

CanoKey 开发实践与进展

党凡

北京交通大学 & TUNA

密码钥匙是什么？

- Cryptography token (GB/T 35291)
- Security key/token



CanoKey Canary



YubiKey 5 NFC



ACS PocketKey

密码钥匙解决什么问题？

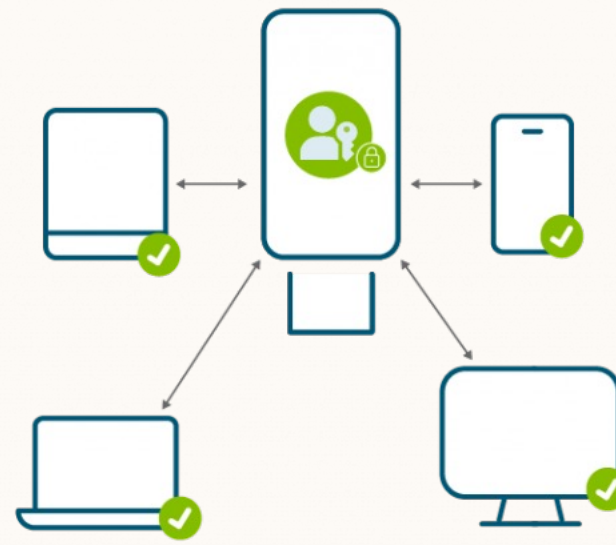
- 无密码认证
 - Passkey
- 多因子认证
 - U2F, 上一代标准, 仅用于多因素认证 (Universal 2nd Factor)
 - TOTP / HOTP
- PKI
 - PIV
- OpenPGP

