Coldstar — DD.xyz / Webacy API Grant Application

Ready to paste into the DD.xyz application form. Contact: MatthewKarstenConnects@gmail.com | @buildcoldstar Submitted by: @buildcoldstar | STIE Member | chainlabs.uno

---

Project Name

Coldstar

Project Website

https://coldstar.dev/colosseum

Project GitHub

https://github.com/ExpertVagabond/coldstar-colosseum

Chain

Solana

Project Description

Coldstar is an open-source air-gapped cold wallet for Solana that transforms any standard USB drive into a hardware-grade security device. Private keys are generated and stored on an offline Alpine Linux system with network drivers blacklisted at the OS level — keys never touch the internet.

The online companion application handles transaction creation, Jupiter DEX swaps, Pyth Network real-time price feeds, and on-chain DAO governance with multi-sig vaults. All signing occurs offline via QR code transfer across the air gap.

Coldstar is the only air-gapped Solana wallet with integrated DeFi access (Jupiter swaps from cold storage), real-time portfolio tracking (Pyth feeds), and on-chain DAO governance — delivering hardware-wallet-grade security at 95% less cost.

Coldstar won recognition in the Colosseum Agent Hackathon (Project #62) and is positioned as core security infrastructure for the emerging Solana agent economy — where AI agents managing significant capital need both physical key isolation and intelligent risk assessment.

How will you integrate DD.xyz / Webacy APIs?

Coldstar's architecture creates a uniquely powerful integration point for Webacy: the **pre-air-gap checkpoint**. Every transaction is created on the online device, then physically transferred to the air-gapped device for signing. Once a transaction crosses that air gap, there's no turning back. Webacy's risk intelligence will be the critical decision layer at that point of no return — giving users comprehensive due diligence before they commit to signing.

Integration 1: Threat Risks API — Recipient Address Screening

Where: Transaction creation flow (online device, before air-gap transfer)

Before generating any unsigned transaction, Coldstar will query `https://api.webacy.com` with the recipient address via the Threat Risks endpoint. The response's `overallRisk` score and threat flags will be rendered directly in our Rich terminal UI alongside the transaction details.

Behavior: - LOW risk – Green indicator, proceed normally - MEDIUM risk – Yellow warning with flag details, user confirms - HIGH risk – Red alert with full threat breakdown, requires explicit override - SANCTIONED/OFAC – Hard block, transaction cannot be created

Use case: Prevents cold wallet users from accidentally sending SOL or SPL tokens to known scam addresses, sanctioned entities, or compromised wallets — before the transaction ever reaches the air-gapped signing device.

Integration 2: Contract Risk API — Jupiter Swap Safety

Where: Jupiter DEX swap flow (online device, before swap TX creation)

Before creating a swap transaction through Jupiter's aggregator, Coldstar will analyze all involved program addresses via the Contract Risk endpoint. Token mint addresses, liquidity pool contracts, and any intermediate routing contracts are all screened.

Behavior: - Display contract risk score, deployer history, and vulnerability flags - Flag suspicious contract patterns (e.g., unverified source, proxy patterns, deployer linked to rug-pulls) - Show risk assessment alongside Jupiter's swap quote (price, slippage, route) - User sees full picture — financial terms AND security profile — before proceeding

Use case: Protects cold wallet users from interacting with malicious or vulnerable smart contracts during DeFi swaps. Especially critical because Coldstar users are high-value targets managing significant holdings.

Integration 3: Transaction Risks API — Pre-Sign Risk Scoring

Where: Final transaction review screen (online device, last step before air-gap transfer)

Every unsigned transaction gets a comprehensive risk assessment via the Transaction Risks endpoint before it's encoded for QR transfer. This is the final gate — the last screen the user sees on the online device.

Behavior: - Aggregate risk score incorporating sender profile, receiver profile, and contract risk - Visual risk breakdown in the terminal UI dashboard - Transaction simulation results showing expected outcomes - Clear go/no-go recommendation

Use case: Provides a holistic risk picture that combines all factors. Even if individual checks pass, the combined transaction context may reveal risks (e.g., a clean address interacting with a risky contract).

Integration 4: Exposure Risk API — Vault Portfolio Dashboard

Where: Coldstar vault dashboard (persistent portfolio view)

Our existing Pyth-powered vault dashboard (showing balances, USD valuations, and price changes) will be enriched with Webacy's Exposure Risk data for each connected wallet.

Behavior: - Display wallet safety score (LOW/MEDIUM/HIGH/SAFEST) as a persistent UI element - Show exposure breakdown: what percentage of holdings involve risky counterparties - Highlight specific assets or historical transactions contributing to elevated risk - Track risk score trends over time

Use case: Users see their portfolio's security posture at a glance. A cold wallet user who primarily transacts with verified protocols will see a healthy score; one who has interacted with flagged addresses will see actionable warnings.

Integration 5: Token Risk Analysis — SPL Token Screening

Where: Token display throughout the application (vault, swaps, transfers)

All SPL tokens displayed in Coldstar (SOL, USDC, USDT, BONK, JUP, RAY, and any new tokens) will carry Webacy risk metadata.

Behavior: - Badge tokens with risk indicators in the vault view - Show token-specific risk data (holder concentration, contract security, market manipulation signals) - Flag tokens with concerning patterns before users include them in transactions - Especially critical for new/unfamiliar tokens encountered through Jupiter swaps

Use case: Prevents users from holding or trading tokens with known security issues. Particularly valuable for the emerging agent economy where automated systems may acquire tokens that humans wouldn't recognize as risky.

