Output notes specify valid owners and valid values

If any of these checks fail, the transaction is rejected immediately.

7.3 Ownership Transfer and Authorization

Ownership of an offline note can only be transferred by its legitimate owner. The sender proves ownership by signing the transaction that consumes the note. The transaction then creates new output notes assigned to the receiver's public key.

This mechanism ensures that offline spending remains secure even in adversarial environments. Simply copying a note object is insufficient to spend it, as spending requires cryptographic authorization from the owner's private key.

7.4 Optional Witness Attestations

KnoxNet supports optional witness attestations for offline transactions. A witness is a nearby device that signs a statement confirming it observed a specific transaction execution event.

Witnesses do not validate transaction correctness and do not determine whether a transaction is accepted. Their role is purely evidentiary. During reconciliation, witness attestations can provide additional context in cases where conflicting histories are submitted or when participants dispute events that occurred during offline execution.

7.5 Absence of Global Uniqueness Enforcement

KnoxNet does not enforce global uniqueness of note consumption during offline execution. In offline environments, the system does not check whether an input note has already been spent in another offline history.

This is an intentional design choice. Enforcing uniqueness in real time would require internet connectivity, global broadcast, or trusted infrastructure—reintroducing network-level observability and undermining the offline-first privacy model.

Instead, global uniqueness is enforced during reconciliation. If the same note identifier appears in multiple conflicting submissions, the contradiction is detected deterministically and penalties are applied through escrow-based enforcement. This approach preserves privacy during execution while still guaranteeing eventual correctness and bounded risk.

8. Encrypted Settlement Constraints

Offline execution in KnoxNet minimizes observability at the moment of value transfer, but global settlement remains necessary to enforce correctness, supply safety, and accountability. Conventional blockchain systems perform settlement by processing plaintext transaction data, exposing transaction flows, balances, and accounting structure to validators and observers.

KnoxNet adopts a different approach. Settlement is enforced over **encrypted data**, allowing the Layer 1 ledger to verify global constraints without learning sensitive transaction details. This section describes the encrypted settlement model and the role of homomorphic encryption in enforcing correctness.

8.1 Settlement Without Plaintext Disclosure

During reconciliation, the KnoxNet Layer 1 ledger must verify a limited set of global properties, including:

- conservation of $KNX value,
- adherence to issuance and escrow limits, and
- consistency of settlement deltas across submissions.

Importantly, enforcing these properties does **not** require access to individual transaction histories, sender–receiver relationships, or plaintext amounts. KnoxNet therefore avoids processing settlement data in plaintext.

Instead, reconciliation submissions encode settlement-relevant information in encrypted form. The ledger verifies constraints directly over encrypted values, ensuring correctness without revealing underlying financial data.

8.2 Homomorphic Encryption Model

KnoxNet uses **homomorphic encryption (HE)** to enable computation over encrypted settlement data. Under this model, values such as balances, aggregates, and deltas are encrypted using the ledger’s homomorphic public key pkHEpk_{HE}pkHE (Section 5.2).

Homomorphic encryption allows specific operations—such as addition and subtraction—to be performed directly on ciphertexts. The result of these operations, when decrypted, matches the result that would have been obtained had the operations been performed on plaintext values.

This property allows the ledger to verify arithmetic constraints without decrypting individual values.

8.3 Enforced Settlement Constraints

Using homomorphic encryption, the KnoxNet Layer 1 ledger enforces the following settlement constraints:

Value Conservation
The sum of encrypted inputs across reconciliation submissions must equal the sum of encrypted outputs, modulo authorized issuance and burns.

Issuance and Escrow Limits
Encrypted settlement updates are verified to ensure that the total circulating $KNX does not exceed the sum of authorized issuance and escrowed value.

Settlement Delta Consistency
Encrypted deltas applied to ledger state must be internally consistent and correspond to valid consumption of offline notes.

These checks are performed without revealing individual transaction amounts or account balances.

8.4 Interaction with Fraud Proofs

Encrypted settlement constraints complement, rather than replace, deterministic fraud proofs.

Fraud proofs rely on structural contradictions—such as reuse of the same offline note identifier—and do not depend on encrypted arithmetic. When a fraud proof is triggered, penalties are applied independently of encrypted settlement checks.

This separation ensures that:

- arithmetic correctness is enforced through encrypted computation, and
- logical correctness is enforced through deterministic contradiction detection.

Together, these mechanisms provide comprehensive enforcement without requiring transparent settlement.