Passkey

- Passkey 是一种无密码 (passwordless) 的身份认证方法
- 由 FIDO2 (WebAuthn + CTAP) 协议支持

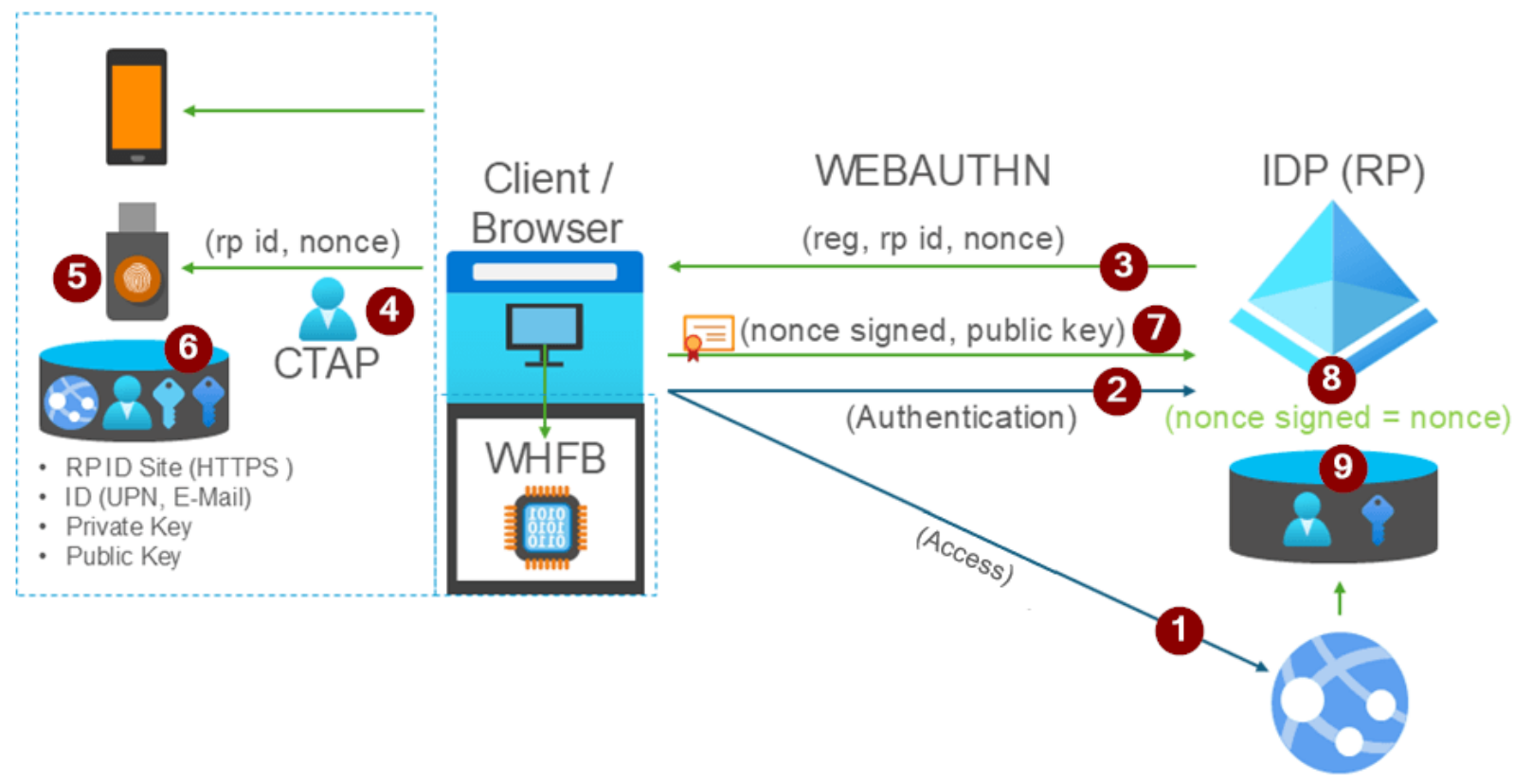


Synced Passkey



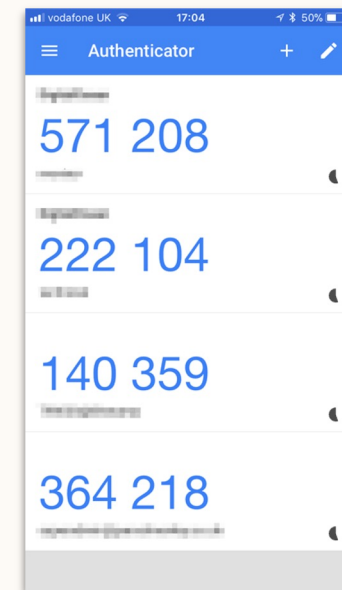
Device-bound Passkey

Passkey



TOTP / HOTP

- 什么是 OTP?
 - One-time password (PIN / passcode)
 - 常见案例：短信、Google Authenticator等
- HOTP
 - 基于 HMAC 算法的 OTP
 - HMAC-SHA1(计数器, 共享密钥), 生成 6 - 8 位数字
- TOTP
 - 计数器 -> 时间戳



PIV

- NIST SP 800-73-4
- 标准只支持 RSA 2048, ECC P-256 / P-384
- 草案增加了 RSA 3072
- 协议简单, 便于扩展其他算法

