IRSB Protocol: System Architecture Overview

Intent Receipts & Solver Bonds - A Complete Guide

IRSB Protocol Documentation

January 2026

System Architecture Overview

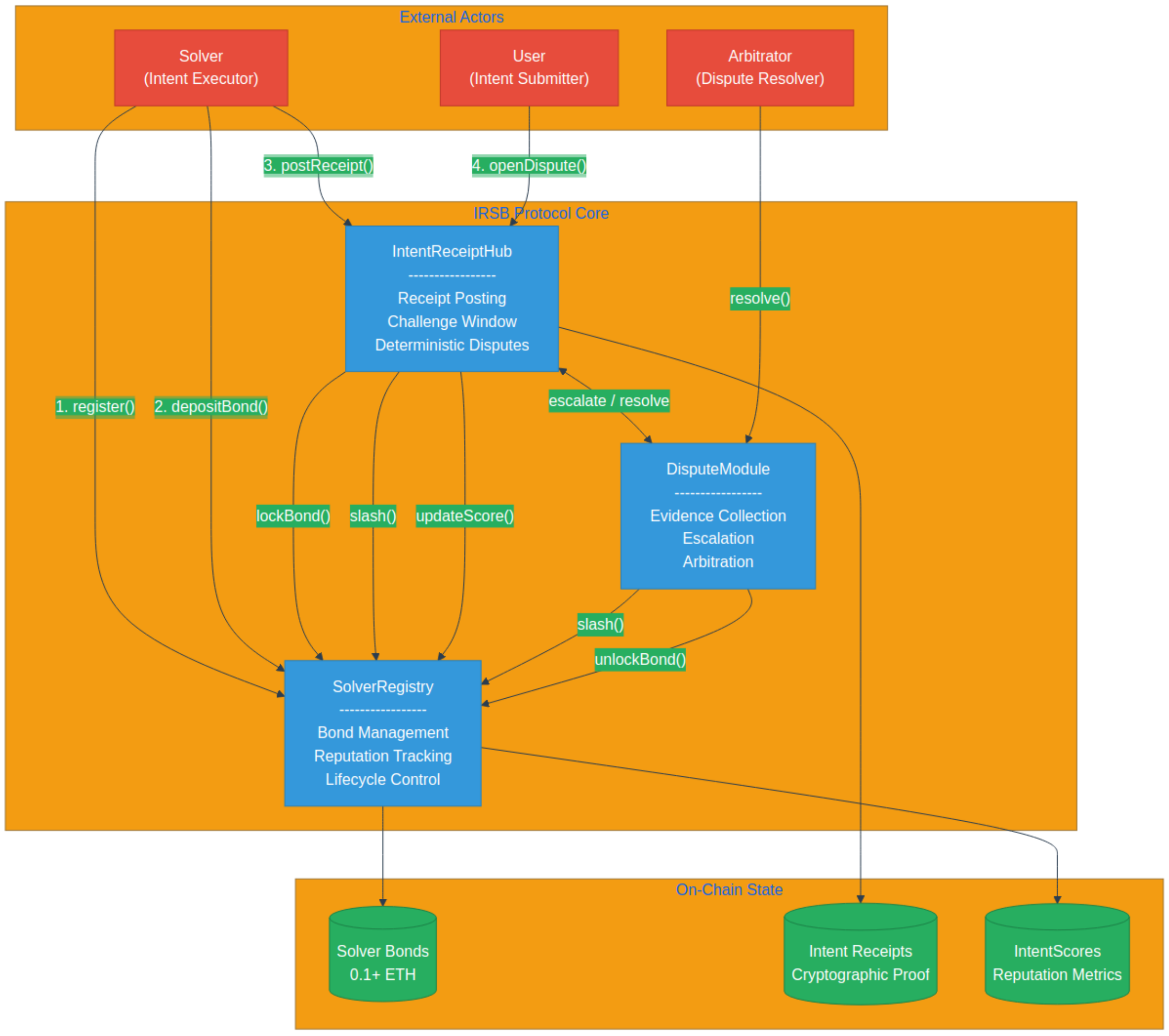

Figure 1: IRSB Protocol System Overview

Overview

The IRSB (Intent Receipts & Solver Bonds) Protocol is an Ethereum-based accountability layer designed to complement ERC-7683 cross-chain intents. It provides on-chain proof of solver execution, economic security through staked bonds, and deterministic enforcement for violations.

