# Privacy SDK

Unified Privacy SDK for Web3 Developers - A single interface for multiple privacy systems

[NPM Version](https://www.npmjs.com/package/@privacy-sdk/core) [TypeScript](https://www.typescriptlang.org/) [License: MIT](https://opensource.org/licenses/MIT) [Documentation](https://privacy-sdk.dev)

## Overview

Privacy SDK provides a unified TypeScript interface for building privacy-preserving applications across multiple Web3 privacy systems. With a single API, you can integrate privacy protocols like Railgun and Aztec into your applications without needing to learn each system’s specific implementation details.

## ✨ Features

* **🛡️ Unified Interface**: One API for multiple privacy systems
* **⚡ Production Ready**: Enterprise-grade error handling and type safety
* **🔌 Plugin Architecture**: Extensible provider system
* **🧩 Recipe System**: Pre-built solutions for common operations (transfers, swaps, shield/unshield)
* **📚 TypeScript First**: Complete type definitions and IntelliSense support
* **🎯 Developer Experience**: Intuitive APIs with comprehensive documentation

## 🚀 Quick Start

### Installation

npm install @privacy-sdk/core

### Basic Usage

import { createPrivacySDK } from '@privacy-sdk/core';  
  
// Initialize the SDK  
const sdk = createPrivacySDK({  
 provider: 'railgun',  
 chainId: 1  
});  
  
await sdk.initialize();  
  
// Execute a private transfer  
const result = await sdk.recipes.privateTransfer({  
 to: '0x742d35Cc6634C0532925a3b8D2aE2E8d8C3b7c7e',  
 amount: '1000000000000000000', // 1 ETH  
 token: '0x0000000000000000000000000000000000000000', // ETH  
 memo: 'Private payment'  
});  
  
console.log('Transaction:', result.transactions[0].hash);

## 🔧 Supported Privacy Systems

| System | Chains | Use Cases | Status |
| --- | --- | --- | --- |
| **Railgun** | Ethereum, Polygon, Arbitrum, Optimism | Private DeFi, transfers, swaps | ✅ Production |
| **Aztec** | Ethereum | Private computation, zkApps | 🚧 In Development |

## 📖 Documentation

* [**Full Documentation**](https://privacy-sdk.dev/docs) - Complete guides and API reference
* [**Getting Started Guide**](https://privacy-sdk.dev/docs#getting-started) - Your first private transaction
* [**API Reference**](https://privacy-sdk.dev/api) - Detailed API documentation
* [**Guides & Examples**](https://privacy-sdk.dev/guides) - Step-by-step tutorials

## 🧪 Recipe System

The Privacy SDK includes a comprehensive recipe system that provides pre-built solutions for common privacy operations:

### Available Recipes

| Recipe | Description | Status |
| --- | --- | --- |
| **privateTransfer** | Transfer tokens privately | ✅ Production |
| **privateSwap** | Swap tokens while maintaining privacy | ✅ Production |
| **shield** | Convert public tokens to private | ✅ Production |
| **unshield** | Convert private tokens to public | ✅ Production |
| **batchTransfer** | Execute multiple transfers efficiently | ✅ Production |
| **crossProviderTransfer** | Transfer between privacy systems | ✅ Beta |

### Recipe Usage Examples

#### Private Swap

const swapResult = await sdk.recipes.privateSwap({  
 fromToken: {  
 address: '0x0000000000000000000000000000000000000000', // ETH  
 symbol: 'ETH',  
 decimals: 18  
 },  
 toToken: {  
 address: '0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48', // USDC  
 symbol: 'USDC',  
 decimals: 6  
 },  
 fromAmount: '1000000000000000000', // 1 ETH  
 minToAmount: '1800000000', // 1800 USDC  
 slippageTolerance: 0.5 // 0.5%  
});

#### Shield/Unshield

// Shield (public -> private)  
const shieldResult = await sdk.recipes.shield({  
 token: {  
 address: '0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48', // USDC  
 symbol: 'USDC',  
 decimals: 6  
 },  
 amount: '1000000000', // 1000 USDC  
 to: 'your-private-address'  
});  
  
// Unshield (private -> public)  
const unshieldResult = await sdk.recipes.unshield({  
 token: {  
 address: '0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48', // USDC  
 symbol: 'USDC',  
 decimals: 6  
 },  
 amount: '1000000000', // 1000 USDC  
 to: 'your-public-address'  
});

#### Batch Transfer

const batchResult = await sdk.recipes.batchTransfer({  
 transfers: [  
 {  
 to: 'recipient-1',  
 amount: '100000000', // 100 USDC  
 token: {  
 address: '0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48', // USDC  
 symbol: 'USDC',  
 decimals: 6  
 }  
 },  
 {  
 to: 'recipient-2',  
 amount: '50000000', // 50 USDC  
 token: {  
 address: '0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48', // USDC  
 symbol: 'USDC',  
 decimals: 6  
 }  
 }  
 ]  
});

#### Cross-Provider Transfer

const crossProviderResult = await sdk.recipes.crossProviderTransfer({  
 sourceProvider: 'railgun',  
 destinationProvider: 'aztec',  
 sourceAddress: 'your-railgun-private-address',  
 destinationAddress: 'your-aztec-private-address',  
 token: {  
 address: '0xA0b86991c6218b36c1d19D4a2e9Eb0cE3606eB48', // USDC  
 symbol: 'USDC',  
 decimals: 6  
 },  
 amount: '1000000000' // 1000 USDC  
});

## 🏗️ Project Structure

privacy-sdk-project/  
├── packages/  
│ ├── sdk/ # Core Privacy SDK package  
│ └── website/ # Documentation website  
├── examples/ # Usage examples  
│ ├── basic-usage.ts # Simple getting started example  
│ ├── multi-provider.ts # Multi-provider setup  
│ ├── railgun-integration.ts # Railgun integration example  
│ └── cross-provider-example.ts # Cross-provider example  
└── docs/ # Architecture documentation

## 🛠️ Development

### Prerequisites

* Node.js 16.0 or higher
* pnpm (recommended) or npm
* TypeScript 4.5 or higher

### Building the SDK

# Install dependencies  
pnpm install  
  
# Build the SDK  
cd packages/sdk  
pnpm build  
  
# Run tests  
pnpm test

### Running Examples

# Run basic usage example  
pnpm tsx examples/basic-usage.ts  
  
# Run multi-provider example  
pnpm tsx examples/multi-provider.ts

## 🤝 Contributing

We welcome contributions! Please see our [Contributing Guide](CONTRIBUTING.md) for details.

### Development Process

1. Fork the repository
2. Create a feature branch
3. Make your changes
4. Add tests and documentation
5. Submit a pull request

## 📄 License

MIT License - see the <LICENSE> file for details.

## 🙏 Acknowledgments

* [Railgun](https://railgun.org) - EVM privacy protocol
* [Aztec](https://aztec.network) - Zero-knowledge protocol with private computation

## 📞 Support

* **Documentation**: [privacy-sdk.dev](https://privacy-sdk.dev)
* **Issues**: [GitHub Issues](https://github.com/privacy-sdk/core/issues)
* **Discord**: [Privacy SDK Community](https://discord.gg/privacy-sdk)
* **Email**: [support@privacy-sdk.dev](mailto:support@privacy-sdk.dev)

**Built with ❤️ by MiniMax Agent**