01 PROTOCOL

THE PRIVACY LAYER FOR SOLANA

Zero-knowledge proofs · Multi-party computation · Confidential balances · Stealth addresses · Quantum-resistant cryptography

14	7	6
SOLANA PROGRAMS	ZK CIRCUITS	STARK AIRS

DESIGN & ARCHITECTURE DOCUMENT

v0.9.1 · March 2026

Volta Team · Developed by Slashy Fx

THE PROBLEM

Your blockchain activity is completely exposed. Traditional blockchains offer pseudonymity, not privacy. Every transaction creates a permanent trail that can be traced back to you.

100%

Transactions are public

Every transfer you make is permanently recorded on the blockchain and visible to anyone with an explorer.

73%

Users deanonymized

Blockchain analytics firms can link your wallet to your real identity through transaction patterns and exchange KYC data.

24/7

Surveillance

Governments and corporations continuously monitor and analyze financial activity on public blockchains.

$4.3B

Stolen via wallet tracking

Bad actors use public transaction data to identify and target high-value wallets for theft and social engineering.

THE SOLUTION

Protocol 01 is a comprehensive privacy layer for Solana that makes every transaction untraceable, every balance invisible, and every identity anonymous — using cutting-edge cryptography.

∞
Recurring
Private subscriptions

100%
Private
Zero knowledge

0
Traces
Unlinkable TXs

~3s
Speed
Shield + unshield

How It Works

01
CONNECT
Create or import wallet

→

02
SHIELD
Deposit to ZK pool

→

03
TRANSACT
Send via ZK proofs

→

04
RECEIVE
Withdraw — zero trace

PRIVACY STACK

Seven cryptographic layers working together to achieve total financial privacy on Solana.

1. Zero-Knowledge Proofs (Groth16 + STARK)

Groth16 (BN254) — 6 Circom circuits for shielded transfers, confidential balances, denominated pools, and subscriber ownership. Proven with snarkjs, verified on-chain via Solana's native `alt_bn128` syscalls. ~200K compute units per verification.

STARK (Goldilocks) — 6 custom AIRs for quantum-resistant proof of ownership. Hash-based (no elliptic curves), immune to Shor's algorithm. Compact ~9KB proofs. Verified on-chain with custom FRI verifier. ~889K compute units.

2. Stealth Addresses (ECDH + ML-KEM-768)

Adapted from Ethereum's EIP-5564 for Solana. Each payment creates a unique one-time address using Elliptic Curve Diffie-Hellman key exchange. **v2 adds post-quantum protection** via ML-KEM-768 (FIPS 203) hybrid encapsulation — safe against both classical and quantum adversaries.

```
Stealth Meta-Address v1: st:01<base58(spending_pub(32) + viewing_pub(32))>
Stealth Meta-Address v2: st:02<base58(spending_pub(32) + viewing_pub(32) + kem_pub(1184))>
Hybrid Secret: HKDF-SHA256(X25519_shared || ML-KEM_shared, "p01-hybrid-stealth-v2")
```

3. Denominated Privacy Pools

Fixed-denomination pools (0.1/1/10/100 SOL, 1/10/100/1000 USDC) break the link between deposits and withdrawals. All notes in a pool share the same value, making them indistinguishable. Epoch-based maturity prevents timing analysis. P2P note sharing via BLE and NFC.

4. Confidential Balances (zkSPL)

Account-model privacy using Poseidon hash commitments (quantum-resistant). Balances hidden behind ZK proofs. Deposit, withdraw, and transfer without revealing amounts on-chain. Unlike Pedersen commitments (ECC-based, broken by Shor), Poseidon is hash-based and quantum-safe.

5. Trustless On-Chain Relay

Decentralized relay network as a Solana program. Relayers stake SOL, accept encrypted jobs, and execute transactions on behalf of users. Slashing for misbehavior. No backend server, no trust assumptions. Breaks the on-chain link between sender and recipient.

6. Multi-Party Computation (Arcium)

Decentralized MPC via Arcium's Cerberus protocol. 9 encrypted circuits for: confidential relay, anonymous registry lookup, hidden nullifier, balance audit, threshold stealth scan, and private voting. Security holds as long as 1 honest node exists in the cluster.

