ZOC (Zombie On Chain) White paper

ZOC: Identification and Quantification of "Zombie On Chain" Smart Contracts for Blockchain Sustainability

Abstract (Version 1.2.2)

The persistent and immutable nature of blockchains leads to an exponential accumulation of obsolete code on the ledger. This silent proliferation of *Digital Debris* creates a systemic health crisis: **The Blockchain's Bad Cholesterol**.

This paper proposes a taxonomy and methodology to address this threat: the concept of the **ZOC (Zombie On Chain) Smart Contract**. The ZOC Tracker is the first tool to perform **Negative Space Auditing**, focusing on the latent risk residing in forgotten code.

A ZOC is defined by the rigorous application of **Criterion X** (continuous external inactivity for nine months) and **Criterion Y** (negligible economic value). To ensure risk accuracy, the **ZOC Score** is calculated via a **Boosted Aggregation Model** based on the **Marginalization** of critical risk factors (Code Complexity, Vulnerability Signatures).

This delay and the value thresholds are established as an initial working hypothesis and are subject to validation and adjustment by data collected by the ZOC Tracker and future community consensus.

The ZOC Tracker is architected as an Analytical Data Ingestion Platform (ADIP), built on Go/ClickHouse, **to translate this taxonomy into a continuous, verifiable audit metric.**

---

1. Introduction: The Paradox of Persistence and the ZOC Crisis

The promise of blockchains rests on **immutability** and **censorship resistance**. However, this permanence generates a structural consequence: the massive accumulation of **dead** or **abandoned code**. The problem is aggravated by the high failure rate in the ecosystem: analyses show that approximately **50% to 70% of crypto projects ultimately cease their activities**, leaving behind non-functional vestiges.

The root of the problem is structural and relies on three fundamental truths of smart contracts:

* A poorly designed contract **cannot be corrected** after deployment.

* It **remains immutable and public** on the ledger, sometimes even with locked funds (Locked ZOC Contracts).

* It becomes a tangible example not to follow, but also a **valuable object** of study for research and developer education.

These structural lessons confirm the necessity of a classification. This work introduces and formally defines the concept of **Zombie On Chain (ZOC) Smart Contract**.

Figuratively, when a smart contract is deployed but serves no purpose: it is there, visible, consumes space, but is functionally lifeless.

---

2. ZOC Taxonomy: The Pollution Diagnosticl Definition: The ZOC Concept (Zombie On Chain)

The term **ZOC** designates smart contracts that are deemed **non-operational and economically insignificant**.

2.1. ZOC Classification Criteria

For a contract to be categorized as a **ZOC**, it must simultaneously satisfy the following two criteria:
$\text{Contract} = \text{ZOC IFF (Temporal Criterion X)} \wedge \text{(Value Criterion Y)}$

Temporal Criterion (Inactivity - X): Initial Period of 9 Months

The inactivity threshold is set at an initial period of **nine months (9 months)**. This period is chosen as a **working hypothesis**, based on the analogy of a complete gestation cycle, confirming abandonment or functional obsolescence.

The contract must not have registered **any external interaction** during this continuous period. On a network like Ethereum, this criterion represents approximately **2 million consecutive blocks** without activity.

Value Criterion (Economic Insignificance - Y)

The contract must present **negligible** total economic value. These thresholds are established as an **initial working basis** for the ZOC Tracker:

1. **Insignificant Native Balance**: The contract balance is less than 0.001 native unit of the network (e.g., 0.001 ETH).
2. **Minimal Secondary Assets:** The total market value of secondary assets held by the contract is **less than 10 USD**.

Methodological Note: The monetary thresholds and the nine-month period are subject to future adjustments. The deployment of the ZOC Tracker will allow, through statistical analysis of contract populations, to optimize these criteria so that they remain relevant to the evolution of blockchain economics and transaction costs.

2.2. Classification of ZOC Types

ZOC Type	Description	Security / Exploitation Potential
🧠 **Inert**	Deployed contract with no active logic, often libraries or tests.	Educational study, code design audit.
🔒 **Locked**	Contract containing funds that are inaccessible due to a withdrawal bug.	Critical bug documentation, analysis of unintentional honeypots.
🧨 **Dangerous**	Contract with a known, unpatched, but inactive vulnerability.	White Hat exploitation for security or exploit documentation.
🧳 **Abandoned**	Contract from a dead or migrated project, but still referenced by external entities.	Traffic recovery and gentle migration via proxy contract.

3. ZOC Score and ADIP Architecture: The Analytical Approach V1.2.2

The ZOC Tracker is more than just a dashboard; it is an **Analytical Data Ingestion Platform (ADIP)** designed to process the entirety of the EVM history with high performance and rigor. This architecture justifies the use of specialized infrastructure required to validate the **ZOC** taxonomy at Big Data scale.

3.1. Negative Space Auditing

The ZOC risk does not lie in the activity they generate, but in the **inactivity and oblivion** they represent. Our approach, **Negative Space Auditing**, focuses on analyzing what is present but **unused**, justified by the existence of dormant vulnerabilities that can be activated years after deployment.

3.2. Boosted Aggregation Model and Marginalization

The **ZOC Score (0-100)** is not a simple average, but the result of a sophisticated analysis process based on statistical **Marginalization**.

- **Marginalization:** The prediction of a risk probability (**Dangerous ZOC –** variable X) is achieved by **integrating (marginalizing over) the joint distribution of all contextual and complexity variables (variables Z).**
- **Boosted Aggregation Model:** This methodology combines the weighting of simple factors (X and Y) with advanced variables (Code Complexity, Advanced Exclusion Risk) to achieve a more precise and granular risk classification.
- **Bayesian Calibration:** The scoring model is subject to continuous **Bayesian Calibration**, supervised by the Security Expert (Phase 2), to refine the weighting of vulnerabilities.

3.3. ADIP Architecture: Production-Grade Requirements

The ADIP architecture is designed for massive data ingestion and reliability, requiring advanced performance solutions:

- **Go & ClickHouse:** The stack remains based on Go (concurrency via Goroutines) for ingestion and ClickHouse (OLAP, MergeTree engine) for analysis.
- **Pipeline Optimization:** Data transfer between Go and ClickHouse uses the binary serialization protocol **Protocol Buffers (Protobufs)** over JSON, to reduce latency, minimize payload size, and ensure schematic rigor.
- **High Performance Network:** The ingestion pipeline uses **TCP Keep-Alive** and **HTTP/2 Multiplexing** to optimize API calls and archiving throughput.

---

4. Case Study: The Cardano Split (November 2025)

The *Cardano Split* incident (mainnet bifurcation due to a malformed transaction) is the most recent empirical validation of the ZOC threat:

- **Proof of the Dormant Bug:** The cause was a dormant, inherited deserialization bug that exploded years after its writing.
- **ZOC Rationale:** The incident proves that idle code (a dormant bug) is a systemic threat. The **ZOC Tracker** is specifically designed to audit these millions of forgotten contracts that could hide such a dormant bug, ready to be exploited.
- **DDM Justification:** The confusion between the two temporary chains highlights the absolute necessity of a **DDM** (Data Delivery Manager) to ensure the data integrity of the ZOC platform itself.

---

5. Conclusion and Perspectives 🚀

By recognizing and measuring the existence of ZOCs, the **ZOC Tracker** establishes itself as a preventive diagnostic tool.

Our approach, based on **Negative Space Auditing** and validated by a **production-grade analytical architecture (ADIP)**, represents the next step towards the maturity and sustainability of decentralized infrastructures.
