T3N Trusted Enterprise Agent — Travel Ops Submission

Agent: Travel Ops Agent

This maintainable Terminal3 tenant agent provides enterprise travel operations workflows:
• search-offers: searches flight offers through the host HTTP capability.
• book-offer: creates a Duffel order while traveler PII stays in host-side profile placeholders.
• Private tenant KV storage holds the provider secret; the WASM contract never receives the API key as a plaintext argument.

The design is split into a small Rust/WASM contract and a TypeScript operational walkthrough so the agent can be updated, tested, and handed over independently.

Public repository
https://github.com/nonggde/terminal3-ai-id-adk-walkthrough

Account / environment
DID: did:t3n:e8c5b5d2ae335df308156a6841d68a116d7240a5
Environment: Terminal3 testnet
Registered contract: z:e8c5b5d2ae335df308156a6841d68a116d7240a5:travel-contracts
Contract ID: 701

Verification
• Rust host tests: 7 passed.
• Release WASM build: passed.
• Tenant authentication and contract registration completed on testnet.
• No paid API key, deposits, or real-money activity were used.
• No-Duffel-key invocation failed safely with a clear missing-secret error.

Bugs / compatibility notes
1. Testnet authentication required trustAnchor: { unsafe_trust_server: true } with SDK 4.39.0.
2. The SDK emitted contract_id / contract_version while the current testnet node expected script_name / script_version; a compatibility adapter is included.
3. The current WIT binding exposes tenant-did as raw bytes, so the tenant map name requires hex encoding before addressing z:<tid>:secrets.
4. Full provider execution requires a third-party Duffel key and egress authorization; the free test stopped before that boundary.

Operations / handover
The repository includes reproducible commands, environment-variable based secret handling, tests, and handover notes. The operator can continue running the agent, or transfer maintenance to T3N without exposing provider secrets in the repository or WASM arguments.

Prepared: September 4, 2026