7. Quantum-Safe Vault

Three defense layers against quantum attacks on Ed25519: **WOTS+** (one-time hash signatures, key rotates per TX), **Hash-Timelock** (SHA-256 preimage lock for cold storage), **Commit-Reveal** (2-phase TX auth prevents quantum front-running).

SYSTEM ARCHITECTURE

Monorepo with 14 Solana programs, 8+ SDKs, 3 client apps, 7 Circom circuits, 6 STARK AIRs, and 1 Rust prover service.

CLIENT LAYER

Mobile App
Expo 54 / RN 0.81
Android + iOS

Extension
Chrome MV3
Vite + React 19

Web App
Next.js 16
React 19 + Tailwind v4

SDK LAYER

specter-sdk
Stealth + wallet

zk-sdk
Shielded pool

zkspl-sdk
Confidential balance

arcium-sdk
MPC compute

p01-js
Merchant SDK

privacy-toolkit
Merkle + commit

auth-sdk
Login with P01

rpc-config
RPC fallback

PROVING LAYER

Groth16 Prover
snarkjs (WebView)
6 Circom circuits

STARK Prover
Winterfell (WASM)
6 custom AIRs

Rust Prover Service
ark-circom
Server-side proving

ON-CHAIN LAYER (SOLANA)

zk_shielded
Shielded pool + denom + subs

p01_zkspl
Confidential balances

specter
Stealth + streams

p01_trustless
Trustless pool

p01_relayer
Decentralized relay

p01_quantum_vault
WOTS+ / HTL / CR

p01_registry
Meta-address dir

p01_stark_verifier
6-circuit STARK

p01_arcium
9 MPC circuits

p01_fee_splitter
Protocol fees

p01_stream
Payment streams

p01_subscription
Recurring payments

whitelist
Access control

p01_bundler
Atomic multi-shield

Tech Stack

Anchor	0.32.1	React	19.1
Solana CLI	2.2.14 (Agave)	Expo	54 / RN 0.81
Rust	1.94.0	Next.js	16
TypeScript	5.9.0 (strict)	Winterfell	0.10 (STARK)
snarkjs	0.7.5	noble/post-quantum	0.2.1 (ML-KEM)
circom2	2.2.2	poseidon-lite	0.3.0

ON-CHAIN PROGRAMS

14 Anchor programs deployed on Solana devnet. 130+ instructions covering shielded transfers, confidential balances, stealth payments, streams, subscriptions, quantum-safe vaults, MPC, bundler, and more.

PROGRAM	PROGRAM ID	INSTRUCTIONS	SIZE
zk_shielded	`GbVM5yve...j27c`	Shield, unshield, transfer, denominated pool, subscription vault, STARK variants	1.1 MB
p01_zkspl	`Eqppog...Ppah`	Confidential deposit, withdraw, transfer, balance proof, viewer management	480 KB
specter	`2tuztg...fbSp`	Stealth send v1/v2, claim, wallet init, streams (create, withdraw, cancel)	320 KB
p01_trustless	`FnTmMx...i43Q`	Trustless shield/unshield (proof = permission), zkSPL trustless	544 KB
p01_relayer	`2okhzL...5BM`	Register relayer, submit/complete/expire job, stake, slash, key rotation	380 KB
p01_quantum_vault	`HazoS6...Th7o`	WOTS+ vault, hash-timelock vault, commit-then-reveal	290 KB
p01_registry	`QaQwpv...hQB`	Register v1/v2, update keys, update name, deregister	210 KB
p01_stark_verifier	`DGY37k...QvGs`	Init/upload/verify/close proof buffer, 6 circuits, resize	479 KB
p01_arcium	`FH1JiQ...TLPT`	9 MPC computation definitions (Arcium encrypted-ixs)	1.0 MB
p01_fee_splitter	`UdxXEv...5BM`	Split SOL/token with hardcoded 0.5% protocol fee	369 KB
p01_stream	`C92xDD...31f4c`	Create stream, withdraw intervals, cancel/refund	250 KB
p01_subscription	`3eDvPJ...BTHW`	Create subscription, process payment, pause/resume, privacy settings	310 KB
whitelist	`WhLst1...9xKm`	Merkle-based access control, allowlist management	180 KB
p01_bundler	`BndLr1...4vQp`	Atomic multi-shield CPI batching, trustless deposit aggregation	280 KB

ZK CIRCUITS & STARK AIRs

7 Groth16 circuits for classical privacy (BN254 curve), 6 STARK AIRs for quantum-resistant proofs (Poseidon over Goldilocks field).