The protocol consists of three core smart contracts that work together to ensure trust and accountability in intent-based transactions:

1. **SolverRegistry** - Manages solver lifecycle, bonds, and reputation
2. **IntentReceiptHub** - Handles receipt posting and challenge windows
3. **DisputeModule** - Processes evidence and arbitration for subjective cases

Component Breakdown

Component	Purpose	Key Functions
SolverRegistry	Bond management & lifecycle control	`registerSolver()`, `depositBond()`, `slash()`
IntentReceiptHub	Receipt posting & disputes	`postReceipt()`, `openDispute()`, `finalize()`
DisputeModule	Arbitration for subjective cases	`submitEvidence()`, `escalate()`, `resolve()`

SolverRegistry

The SolverRegistry is the foundation of the protocol. It tracks:

- Solver registration and operator addresses
- Bond balances (both available and locked during disputes)
- Reputation scores via the IntentScore system
- Lifecycle states: Inactive, Active, Jailed, Banned

IntentReceiptHub

The IntentReceiptHub serves as the main entry point for solvers to post cryptographic proof of their intent execution. Key features:

- Signature verification using ECDSA
- 1-hour challenge window for disputes
- Deterministic resolution for timeout and signature violations
- Batch posting support (up to 50 receipts)

DisputeModule

For cases that cannot be resolved deterministically, the DisputeModule provides:

- 24-hour evidence submission window
- Escalation to human arbitrator
- 7-day arbitration timeout with default resolution
- Fair slashing distribution (70/20/10 split)

Code Reference

```
// SolverRegistry.sol - Core constants
uint256 public constant MINIMUM_BOND = 0.1 ether;
uint64 public constant WITHDRAWAL_COOLDOWN = 7 days;
uint8 public constant MAX_JAILS = 3;

// IntentReceiptHub.sol - Challenge window
uint64 public constant DEFAULT_CHALLENGE_WINDOW = 1 hours;

// DisputeModule.sol - Arbitration
uint64 public constant EVIDENCE_WINDOW = 24 hours;
uint64 public constant ARBITRATION_TIMEOUT = 7 days;
```

Key Concepts

- **Intent**: A user's desired outcome (e.g., swap 1 ETH for USDC on Arbitrum)
- **Solver**: An authorized party that executes intents and posts receipts
- **Receipt**: Cryptographic proof that a solver executed an intent
- **Bond**: Collateral staked by solvers, slashable for violations
- **Challenge Window**: Time period where anyone can dispute a receipt
- **IntentScore**: On-chain reputation metrics tracking solver performance

---

Example Scenario

Happy Path Flow:

1. Alice submits an intent to swap 1 ETH for 3000 USDC
2. Solver Bob picks up the intent and executes the swap on Arbitrum
3. Bob posts a receipt to IntentReceiptHub with his signature
4. The 1-hour challenge window passes with no disputes
5. The receipt is finalized, and Bob's IntentScore is updated

Dispute Path Flow:

1. Same as above, but Charlie notices Bob only delivered 2500 USDC
2. Charlie opens a dispute with a 0.01 ETH challenger bond
3. Bob's 0.1 ETH bond is locked during resolution
4. Deterministic verification confirms MinOutViolation
5. Bob is slashed: 80% to Alice, 15% to Charlie, 5% to treasury

---

Review Questions

1. What are the three core contracts in the IRSB Protocol?
2. How long is the default challenge window for receipts?
3. What is the minimum bond required for a solver to be active?
4. What happens to a solver after losing 3 disputes?
5. Which contract handles escalation to human arbitration?