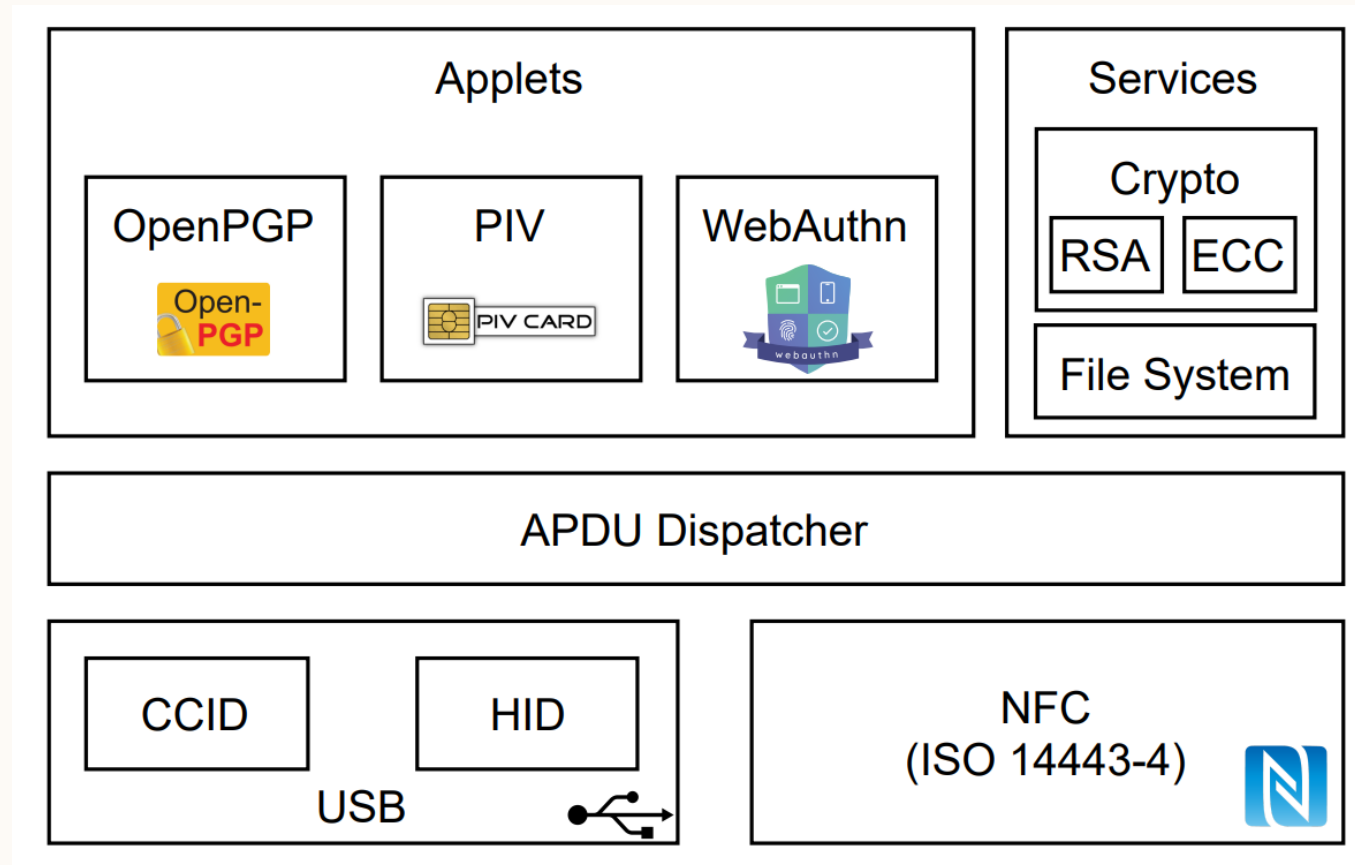
OpenPGP

- PGP (Pretty Good Privacy) 是 1991 年开发的商业软件
- OpenPGP 是 IETF 对 PGP 技术做标准化的结果, 主要包括 RFC 9580 / 3156 / 6091
- GnuPG 曾经是 OpenPGP 的一种实现
- OpenPGP Smart Card Specification 是由 g10 Code 提出的开放规范
- 近年社区产生了分裂
 - OpenPGP (RFC 9580) 采用新格式, 典型工具: Sequoia PGP
 - LibrePGP 延续了旧协议, 典型工具: GnuPG