8.5 Decryption and Trust Distribution

Decryption of settlement-related ciphertexts is not required for routine verification. In cases where decryption is necessary—such as auditing or controlled disclosure—decryption capability may be distributed across a validator committee using threshold decryption, as described in Section 5.2.

This design avoids concentration of decryption authority and reduces trust assumptions. No single validator is able to unilaterally decrypt settlement data.

8.6 Privacy Properties of Encrypted Settlement

Encrypted settlement ensures that the privacy gains achieved during offline execution are not undone during reconciliation. Specifically, the settlement process does not reveal:

- individual transaction amounts,
- participant balances,
- detailed transaction graphs, or
- temporal patterns of execution.

Only the minimal information required to enforce global correctness becomes observable.

By combining offline execution with encrypted settlement, KnoxNet addresses privacy leakage at both the execution layer and the settlement layer.

8.7 Design Tradeoffs

Encrypted settlement introduces computational overhead compared to plaintext verification. KnoxNet accepts this cost deliberately, as settlement occurs asynchronously and can be amortized across batches of offline activity.

This tradeoff reflects a broader design philosophy: settlement efficiency is secondary to reducing observability at privacy-critical boundaries. Encrypted settlement is therefore treated as a core system component rather than an optional optimization.

9. Security Properties and Guarantees

This section summarizes the security properties provided by KnoxNet under the assumptions described in previous sections. These guarantees arise from the combination of offline execution, encrypted settlement, and deterministic reconciliation, rather than from any single cryptographic primitive.

9.1 Supply Safety

KnoxNet guarantees **global supply safety** of the native asset $KNX.

At all times, the total value of circulating $KNX—across both offline notes and on-chain balances—is bounded by authorized issuance and escrowed value on the KnoxNet Layer 1 ledger. Offline execution cannot introduce inflation, as all offline notes are cryptographically authorized by the ledger and reconciliation enforces conservation constraints.

Any attempt to exceed issuance limits or redeem unauthorized value is deterministically rejected during settlement.

9.2 Authorization and Ownership Safety

Only the legitimate owner of an offline note can authorize its transfer.

This guarantee follows from the unforgeability of digital signatures and the binding of note ownership to user public keys. Unauthorized transfers, note forgery, or reassignment without the corresponding private key are rejected during offline execution or reconciliation.

9.3 Double-Spend Detectability

KnoxNet guarantees **eventual detection of all double-spends**.

Because offline note identifiers are globally unique and enforced during reconciliation, any attempt to consume the same note in multiple conflicting histories results in a deterministic contradiction. Such contradictions are exposed through fraud proofs and do not rely on probabilistic detection, timing assumptions, or honest majority assumptions.

9.4 Bounded Economic Risk

Offline execution introduces bounded risk rather than systemic vulnerability.

The maximum value that can be misused by a participant is strictly limited by the amount of $KNX they have escrowed on the Layer 1 ledger. Upon detection of misbehavior, penalties are applied by slashing the associated escrow. This ensures that adversarial actions cannot cause unbounded loss or compromise the system's integrity.

9.5 Privacy Guarantees

KnoxNet provides privacy guarantees at both the execution and settlement layers.

During execution, transactions occur locally without global broadcast, preventing continuous network-level observability and metadata leakage. During settlement, encrypted enforcement ensures that transaction amounts, balances, and detailed accounting flows are not revealed to validators or observers.

While KnoxNet does not claim absolute anonymity, it significantly reduces observability compared to always-online systems by eliminating real-time exposure and minimizing settlement disclosure.

9.6 Eventual Correctness and Finality

KnoxNet guarantees **eventual correctness**.

All valid offline executions are eventually reflected in global state through reconciliation. All invalid or conflicting executions are inevitably detected and penalized. Finality is deferred rather than immediate, but correctness is guaranteed under the protocol's assumptions.

9.7 Assumptions and Scope

These guarantees hold under standard cryptographic assumptions, including secure digital signatures and encryption schemes, and assume that participants eventually reconnect for reconciliation.

KnoxNet does not assume trusted hardware, continuous connectivity, or honest-majority behavior during offline execution. Security derives from economic containment and deterministic verification rather than from real-time consensus.

10. Threat Model and Adversarial Analysis

This section describes the adversarial assumptions under which KnoxNet operates and analyzes the system's behavior against relevant threat classes. KnoxNet is designed to tolerate adversarial behavior during offline execution while preserving global correctness and bounded economic risk through reconciliation and enforcement.