ZK Circuits (Groth16)

CIRCUIT	CONSTRAINTS	PUBLIC INPUTS	PURPOSE
transfer	12,222	merkle_root, nullifiers, output_commitments, public_amount	2-in-2-out UTXO transfer
denominated_pool	4,273	nullifier, merkle_root, min_epoch, denomination	Fixed-amount pool unshield
denominated_transfer	5,628	nullifier, merkle_root, new_commitment	Pool note ownership transfer
confidential_balance	1,382	old_commitment, new_commitment, delta_hash	Balance update with hidden amount
balance_proof	644	commitment, threshold	Prove balance ≥ threshold
subscriber_ownership	~500	commitment	Prove subscriber identity for ZK subscriptions
note_split	~10,000	nullifier, merkle_root, output_commitments[], denomination	Cross-pool note splitting (up to 20 outputs)

STARK AIRs (Quantum-Resistant)

CIRCUIT ID	AIR	WIDTH	LENGTH	LDE BLOWUP	PROOF SIZE
0	**subscriber_ownership**	3 cols	32 rows	8x	~9 KB
1	**pool_commitment**	3 cols	32 rows	8x	~9 KB
2	**balance_proof**	4 cols	256 rows	8x	~12 KB
3	**merkle_path**	3 cols	32 rows	8x	~9 KB
4	**confidential_balance**	4 cols	256 rows	8x	~12 KB
5	**transfer**	6 cols	512 rows	8x	~15 KB

Proof Pipeline

Groth16 (Classical)

```
circom circuit.circom --r1cs --wasm --sym
snarkjs groth16 setup circuit.r1cs pot20_final.ptau
snarkjs zkey contribute → beacon finalize
snarkjs groth16 prove → 256 bytes proof
```

STARK (Quantum-Safe)

```
p01-stark build AIR trace (Winterfell v0.10)
compact NTT → LDE → Merkle (Blake3)
Fiat-Shamir Blake3(trace_root || commitment)
FRI 16 queries → 9-15 KB proof
```

SECURITY MODEL

Defense-in-depth: every layer of the stack is hardened against classical and quantum adversaries.

LAYER	MECHANISM	QUANTUM SAFE
Seed phrase	AES-256-GCM, PBKDF2 (100K iterations)	Yes (symmetric)
Note storage	NaCl secretbox (XSalsa20-Poly1305)	Yes (symmetric)
PIN storage	SHA-256 with per-device salt, constant-time compare	Yes (hash-based)
Session keys	Hardware-backed SecureStore (not AsyncStorage)	Platform-dependent
Key management	Spending key NEVER leaves device. No remote prover fallback.	N/A
ZK soundness	Groth16: invalid proofs cannot pass verification	No (BN254)
ZK completeness	Valid spends always produce valid proofs	N/A
STARK proofs	Hash-based (Poseidon over Goldilocks), 128-bit security	Yes
Stealth v2	X25519 + ML-KEM-768 (FIPS 203) hybrid ECDH	Yes (lattice-based)
Quantum vault	WOTS+ (SHA-256 hash chains), hash-timelock, commit-reveal	Yes (hash-based)
Commitments	Poseidon (algebraic hash, not ECC-based)	Yes
Double-spend	Nullifier PDAs on-chain (unique per note)	N/A
MPC threshold	Arcium Cerberus: 1-of-N honest node guarantees correctness	Protocol-level
Relay privacy	Encrypted job submission + threshold decryption	Yes (symmetric)
Clipboard	Auto-clear after 60s on all sensitive copies	N/A
Biometric	Device fallback when hardware unavailable, no bypass	N/A
PIN brute-force	Progressive lockout: 5→30s, 8→60s, 10→300s	N/A
Backup	android:allowBackup="false" (prevents adb backup)	N/A

THREAT MODEL & CRYPTOGRAPHY

Adversary Analysis

Blockchain Observer

Cannot link senders to recipients. Cannot determine amounts. Cannot analyze spending patterns. Stealth addresses create unlinkable one-time endpoints.

Quantum Adversary

STARK proofs are hash-based (immune to Shor). ML-KEM-768 stealth is lattice-based. WOTS+ vault signatures rotate per TX. Poseidon commitments are algebraic hash (not ECC).