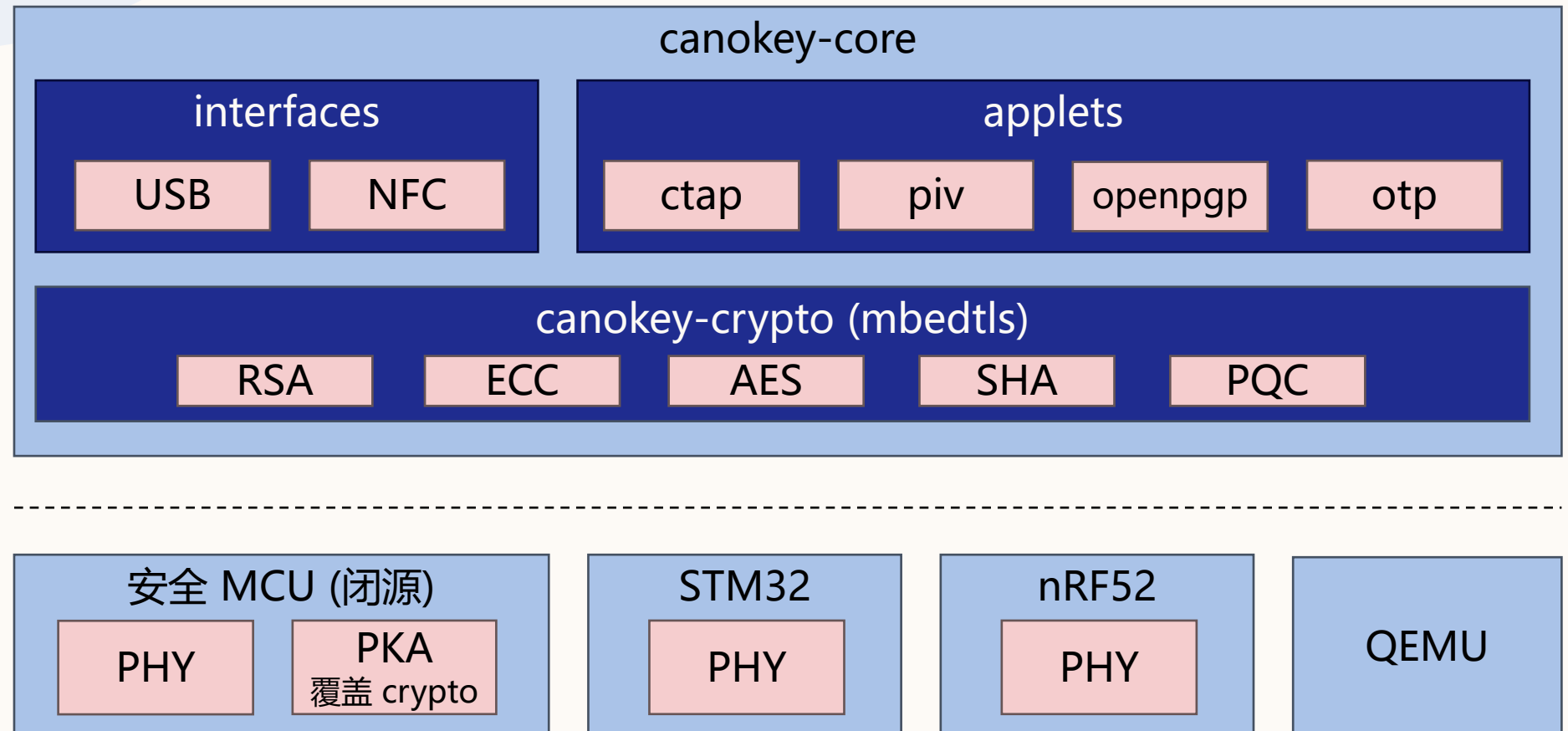
小结

- 协议实现
- 密码算法
- 外部接口

CanoKey 架构 (逻辑)



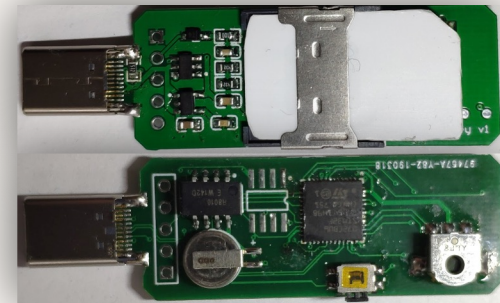
CanoKey 架构 (工程)



开发历程

CanoKey 的前辈 —— YutriKey

- 作者: @cq_z4yx
- 技术方案: STM32 MCU + Javacard
 - STM32 (~10 RMB): USB接口和用户交互
 - JavaCard (<10 RMB): 密码学算法
- 软件
 - MCU: 自己开发
 - TOTP: github.com/JavaCardOS/Oath-Applet
 - U2F: github.com/LedgerHQ/ledger-u2f-javacard
 - OpenPGP: github.com/JavaCardOS/OpenPGPApplet
 - PIV: github.com/arekinath/PivApplet