Integration 6: URL Risk API — Companion PWA Protection

Where: Coldstar's companion Progressive Web App (mobile-friendly online interface)

Any external URLs encountered in the PWA experience (dApp links, token metadata URLs, governance proposal links) are validated against Webacy's URL Risk endpoint before navigation.

Behavior: - Intercept navigation to external URLs - Display safety verdict before allowing redirect - Block known phishing/malware domains - Log flagged URLs for community reporting

Use case: Protects Coldstar users from phishing attacks that target cold wallet users — who are high-value targets precisely because they hold significant assets.

Technical Implementation Details

Base URL: `https://api.webacy.com` **Auth:** `x-api-key` header on all requests **Chain parameter:** `solana` for all Solana-native queries **Response handling:** Parse `overallRisk` percentage, threat severity counts, and flagged issues from JSON responses **Caching:** 5-minute TTL cache for repeated address lookups (matching Webacy's server-side cache)

Architecture fit: All Webacy API calls happen on the online device only. The air-gapped device never makes network requests — it only receives unsigned transaction data (which now includes the risk assessment metadata for display on the offline signing screen). This preserves Coldstar's core security guarantee while adding intelligence.

Language/Stack: Python 3.11+ with `httpx` for async API calls, integrated into existing Rich TUI framework.

Integration Timeline (12 weeks)

Week	Milestone	API Endpoints	Deliverable
1-2	Threat Risks + Address Screening	Threat Risks API	Pre-transfer recipient validation in CLI
3-4	Contract Risk + Jupiter Safety	Contract Risk API	Swap safety checks with risk scores
5-6	Transaction Risk Scoring	Transaction Risks API	Pre-sign holistic risk assessment
7-8	Portfolio Risk Dashboard	Exposure Risk API	Vault dashboard with safety scores
9-10	Token + URL Risk	Token Analysis, URL Risk	Token badges, PWA link protection
11-12	Polish, Testing, Documentation	All endpoints	Public release, docs, blog post

Each milestone will be deployed incrementally. Bi-weekly progress updates provided.

What makes Coldstar different from other projects?

1. **Unique integration point that no other project has.** Coldstar's air-gap architecture creates a natural "point of no return" — once a transaction crosses to the offline device, it gets signed. Webacy's intelligence at this checkpoint is more impactful than in any hot wallet or browser extension where users can still cancel. The stakes of the decision are higher, making the due diligence more valuable.

2. **Security-first user base.** Coldstar users have already chosen maximum security by using an air-gapped wallet. They are exactly the audience that values due diligence — they will actively use and appreciate Webacy's risk intelligence rather than dismissing it.

3. **Agent economy infrastructure.** AI agents managing treasuries on Solana need automated risk assessment. Coldstar's MCP integration architecture (hot wallet for small transactions, cold wallet for large ones) means Webacy's APIs can provide programmatic risk gates for autonomous agent operations.

4. **Open source reference implementation.** All Webacy integration code will be publicly visible in our GitHub repo. This serves as a working reference for other Solana projects considering Webacy integration — amplifying DD.xyz's reach.

5. **DAO governance + risk screening.** Multi-sig treasury proposals validated against Webacy's risk intelligence before members cast air-gapped votes is a novel combination that institutional and DAO users need.

6. **Proven, shipped product.** Coldstar is not a concept — it's a deployed, working application that won recognition in the Colosseum Agent Hackathon. Integration can begin immediately.

Estimated API Usage

- **Threat Risks API:** ~500-2,000 calls/month (every outbound transaction + address lookups)
- **Contract Risk API:** ~200-800 calls/month (Jupiter swap validations)
- **Transaction Risks API:** ~500-2,000 calls/month (every transaction pre-sign)
- **Exposure Risk API:** ~100-500 calls/month (dashboard refreshes, cached)
- **Token Analysis:** ~200-1,000 calls/month (token metadata enrichment)
- **URL Risk API:** ~100-500 calls/month (PWA link validation)

Total estimated: 1,600-6,800 calls/month, scaling with user adoption.

Team

@buildcoldstar — Founder & Developer - GitHub: @ExpertVagabond - Twitter/X: @buildcoldstar - Full-stack developer across TypeScript, Python, Rust, Ruby, Go - Built and shipped Coldstar as a solo developer in the Colosseum Agent Hackathon - Deep Solana ecosystem experience: Jupiter, Pyth, Anchor programs, SPL tokens - Active builder: 1,800+ commits, multiple open-source projects in the Solana ecosystem - Background in security infrastructure, AI agent systems, and open-source tooling

Contact

- **Email:** MatthewKarstenConnects@gmail.com
- **Twitter/X:** @buildcoldstar
- **Telegram:** Available on request
- **GitHub:** ExpertVagabond

Additional Notes

- Coldstar already integrates with Jupiter (DEX) and Pyth Network (oracles) — Webacy would be our third major integration partner, completing the security stack
- We're open to co-marketing: featuring DD.xyz branding in our TUI, documentation, and community communications
- We'd welcome Webacy PRO access passes to distribute to our community of cold wallet users
- Happy to provide bi-weekly integration updates and maintain active communication throughout
- Whitelabel or DD.xyz branded UI — we're flexible on either approach
- The JavaScript snippet for frontend UI would be used in our companion PWA

Application prepared February 2026 for DD.xyz API Credits Grant Program Project: Coldstar — https://coldstar.dev/colosseum Applicant: @buildcoldstar | STIE Member | chainlabs.uno