10.1 Adversarial Capabilities

KnoxNet assumes adversaries with the following capabilities:

- The ability to control one or more user devices.
- The ability to execute arbitrary offline transactions, including conflicting or malicious histories.
- The ability to delay or withhold reconciliation submissions.
- The ability to observe and analyze globally visible network activity during online settlement.
- The ability to collude with other adversarial participants.

Adversaries are not assumed to have the ability to break standard cryptographic primitives, including digital signatures, hash functions, or encryption schemes.

10.2 Network-Level Surveillance

KnoxNet is explicitly designed to reduce exposure to network-level surveillance.

During offline execution, transactions are not broadcast over public networks and do not interact with shared infrastructure. As a result, adversaries observing internet traffic, RPC endpoints, or blockchain mempools cannot observe transaction execution or derive metadata such as timing or frequency of offline transfers.

During online settlement, encrypted enforcement ensures that observers cannot infer transaction amounts, balances, or detailed accounting structure from reconciliation submissions.

10.3 Double-Spending and Conflicting Histories

Adversaries may attempt to double-spend offline notes by executing multiple conflicting transactions while offline.

KnoxNet does not attempt to prevent such behavior in real time. Instead, conflicting use of the same offline note identifiers is deterministically detectable during reconciliation. Upon detection, fraud proofs are generated and penalties are applied automatically.

This design ensures that double-spending is **detectable and punishable**, even if it is not prevented at the moment of execution.

10.4 Denial of Reconciliation

An adversary may attempt to avoid penalties by refusing to reconcile or by indefinitely delaying reconciliation submissions.

Such behavior does not compromise system integrity. Unreconciled offline notes cannot be redeemed on-chain, and escrowed $KNX remains locked. This creates a strong incentive for reconciliation, as withholding reconciliation only harms the adversary by rendering offline value unusable.

10.5 Collusion and Witness Abuse

Adversaries may collude to provide false witness attestations during offline execution.

Witness attestations are optional and non-authoritative. They do not determine transaction validity and cannot override deterministic reconciliation outcomes. If witness attestations are used in conjunction with staking, false or contradictory attestations are detectable and punishable during settlement.

Collusion among witnesses does not allow adversaries to bypass global enforcement or inflate supply.

10.6 Key Compromise and Device Loss

If a user's private key is compromised, an adversary may gain the ability to authorize offline transactions using that key.

This threat is inherent to all systems based on cryptographic authorization. KnoxNet mitigates the impact of such events through bounded exposure: the maximum value at risk is limited by escrowed $KNX and optional device-level issuance caps.

Recovery and revocation mechanisms may be implemented at the application layer but are outside the scope of the core protocol.

10.7 Limitations and Non-Goals

KnoxNet does not attempt to protect against:

- adversaries with complete physical control over a user and their devices,
- side-channel attacks on device hardware,
- coercion or forced disclosure of private keys,
- permanent network partition without eventual reconciliation.

These threats are acknowledged but considered outside the scope of the protocol's design goals.

10.8 Summary of Threat Coverage

KnoxNet is designed to tolerate adversarial behavior during offline execution while preserving global correctness, bounded loss, and reduced observability. By shifting enforcement from real-time prevention to deterministic post-execution verification, the protocol accepts temporary inconsistencies in exchange for stronger privacy guarantees and resilience under adverse conditions.

1. Relationship to Zero-Knowledge Systems

Zero-knowledge (ZK) techniques are widely used in privacy-preserving blockchain systems to conceal transaction contents while maintaining public verifiability. These systems demonstrate that correctness can be proven without revealing sensitive data such as balances, transaction amounts, or internal state.

KnoxNet is compatible with these goals, but operates at a broader architectural and cryptographic scope. Rather than focusing exclusively on hiding data within a globally observable system, KnoxNet extends privacy guarantees to **execution, settlement, and network observability**, using a combination of offline execution and encrypted online enforcement.

11.1 ZK Systems as a Special Case

In conventional ZK-based blockchains, transactions are always executed online and broadcast to a global network. Zero-knowledge proofs conceal *what* is being transacted, but not *that* a transaction occurred, *when* it occurred, or *how frequently* a participant interacts with the system.

Within this model, ZK proofs are primarily used to attest to the correctness of individual transactions or state transitions. While highly effective for data confidentiality, this approach operates within the constraints of an always-online execution environment.

KnoxNet generalizes this model. In its online settlement domain, KnoxNet enforces correctness over encrypted data in a manner that subsumes ZK-style confidentiality guarantees, while also extending privacy to execution itself through offline operation.