Compromised Node

MPC Cerberus protocol ensures correctness as long as 1 honest node exists in the cluster. Relay jobs are encrypted end-to-end. Threshold decryption prevents single-node compromise.

Cryptographic Primitives

SYMMETRIC	
XSalsa20-Poly1305	Note vault encryption
AES-256-GCM	Seed encryption (ext)
Rescue-CTR	Arcium MPC inputs
ASYMMETRIC	
Ed25519	Solana signatures
X25519	Stealth ECDH
ML-KEM-768	Post-quantum hybrid (FIPS 203)

HASHING	
Poseidon (BN254)	Commitments, nullifiers
Poseidon (Goldilocks)	STARK AIR constraints
SHA-256	WOTS+, PIN, view tags
Blake3	STARK Merkle trees
SIGNATURES & KEY DERIVATION	
WOTS+ (SHA-256)	Quantum-safe OTS
HKDF-SHA256	Key derivation

Trust Model

Client Side (Zero Trust)

- Spending key **NEVER** leaves device
- Proofs generated locally (snarkjs / WebView)
- No remote prover fallback (hard-fail)
- Stealth scanning via on-chain RPC (no relayer)

On-Chain (Trustless)

- Nullifier PDAs prevent double-spend
- Groth16/STARK verification on-chain
- proof = permission (no admin keys)
- MPC: 1-of-N honesty for correctness

FEATURE SET

Denominated Privacy Pools

Fixed-denomination anonymous pools. STARK-proven unshield (quantum-resistant). Epoch-based maturity protection.

- SOL: 0.1 / 1 / 10 / 100 | USDC: 1 / 10 / 100 / 1,000
- P2P note sharing via BLE + NFC, encrypted backups

Stealth Transfers

Unique one-time addresses via ECDH. View tags for O(1) scanning. v2 adds ML-KEM-768 (quantum-resistant).

- Unlinkable stealth addresses + 1-byte view tag filter
- On-chain registry (EIP-5564) + hybrid X25519 + ML-KEM

Confidential Balances (zkSPL)

Balances hidden behind Poseidon commitments. Deposit, withdraw, transfer without revealing amounts.

- 4 operations + balance proof (solvency without exposure)
- Viewer key management for authorized audits

AI Agent

On-device AI for managing private finances. Voice input via Groq Whisper. No data leaves the device.

- Multi-provider: Groq / Gemini / Llama (local)
- Live SOL price + Fear & Greed + conversation history

Payment Streams

Interval-based recurring payments. Amount noise (±20%), timing noise, stealth per payment.

- Personal streams + services (Netflix, Spotify, etc.)
- Privacy score + ZK subscriber vaults

Quantum-Safe Vault

Three defense layers against quantum attacks on Ed25519. SHA-256 preimage is the security boundary.

- WOTS+ (32 hash chains, key rotates per TX)
- Hash-Timelock + Commit-Reveal (2-phase TX auth)

Arcium MPC — 9 Encrypted Circuits

Confidential Relay
Threshold TX decryption across MPC cluster

Anonymous Lookup
Private stealth meta-address query

Hidden Nullifier
SHA3 commitment via MPC

Balance Audit
Solvency proof without exposure

Stealth Scan
Viewing key sharded across nodes

Private Vote
Encrypted ballots + threshold tally

PRIVACY INFRASTRUCTURE

Four new systems introduced in v0.9.1 to eliminate metadata leakage, automate shielding, enable cross-pool denomination routing, and batch deposits atomically.

Auto-Shield

One-time receive addresses with autonomous shielding. Your main wallet **never** appears on-chain.

```
1. User opens Receive → fresh stealth address
2. Sender sends SOL to one-time address
3. Autonomous runner detects funds (≥ 0.05 SOL)
4. Funds auto-shielded into denominated pool
5. One-time address discarded — zero link
```

- Threshold: 0.05 SOL minimum trigger
- Max pending age: 7 days before cleanup
- Spending key never leaves device

Privacy Relay

Railway-hosted RPC proxy that strips IP, metadata, and timing patterns before forwarding to Solana.

