NOVAK PROTOCOL LAWS

Law L2:
Rule Hash (HR)

The Cryptographic Identity of Governance

Authoritative Edition

The Digital Fingerprint

Every rule—whether it's a Python function, a federal regulation, or a configuration file—must have a unique cryptographic identity.

We call this the **Rule Hash (HR)**.

Identity Locked.

The Equation

Text is Math. Code is Math.

```
HR = SHA3-512( Rule Definition )
```

If you change a single comma, space, or letter in the rule:
The Hash Changes Completely.

The Enforcement Check

Before any code runs, the system compares the current hash to the approved hash.

MATCH

Execute

MISMATCH

BLOCK & ALERT

What is a "Rule"?

Source Code

Python, C++, Rust logic.

Config

Thresholds, parameters, limits.

Regulation

Legal text, statutes, policies.

Anything that controls a decision is a Rule.

Why Hash Rules?

Without L2, systems suffer from:

- **Silent Updates:** Changing logic without notice.
- **Backdoors:** Developer-inserted bypasses.
- **Drift:** Accidental version mismatches.

Immutable Governance

The Freeze

Once a rule is deployed, it is frozen. It cannot be edited.

The Update

To "change" a rule, you must deploy a **New Rule** with a **New Hash** (HR').

This creates a perfect, auditable history of every logic change ever made.

Visualizing L2

```python
def approve_loan(score):
  if score > 700:
    return True
  return False
```

The code is frozen in cryptographic amber.

Summary

L2

Rule Identity

Code is Law

L2 ensures that "Code is Law" isn't just a metaphor—it's a cryptographic reality. If the hash doesn't match, the law isn't executed.

Questions?

NOVAK Protocol Standards Series

Law L2: Rule Hash (HR)