11.2 Offline Execution Beyond ZK

KnoxNet provides privacy guarantees that are structurally inaccessible to always-online ZK systems by removing the internet from the execution path entirely.

Offline execution ensures that:

- transactions are not broadcast,
- execution timing is not globally observable, and
- network-level metadata is not continuously generated.

No zero-knowledge construction can eliminate these forms of leakage, as they arise from global observability rather than insufficient cryptographic secrecy. In this respect, KnoxNet extends privacy to a layer that ZK systems do not address.

11.3 Homomorphic Encryption as a Stronger Settlement Primitive

Beyond offline execution, KnoxNet strengthens privacy within the online domain itself by using **homomorphic encryption (HE)** for settlement enforcement.

The distinction between HE and ZK is not merely one of implementation, but of **expressiveness**.

- **Zero-knowledge proofs** assert that a specific computation was performed correctly.
- **Homomorphic encryption** allows the network to *perform the computation itself* while the data remains encrypted.

This difference has profound implications for settlement.

Using HE, KnoxNet can:

- aggregate encrypted balances across many participants,
- compute encrypted settlement deltas,
- enforce global supply and issuance constraints,
- validate conservation of value across batches of activity,

without ever decrypting individual values or generating per-transaction proofs.

In contrast, ZK-based systems typically require:

- a proof per transaction or per state transition,
- explicit circuit definitions for each verification rule,
- significant overhead to compose proofs across large batches.

HE allows KnoxNet to treat settlement as an **encrypted accounting problem**, rather than as a sequence of individually proven assertions.

11.4 Why HE Enables Stronger Privacy at Settlement

The use of homomorphic encryption enables privacy properties that are difficult or impractical to achieve with ZK alone:

- **Amortized enforcement**
  Settlement constraints can be enforced over large batches of offline activity, reducing disclosure and verification overhead.

- **Minimal structural leakage**
  Validators do not learn transaction graph structure, balance distributions, or flow patterns.

- **Constraint-centric verification**
  The ledger verifies *invariants* (e.g., total supply, conservation), not individual transactions.

- **Reduced proof surface**
  There is no need to expose per-transaction proofs or circuit-specific metadata.

These properties are especially important in KnoxNet, where settlement occurs asynchronously and must not reintroduce observability that offline execution intentionally removes.

11.5 Tradeoffs and Design Choice

Homomorphic encryption incurs higher computational cost than many ZK constructions. KnoxNet accepts this cost deliberately by confining HE to the settlement layer, where computation can be batched, delayed, and amortized.

This tradeoff reflects a core design philosophy: **privacy at settlement is more valuable than settlement latency**. By accepting higher computational cost, KnoxNet achieves a level of confidentiality and expressiveness that is difficult to replicate with proof-based systems alone.

Zero-knowledge proofs remain compatible with KnoxNet and may be layered on top of the protocol. However, they are not required for the system's primary privacy guarantees.

11.6 Superset Privacy Model

Taken together, KnoxNet provides a **superset privacy model**:

- ZK systems protect **data confidentiality** within observable execution environments.

- KnoxNet protects **execution privacy**, **settlement privacy**, and **network-level privacy**.

By combining offline execution with homomorphic encrypted settlement, KnoxNet addresses privacy leakage at layers that zero-knowledge techniques alone do not reach.

12. Conclusion

Digital privacy has steadily eroded as financial systems have become inseparable from constant connectivity and global observability. Even the most advanced cryptographic techniques cannot fully protect users when every transaction must be broadcast, timestamped, and processed by shared infrastructure.

KnoxNet begins from a different premise. Privacy is not something to be layered on top of always-online systems; it must be built into the architecture itself.

By separating execution from settlement, KnoxNet allows value to move locally and privately, without continuous internet access or real-time global visibility. Offline execution removes transactions from the network at the moment they occur, eliminating entire classes of metadata leakage. Encrypted online settlement ensures that when global coordination is required, correctness is enforced without exposing transaction amounts, balances, or detailed accounting flows.

Together, these design choices establish a dual-domain ledger model in which privacy arises from reduced observability rather than from obfuscation alone. KnoxNet does not rely on trust, surveillance, or perfect cryptographic secrecy. Instead, it combines architectural privacy with encrypted enforcement to achieve strong guarantees while preserving global correctness and economic safety.

As digital systems continue to expand into every aspect of daily life, the ability to transact privately—without constant observation—becomes increasingly important. KnoxNet demonstrates that this is not only possible, but practical, by rethinking how and where digital value is allowed to move.