- Strips User-Agent, Origin, Referer headers
- Random jitter: 30–120ms per request (timing decorrelation)
- Header whitelist: only Content-Type + Solana headers
- Optional Tor routing for maximum anonymity

4 privacy tiers: direct → rotation → proxy → tor

```
Client → Privacy Middleware → Railway Relay → He
lius RPC
         (strip headers)    (strip IP/jitter)
(Solana)
```

Privacy Router — Note Splitting

Cross-pool denomination routing via the `note_split` ZK circuit (~10K constraints). Break large notes into smaller denominations without revealing the link.

- Up to 20 output notes per split operation
- Dynamic epoch delay for maturity enforcement
- Atomic nullifier + Merkle insertion (no partial state)
- 0.3% protocol fee via p01_fee_splitter

Autonomous runner: polls every 60s, auto-refreshes maturity every 5min, resumes pending routes across sessions

On-Chain Bundler

Atomic multi-shield CPI batching via `p01_bundler`. Multiple users deposit in a single Solana transaction — impossible to correlate via timing.

- Trustless, permissionless, serverless
- CPI calls into zk_shielded for each deposit
- Coinbase CDP SDK integration for fiat on-ramp
- Breaks timing correlation across all participants

Deployed as standalone program — no admin keys, no upgrade authority

MOBILE APPLICATION

Native privacy wallet for Android & iOS. Expo 54 / React Native 0.81. On-device ZK proving, biometric security, Privy authentication.

Tab Navigation

Wallet
Balance, send, receive, swap

Privacy
Pools, shielded, confidential

Streams
Recurring, subscriptions

Agent
AI, voice, market data

Key Screens

Wallet Dashboard

- Balance card with visibility toggle + quick actions
- Privacy summary pill, token list, recent activity

Privacy Dashboard

- Privacy Pool hero (STARK proofs) + quick actions
- Arcium MPC status + legacy shielded migration

Denominated Pool Screens

- Notes list, shield/unshield, STARK proof generation
- P2P transfer (BLE/NFC), import/export encrypted

Streams Dashboard

- Monthly outflow + privacy score + noise badges
- Service subs + auto-payment processing

On-Device Proving

Groth16 Prover (WebView)

snarkjs running in a hidden WebView. Queue-based (1 proof at a time). 120s timeout for large circuits. Base64-bundled circuit files (19MB transfer circuit). No remote fallback — hard fail.

STARK Prover (WASM)

Winterfell compiled to 82KB WASM module. Hidden WebView execution. All 6 circuits supported. Message-passing for Merkle path strings. Mounted in app root via StarkProverProvider.

Security Features

- Privy authentication (email, SMS, social, wallet)
- PIN with SHA-256 + per-device salt
- Biometric auth (Face ID / fingerprint)
- Progressive lockout (5→30s, 8→60s, 10→300s)
- SecureStore for all secrets (hardware-backed)
- Clipboard auto-clear (60 seconds)
- App switcher blur (prevents screenshots)
- android:allowBackup="false"
- Lock screen enforced even if "none" selected

DESIGN & ARTISTIC DIRECTION

Cyberpunk aesthetic inspired by Hatsune Miku, NEEDY STREAMER OVERLOAD, and ULTRAKILL. Dark-only, high-contrast, industrial monospace with neon cyan and pink accents.

Color System

Backgrounds

■	`void`	#0a0a0c	Primary BG
■	`dark`	#0f0f12	Secondary
■	`surface`	#151518	Cards
■	`surface-2`	#1a1a1e	Elevated

Borders

■	`border`	#2a2a30
■	`border-hover`	#3a3a42

Accents

■	`cyan`	#39c5bb	Primary accent
■	`cyan-bright`	#00ffe5	Neon highlight
■	`pink`	#ff77a8	Secondary accent
■	`pink-hot`	#ff2d7a	Glitch layers
■	`yellow`	#ffcc00	Agent / warnings
■	`red`	#ff3366	Error / destructive

Text

□	`text`	#ffffff
■	`muted`	#888892
■	`dim`	#555560

Typography

Orbitron

Display / Headlines
Weight 400–900

Inter

Body Text
Weight 300–900

JetBrains Mono

Code / Technical
Weight 400–700

Visual Effects

Glitch Effects (ULTRAKILL-inspired)

- Chromatic aberration: cyan + pink offset layers
- Screen tearing: horizontal clip-path slices
- Flicker: frame-based opacity jitter
- Shake/skew: ±5° transform during bursts
- Noise bars: horizontal colored bands
- SVG turbulence overlay at 0.02 opacity

Glow & Glass Effects

- Multi-layer box-shadow (20/40/60px blur)
- Text-shadow for neon glow on headings
- LiquidGlassTabBar: BlurView + gradient + specularity
- Scanline animation: 8s vertical sweep
- Shimmer: gradient sweep at 135°
- 8-layer depth background system (web)

DESIGN PRINCIPLES

10 core principles that guide every design decision across all Protocol 01 interfaces.

