开源硬件密钥设备 Web 控制台

技术方案

项目描述

该项目试图开发一个在web上运行的类似Yubikey Management Tool的管理工具，用来查看和管理OATH、FIDO2、OpenPGP、PIV四种加密方式的私钥和证书。让用户能够直接在浏览器中以可视化方式管理自己的加密设备，大大增加了密码学工具可用性。

项目难点

目前已经实现了CanoKey硬件的基本功能，定义了通讯协议。WebConsole的开发过程主要需要关注于各个加密协议文档的内容和传续协议的具体实现，搭建起web控制和智能卡之间的桥梁。目前网站利用Create React App框架搭建，已经实现了主页面布局和OATH的部分查看、管理功能。

详细方案

利用Create React App框架搭建整个网站，以material-ui为UI框架，以JS为主要开发语言。

通过ISO7816协议实现数据传输。数据以异步半双工方式经I/O线在电脑和智能卡双向传输。据我观察具体采用的应该是T=0时的异步半双工字符传输协议。

caonkey的说明文档中定义好了每个操作的code，例如可查到Write FIDO key的Request通信协议如下，而且需要先验证PIN。通过定义好的APDU即可传输数据。

Field	Value
CLA	00h
INS	01h
P1	00h
P2	00h
Lc	Length of the key(20h)
Data	Private key

OATH

阅读已有的OATH实现代码，整理出实现框架，对原有代码进行重构。

PIV

理解PIV的NIST Special Publication 800-73-4文档，并以此为基础设计和提炼出PIV的通信格式。

FIDO2

理解FIDO2的Client to Authenticator Protocol文档，并以此为基础设计和提炼出FIDO2的通信格式。实现支持最多64位的resident keys和HMAC扩展。

OpenPGP

理解FIDO2的Functional Specification of the OpenPGP application on ISO Smart Card Operating Systems文档，并以此为基础设计和提炼出OpenPGP的通信格式。实现支持RSA 2048、ECDSA and ECDH：secp256r1 (NIST P256) 、ED25519 and Curve25519多种加密算法。

时间规划

查看、管理 OATH（TOTP、HOTP） 7.1 - 7.14

- 阅读OATH APPLET文档 7.1-7.3
- 梳理已有的OATH实现框架 7.3 - 7.7
- 重构现有的代码 7.8 - 7.13
- 测试 7.14

导入 FIDO2 私钥和证书 7.15 - 8.4

- 阅读FIDO2相关文档7.15 - 7.17
- 设计和提炼通信格式7.18 - 7.21
- 实现双向通信7.22 - 7.31
- 设计并实现相关网页界面8.1- 8.3
- 测试 8.4

管理 OpenPGP 的基本信息和密钥 8.5 - 8.25

- 阅读OpenPGP相关文档8.5 - 8.7
- 设计和提炼通信格式8.8 - 8.11
- 实现双向通信8.12 - 8.21
- 设计并实现相关网页界面8.22 - 8.24
- 测试 8.25

管理 PIV 的证书和密钥 8.26 - 9.15

- 阅读PIV相关文档8.26 - 8.28
- 设计和提炼通信格式8.29 - 9.1
- 实现双向通信9.2 - 9.11
- 设计并实现相关网页界面9.12 - 9.14
- 测试 9.15

管理 FIDO2 的 resident key 9.16 - 9.30

- 实现双向通信9.16 - 9.25
- 设计并实现相关网页界面 9.26 - 9.29
- 测试 9.30