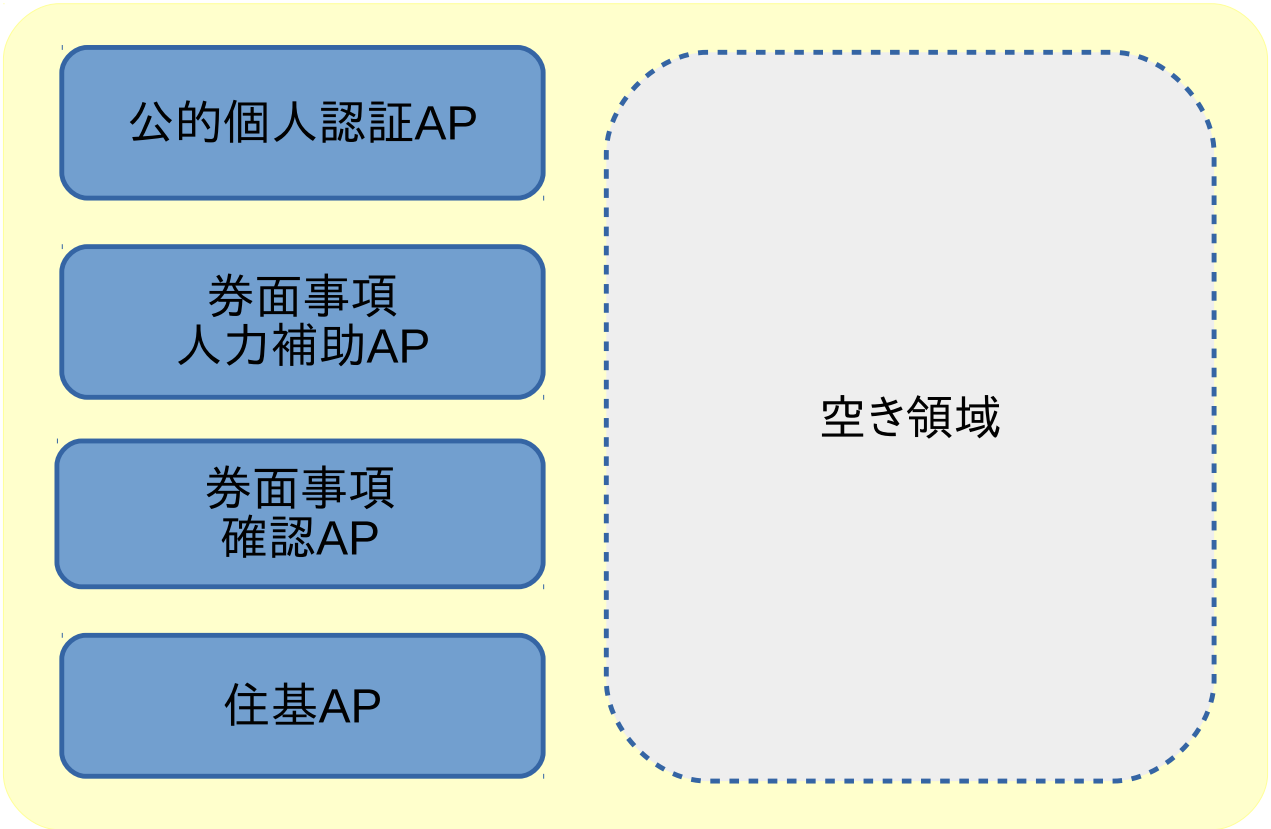
OpenSC JPKIカードドライバ

Open Source Solution Technology Corporation
HAMANO Tsukasa <hamano@osstech.co.jp>