CanoKey 的前身 —— Sirius Key

- 只支持 U2F
- 照片已经找不到了
- Sirius – 天狼星，除太阳外最亮的恒星
- Canopus – 老人星，第二亮的恒星

CanoKey 的密码学算法

- RSA 2048, ECC P-256 / 384 由厂家提供
- 基于 ARM SC000 核心, 采用大端序, RSA 算法为硬掩膜实现, ECC 算法提供了二进制库
- 对着文档手搓了 CTAP 2.0, 只支持 ECC P-256, 但同期最新的 YubiKey 开始支持 ed25519 了
- 2019 年, 在前大模型时代, 经过分析以及查阅论文、专利等资料, 结论如下:
 - PKA (公钥算法加速器) 采用 Montgomery 模乘器, 最大位宽 2048
 - PKA 有独立的 3 KiB RAM
 - PKA 支持加、减、乘法运算

CanoKey 的密码学算法

- 参考 μNaCl , 人肉编译了 Montgomery ladder

```
sm2MonAdd(b5, b1, b2);      // A = X2+Z2
sm2MonSub(b6, b1, b2);      // B = X2-Z2
sm2MonAdd(b1, b3, b4);      // C = X3+Z3
sm2MonSub(b2, b3, b4);      // D = X3-Z3
sm2MonMul(b3, b2, b5);      // DA= D*A
sm2MonMul(b2, b1, b6);      // CB= C*B
sm2MonAdd(b1, b2, b3);      // T0= DA+CB
sm2MonSub(b4, b3, b2);      // T2= DA-CB
sm2MonSqr(b3, b1);          // X5==T1= T0^2
sm2MonSqr(b1, b4);          // T3= t2^2
sm2MonMul(b4, b1, X0);      // Z5=X1*t3
sm2MonSqr(b7, b5);          // AA=A^2
sm2MonSqr(b5, b6);          // BB=B^2
sm2MonSub(b8, b7, b5);      // E=AA-BB
sm2MonMul(b1, b7, b5);      // X4= AA*BB
sm2MonMul(b6, b8, CONST121666); // T4 = a24*E
sm2MonAdd(b7, b6, b5);      // T5 = BB + t4
sm2MonMul(b2, b7, b8);      // Z4 = E*t5
```

CanoKey 的密码学算法

- 2020 年，设计了使用 2048 位模乘器计算 RSA 4096 的算法
- 算法部分以论文的形式做了公开

Fan Dang, Lingkun Li, Jiajie Chen, xRSA: Construct Larger Bits RSA on Low-Cost Devices, in IEEE ICPADS 2021

- 2022 年（LLM前夜）进一步实现了 BPSW 算法，支持在芯片内生成 2048 位素数
- 2026 年，借助 GPT-5.5/5.6，素数生成时间平均减半

CanoKey 的密码学算法

- 2026 年, PQC 越来越流行
- 但大部分 MCU 的 RAM 都极小, 而 ML-DSA-65 的签名大于 3 KiB
- 在 Claude 和 GPT 的帮助下, 实现了 ML-DSA 和 ML-KEM 的流式计算

Fan Dang, Xiaoyi Liu, Shengqi Chen, and Yichuan Gao, A Stack-Efficient Design of ML-DSA-65 for Resource-Constrained Secure Elements. In ACM-TURC 2026

- 由于 OpenPGP 社区分裂, 暂时只计划在 WebAuthn 和 PIV 中支持 PQC

CanoKey 的 NFC 实现

- CanoKey 采用的芯片 (HED、STM32 等) 本质不是卡芯片, 只能外挂 NFC 芯片
- NFC 也是一个很复杂的家族, 物理层不同, 应用层也不同
- 大部分智能卡采用 NFC Type A (ISO 14443-A) + ISO-DEP (ISO 14443-4) 通信
- CanoKey 第一版采用复旦微电子 FM11NC08 芯片
 - Vout 限流
 - 不支持 reset 协议栈
 - NFC 体验极差
 - 贡献了仅有的 FM11NC08 T4T (Type 4 Tag) 开源代码

CanoKey 的 NFC 实现

- 调研过其他天价方案，搭过分立元件方案，但都不好用
- 现在使用复旦微电子 FM11NT08 芯片
 - 不再限流
 - 支持 reset 协议栈
 - 稳定性明显提升，但效果仍远不如专用卡芯片
 - 贡献了仅有的 FM11NT08 T4T (Type 4 Tag) 开源代码

