NOVAK PROTOCOL SERIES

Standard Protocol-4 (SP-4): Implementation Requirements

Mandatory Rules, Controls, and Verification Steps

Version 1.0 (Dec 2025)

The Blueprint

SP-4 defines the binding implementation requirements for any system claiming NOVAK compliance. It builds upon the foundations of SP-1, SP-2, and SP-3.

- **Binding:** Requirements are mandatory.
- **No Weakening:** Controls cannot be bypassed.
- **Complete:** Covers Data, Process, and Audit.

Core Variable Definitions

R
Governing Rule
(Pure/Deterministic)

D
Attested Input
(Raw/Bit-for-Bit)

O
Output
(Deterministic Result)

I
Identity
(Executor ID)

T
Timestamp
(ISO 8601)

SG
Safety Gate
(True/False)

Requirement: Deterministic Rules (R)

Functionally Pure

Rules must execute without side effects. Given the same input, they must produce the exact same output across any hardware or OS.

- No Randomness
- No External State
- No Timing Dependencies

Hash Anchored

Rules cannot be "floating" text. They must be frozen and referenced by their cryptographic hash.

HR = SHA-256(rule_text)

Requirement: Input (D) & Output (O)

Input Attestation

Inputs must be captured in their raw, bit-for-bit form. Normalization is forbidden unless explicitly recorded.

🔗 **Bound To:** Identity & Timestamp.

Output Determinism

Output is derived solely from the Rule applied to the Data.

$$O = R(D)$$

🚫 No intermediate outputs cached.

Requirement: HVET & EIR

HVET Construction

The hash must be a SHA-256 concatenation of Rule, Data, Output, and Time.

```
SHA256(HR||HD||HO||T)
```

EIR Signing

The Receipt must include the HVET and be digitally signed by the device or system private key.

```
SIG(Private_Key, HVET)
```

Requirement: RGAC Continuity

Implementations must record EIRs in a hash-linked sequence to prevent history rewriting.

```
RGAC[n].link = SHA256( RGAC[n-1].hvet || RGAC[n].hvet )
```

✓ No Reordering

✓ No Deletion

✓ No Insertion

Mandatory Execution Cycle

All NOVAK implementations must follow this strict sequence. No steps may be skipped or reordered.

01 Capture Input (D)

02 Compute HR, HD, HO

03 Generate HVET

04 Generate EIR (Pre-Exec)

05 Append to RGAC

06 **RUN SAFETY GATE**

07 Execute OR Reject

Requirement: Safety Gate Logic

The Safety Gate (SG) function must evaluate to TRUE only if ALL conditions pass.

```
SG =
(RulePure)  AND  (HVETValid)  AND
(EIRValid)  AND  (RGACContinuous)  AND
(PLXStable)  AND  (PSXLegitimate)
```

IF SG = FALSE → BLOCK EXECUTION

Requirement: Integrity Checks

PL-X (Physical)

Implementations must monitor hardware stability.

- Clock Drift Monitoring
- Timing Jitter Checks
- Voltage Anomaly Detection
- Metastability Detection

PS-X (Human)

Implementations must detect intent manipulation.

- Adversarial Inputs
- Deceptive Phrasing
- Social Engineering Patterns
- Ambiguous Intent

Requirement: Interoperability APIs

```
POST /hvet/verify

POST /eir/verify

POST /rgac/verify

POST /novak/execute


GET  /novak/rules

GET  /novak/status
```

All POST endpoints must reject unauthenticated requests.

Implementation Summary

SP4

Mandatory Controls

The Final Rule

SP-4 completes the technical core. Any system missing **ANY** requirement listed in this document is effectively **NOT** NOVAK-compliant.

Status: Effective Dec 2025

Questions?

NOVAK Protocol Standards Series

Category: PBAS-Implementation (SP-4)