01. Dark-Only

No light theme. Ever. Optimized for OLED displays. Deep void backgrounds with high-contrast neon accents.

02. Cyberpunk Aesthetic

Neon on void, industrial edges. Inspired by Hatsune Miku, NEEDY STREAMER OVERLOAD, ULTRAKILL.

03. No Black Text

All text is white, gray, or colored. Black text is forbidden to maintain contrast hierarchy.

04. Modular Colors

Each tab has its own accent color. Wallet = cyan, Privacy = pink, Streams = bright-cyan, Agent = yellow.

05. Haptic Everything

Tactile feedback on all interactions. Light for navigation, medium for actions, heavy for confirmations.

06. Spring Physics

Bouncy, organic animations using react-native-reanimated spring configs. No linear easing.

07. Glass Morphism

Blur + transparency layers. LiquidGlassTabBar with BlurView, gradient overlay, and specularity.

08. System Metaphor

Terminal-style status indicators: PROTOCOL::01 / STATUS::ACTIVE. Monospace for all technical data.

09. Asymmetric Glitch

Chaotic branding animations. Chromatic aberration, screen tearing, flicker, shake/skew during bursts.

10. Security First

App switcher blur, clipboard auto-clear, no data leaks. Privacy is a design constraint, not an afterthought.

WEB APP & BROWSER EXTENSION

Web Application (Next.js 16)

Marketing site, SDK demo, and documentation portal. Built with React 19, Tailwind v4, and Framer Motion for cinematic interactions.

Landing Page

- Hero: Animated "01" with ULTRAKILL glitch effect
- System status display: PROTOCOL::01 / STATUS::ACTIVE
- 8-layer depth background (grid, mesh, particles, scanlines, vignette, shimmer, noise)
- Semi-transparent Miku mascot (15% opacity, grayscale)
- Feature sections with alternating code previews
- Problem/solution statistics with visual contrast
- Download section: APK + Extension + Source

Pages

- **/** — Landing page with hero, features, how it works
- **/docs** — Technical documentation (all 12 programs, SDKs, circuits)
- **/sdk-demo** — Interactive SDK playground
- **/download** — APK + extension download

Web Design Highlights

- Grid pattern overlay (40px, cyan @ 5% opacity)
- Terminal-style code previews (3-dot header)
- Gradient text: cyan → pink → bright-cyan
- Custom scrollbar (0.5rem, cyan thumb on hover)
- Badge system: cyan, pink, yellow, bright-cyan

Browser Extension (Chrome MV3)

Full Solana wallet as a Chrome/Brave extension. Private by default with integrated stealth addresses and shielded transfers.

Features

- Wallet creation & import (BIP39 mnemonic)
- SOL & SPL token management with live prices
- Privacy Zone: stealth addresses + shielded transfers
- Confidential balances (zkSPL)
- Payment streams dashboard
- dApp connection (Wallet Standard)
- AES-256-GCM seed encryption
- Rate limiting on sensitive endpoints

Architecture

- Manifest V3 (Chrome + Brave compatible)
- Vite build system + React 19
- Background service worker for wallet state
- Popup UI for quick actions
- Content script for dApp injection
- RPC fallback via @p01/rpc-config
- Seed-based key derivation (BIP44)

SDK ECOSYSTEM

8 TypeScript packages for developers to integrate Protocol 01 privacy into their applications.

PACKAGE	PURPOSE	KEY EXPORTS
@p01/specter-sdk	Core privacy SDK	Stealth, wallet, transfer, relay, quantum, registry, indexer, prover
@p01/zk-sdk	ZK primitives	ShieldedClient, Note, MerkleTree, ZkProver, viewing keys
@p01/zkspl-sdk	Confidential tokens	Confidential deposit, withdraw, transfer, balance proof
@p01/arcium-sdk	MPC compute	ArciumClient, 6 use-case modules, Rescue cipher
@p01/p01-js	Merchant SDK	Protocol01 client, subscriptions, payments, React components
@p01/privacy-toolkit	Merkle + commit	Incremental tree, Poseidon, amount hash, proof format
@p01/auth-sdk	Auth integration	P01AuthClient, P01AuthServer, session management
@p01/rpc-config	RPC infrastructure	RpcConnectionManager, fallback chain, URL sanitization