CanoKey 的周边工具

- CanoKey Console: 支持通过 WebUSB 管理 CanoKey, 采用 flutter 开发
- CanoKey QEMU: 支持给虚拟机挂载虚拟的 CanoKey

其他开源工具

flutter_nfc_kit 3.6.2

Published 7 months ago • [nfc.im](#)

SDK | FLUTTER | PLATFORM | ANDROID | IOS | WEB

278

[Readme](#) | [Changelog](#) | [Example](#) | [Installing](#) | [Versions](#) | [Scores](#)

Flutter NFC Kit

pub v3.6.2 Test passing Build Example App passing

Yet another plugin to provide NFC functionality on Android, iOS and browsers (by WebUSB, see below).

This plugin's functionalities include:

- read metadata and read & write NDEF records of tags / cards complying with:
 - ISO 14443 Type A & Type B (NFC-A / NFC-B / MIFARE Classic / MIFARE Plus / MIFARE Ultralight / MIFARE DESFire)
 - ISO 18092 (NFC-F / FeliCa)
 - ISO 15693 (NFC-V)
- R/W block / page / sector level data of tags complying with:
 - MIFARE Classic / Ultralight (Android only, MIFARE Classic Read & Write block for iOS)
 - ISO 15693 (iOS only)
- transceive raw commands with tags / cards complying with:
 - ISO 7816 Smart Cards (layer 4, in APDUs)
 - other device-supported technologies (layer 3, in raw commands, see documentation for platform-specific supportability)

Note that due to API limitations, not all operations are supported on all platforms. You are welcome to submit PRs to add support for any standard-specific operations.

This library uses [ndef](#) for NDEF record encoding & decoding.

278 | 160 | 71k
LIKES | POINTS | DOWNLOADS

[Documentation](#)

[API reference](#)

[Publisher](#)

[nfc.im](#)

[Weekly Downloads](#)



[Metadata](#)

Provide NFC functionality on Android, iOS & Web, including reading metadata, read & write NDEF records, and transceive layer 3 & 4 data with NFC tags / cards

[Repository \(GitHub\)](#)
[View/report issues](#)
[Contributing](#)

其他开源工具

ccid 0.1.8

Published 18 months ago • [nfc.im](#)

SDK | **FLUTTER** | PLATFORM | ANDROID | IOS | LINUX | MACOS | WINDOWS

👍 4

Readme | Changelog | Example | Installing | Versions | Scores

ccid

pub v0.1.8 Build Example App no status

A Flutter plugin for reading and writing smart cards using the CCID protocol with PC/SC-like APIs.

Installation

Android

This plugin uses AGP 8.7, thus requires Gradle 8.7+ (which needs Java 17+ to run).

Linux / Windows

This plugin uses [dart_pcsc](#), thus has a runtime dependency on [PCSC Lite](#) to provide PC/SC APIs on Linux:

- Debian / Ubuntu: `sudo apt-get install pcscd libpcsc-lite1`
- RHEL / Fedora: `sudo dnf install pcsc-lite`

Should you encounter any permission problem, please check [README.polkit](#) from PCSC Lite.

macOS / iOS

This plugin uses [CryptoTokenKit](#) on macOS, which is available since macOS 10.10 / iOS 13.0.

The `com.apple.security.smartcard` entitlement is required.

4 150 966
LIKES POINTS DOWNLOADS

Documentation

[API reference](#)

Publisher

[nfc.im](#)

Weekly Downloads



Metadata

A Flutter plugin for smart card reader using CCID protocol with PC/SC-like APIs.

