Law L11:
No Hidden States

Total State Explicitness & Anti-Obscurity

Authoritative Edition

L11 Definition

"All inputs, rules, and outputs must be explicitly included in HR, HD, HO, and HVET. No undocumented variables."

The Hidden Variable Problem

Legacy systems rely on invisible states that change outcomes without changing inputs.

👻 Environment Variables.

🕐 System Time.

🎲 Random Seeds.

⚙️ Local Config Files.

Visible Input (API)

↓

Hidden ENV_VARS
Hidden CONFIG
Hidden STATE

"It works on my machine" is L11 violation.

Explicit Declaration

If it affects the output, it must be an input.

Input_A   Input_B   ~~ENV_VAR~~   Config_Hash

L11 forces all "Shadow Variables" into the light.

Code: Good vs Bad
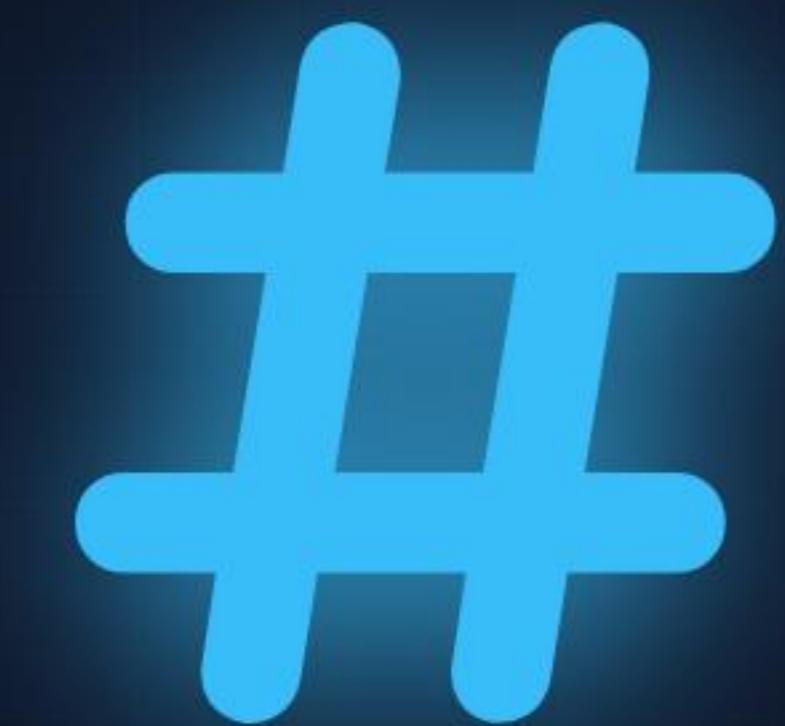
```
// VIOLATION

func calculate(amount) {
  tax = getEnv("TAX_RATE")
  return amount * tax
}

// Result depends on hidden ENV
```

```
// COMPLIANT

func calculate(amount, tax) {
  return amount * tax
}

// Result depends ONLY on input
```

The Hash Constraint

The Rule:

If a variable changes the Output (HO)...

...but that variable is NOT in the Input Hash (HD)...

The system is INVALID.

Anti-Obscurity

No Magic Numbers

Hardcoded constants must be defined in the Rule Hash (HR).

No Local Config
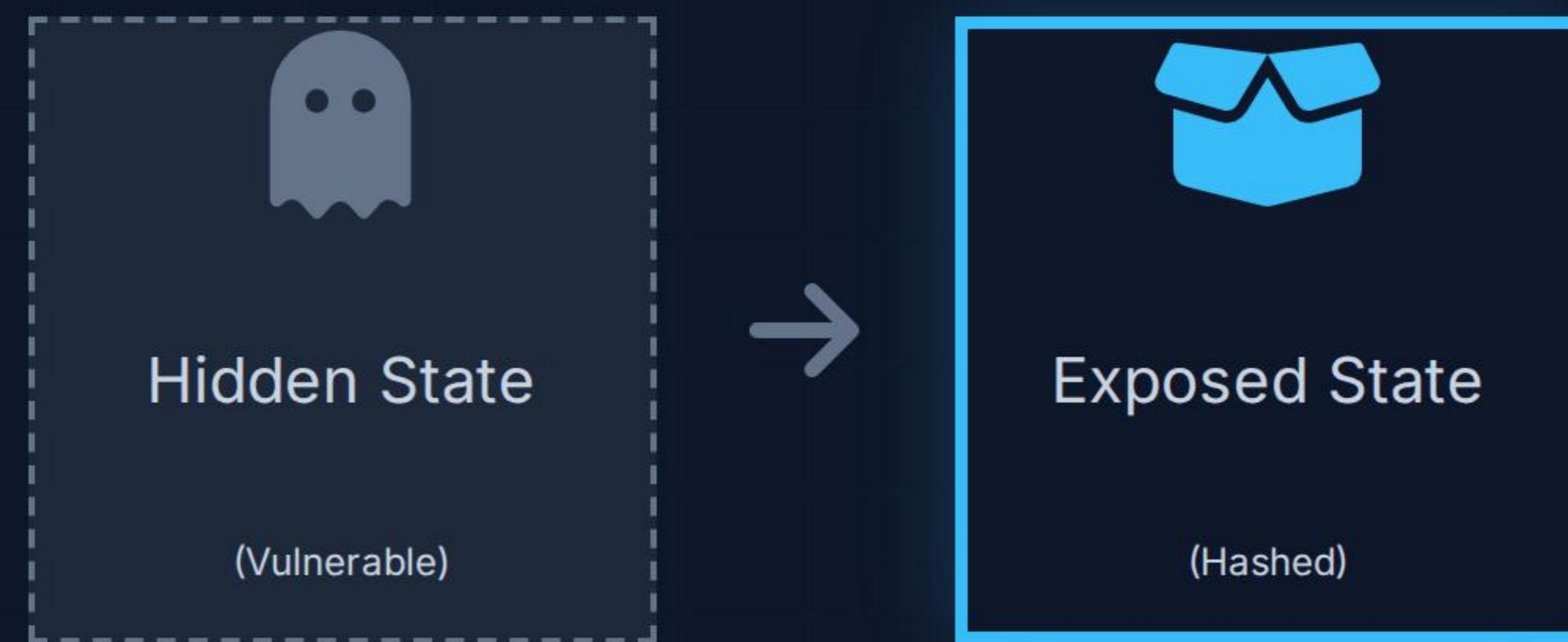
Local files must be hashed and passed as Input (HD).

No System Time

Time must be an explicit input parameter (T).

Total State Capture

Hidden State

(Vulnerable)

Exposed State

(Hashed)

L11 converts implicit context into explicit data.

Summary

L11
Explicit State

What You See Is All There Is

L11 guarantees that the HVET contains 100% of the information required to reproduce the event. Nothing is hidden.

Questions?

NOVAK Protocol Standards Series

Law L11: No Hidden States