Integration Example

```
import { StealthWallet } from '@p01/specter-sdk';
import { ShieldedClient } from '@p01/zk-sdk';

// Create stealth address for recipient
const stealth = await StealthWallet.generateStealthAddress(recipientMeta);

// Shield tokens into the privacy pool
const client = new ShieldedClient(connection, wallet);
const note = await client.shield({ amount: 1_000_000n, mint: SOL_MINT });

// Transfer privately (2-in-2-out UTXO)
await client.transfer({ inputs: [note], outputs: [recipientNote, changeNote] });
```

TESTING & QUALITY ASSURANCE

Comprehensive test suite with 370+ stress tests, 25+ program tests, 30+ SDK unit tests, and automated E2E flows covering every instruction and cryptographic primitive.

Test Suite Overview

CATEGORY	FILES	TESTS	COVERAGE
Program Tests (ts-mocha)	25	~200	All 12 programs, real devnet transactions
SDK Unit Tests (vitest)	30+	~150	specter-sdk, zk-sdk, privacy-toolkit, auth-sdk
E2E Tests	8	~80	Cross-program flows, STARK lifecycle, shield/unshield
Stress: Programs	1	106	All 12 programs, every instruction, concurrent ops
Stress: Crypto	1	124	Poseidon, stealth, WOTS+, Merkle, NaCl, HKDF, ML-KEM
Stress: Mobile	1	140	Vault encryption, PIN, noise, STARK builders, BLE/NFC
Rust Tests (STARK)	6	69	All 6 AIR implementations, compact proofs

Performance Benchmarks

OPERATION	PERFORMANCE
Poseidon 2-input hash (1000x)	**< 5s**
Stealth address gen (100x)	**< 3s**
WOTS+ keygen (50x)	**< 5s**
WOTS+ sign+verify (50x)	**< 10s**

OPERATION	PERFORMANCE
Merkle insert+proof (100x)	**< 10s**
Note encrypt/decrypt (1000x)	**< 5s**
Vault encrypt/decrypt (1000x)	**< 5s**
PIN hash (1000x)	**< 2s**

ROADMAP

Protocol 01 is approximately 90% code-complete. The remaining work focuses on production readiness: external audits, trusted setup ceremony, and mainnet deployment.

Completed (~90%)

- 14 Solana programs deployed on devnet
- 7 Groth16 circuits compiled + trusted setup
- 6 STARK AIRs + on-chain verifier
- Mobile app (Android) fully functional
- Chrome extension wallet
- Web app (Next.js 16)
- 8 SDK packages
- Arcium MPC integration (9 circuits)
- Quantum vault (WOTS+, HTL, CR)
- On-chain registry + trustless relayer
- Security audit fixes + hardening
- RPC fallback infrastructure
- Privacy Relay (Tor-capable) + Auto-Shield
- Privacy Router + note splitting
- On-chain bundler (atomic multi-shield)
- 370+ stress tests

Remaining for Mainnet

- **External Security Audit**
  OtterSec / Neodyme / Trail of Bits
- **Trusted Setup Ceremony**
  3+ contributors (currently 1)
- **iOS Build & Testing**
  Only Android verified so far
- **Mainnet RPC Contract**
  Dedicated provider (QuickNode/Helius paid)
- **DeFi Composability**
  Integration with Jupiter, Raydium, etc.
- **Certificate Pinning**
  TrustKit for mobile RPC endpoints

01

> THE SYSTEM CANNOT SEE YOU.

One app. Total invisibility.
Denominated pools. Confidential balances.
Anonymous subscriptions. Stealth transfers.
Quantum-resistant privacy on Solana.

PROGRAMS	ZK CIRCUITS	MPC CIRCUITS
14	13	9

PROTOCOL-01.VERCEL.APP   @PROTOCOL01_   ISSLASHY/PROTOCOL-01