[Repository \(GitHub\)](#)
[View/report issues](#)

License

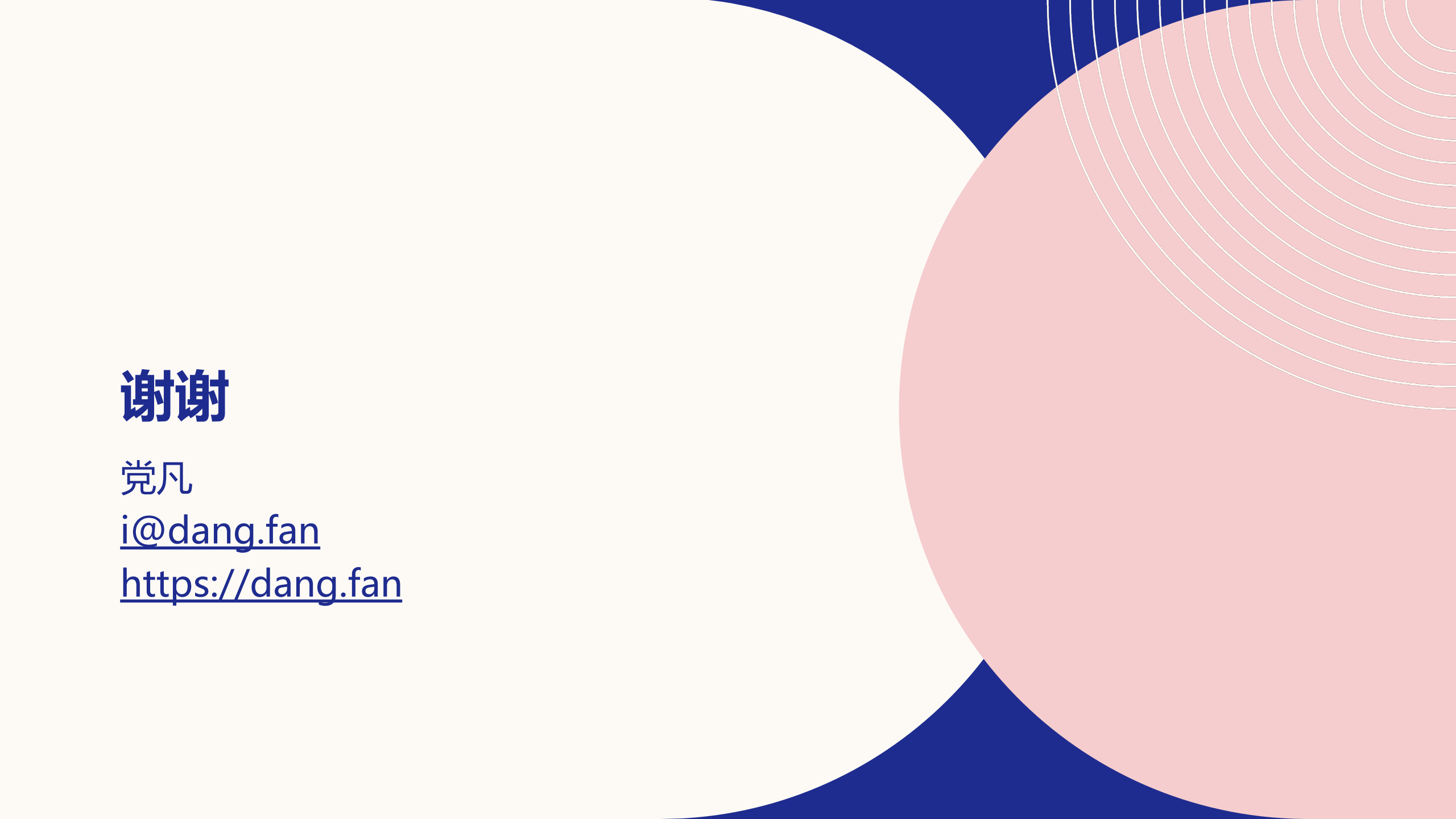
[MIT \(license\)](#)

开发计划

- 今年完成 PIV 对 PQC 的支持
- 今年 (?) 完成 canokey-pkcs11 和 canokey-mini-driver 的开发
- 更新 canokey-qemu 等开源版本

感谢以下开源软件

- littlefs
- tinycbor
- mbedtls (tf-psa-crypto)
- μ NaCl
- yubikey-manager

The background features a large white circle on the left and a large pink circle on the right, both partially overlapping a dark blue area at the top and bottom. The pink circle contains several thin, white, concentric circular lines.

谢谢

党凡

i@dang.fan

<https://dang.fan>