PKI Day 2017

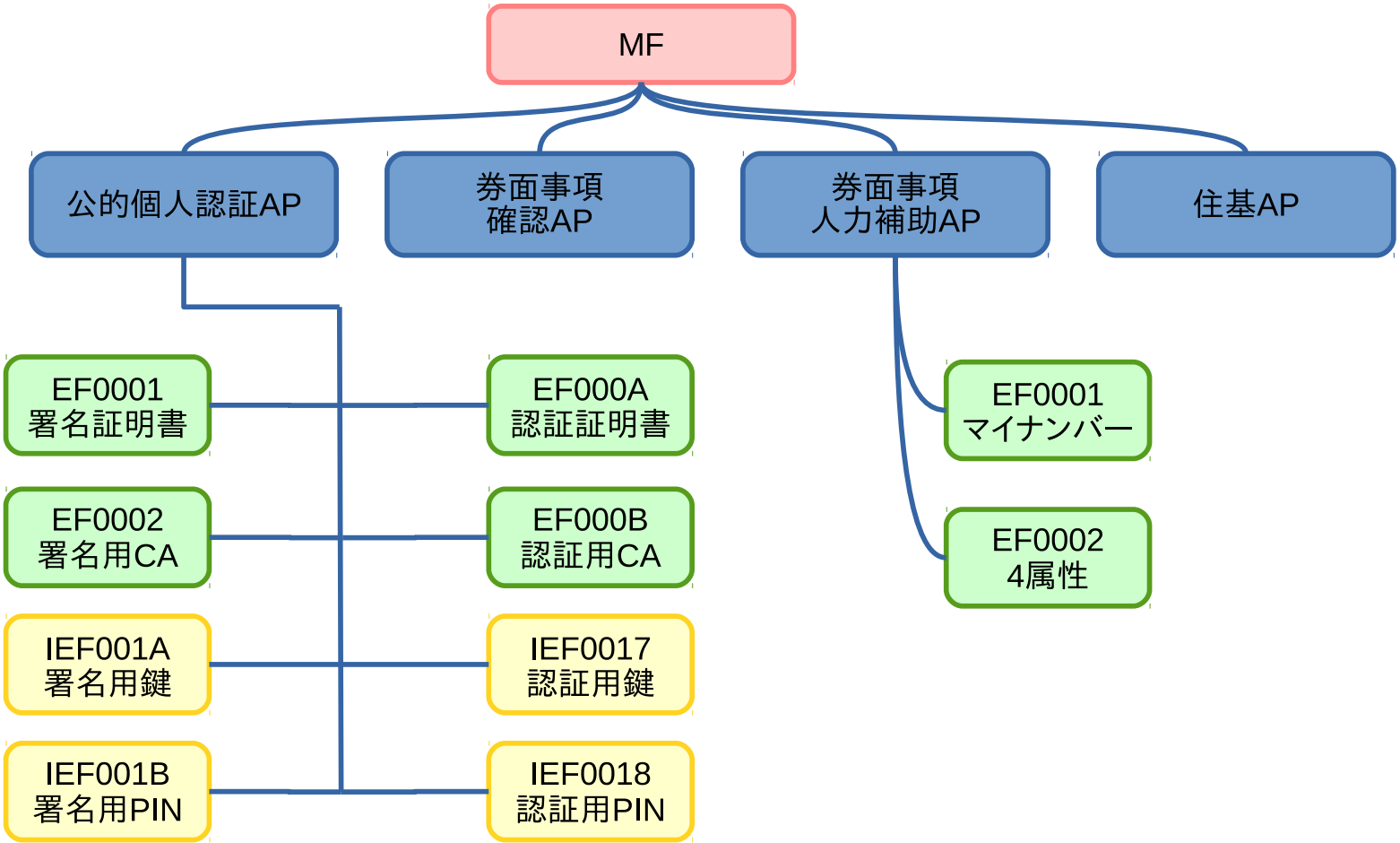
個人番号カード(マイナンバーカード)



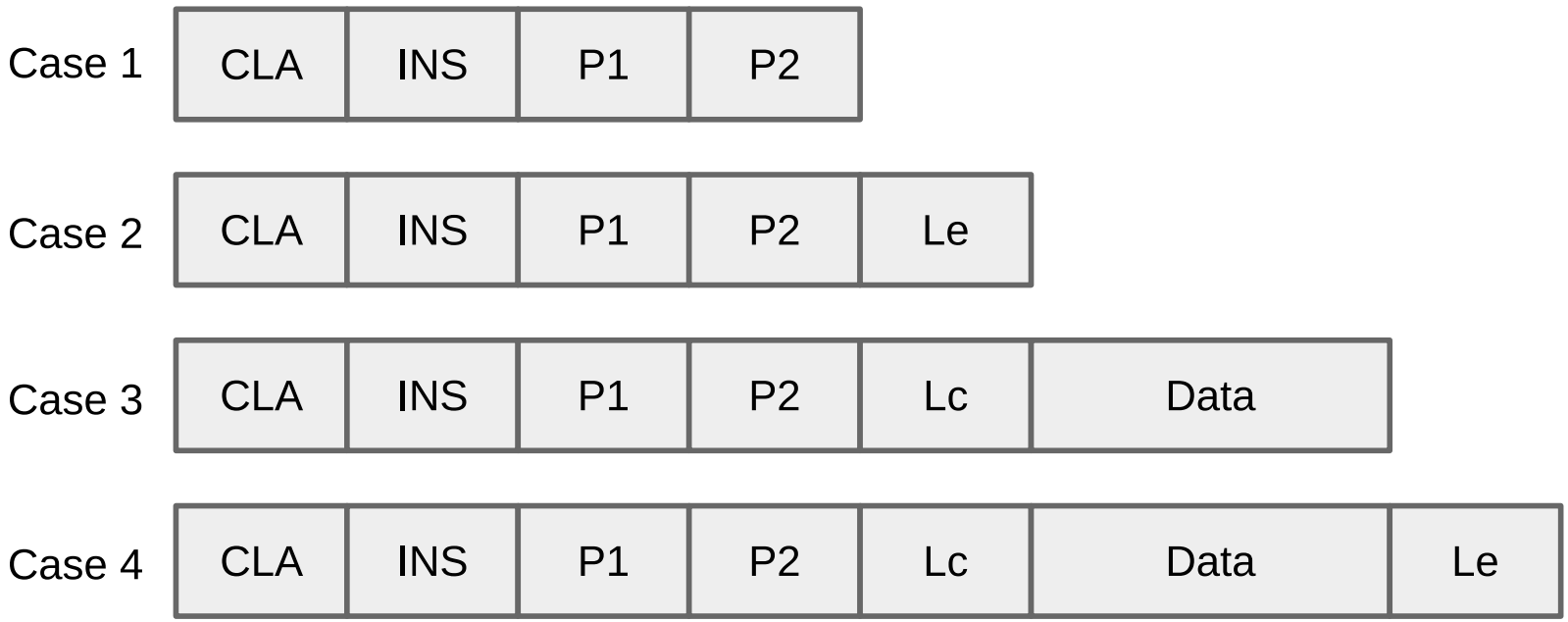
２種類の証明書

- 署名用証明書
  - 身分証明
  - 電子申請
- 利用者認証用証明書
  - 行政・民間サイトでの認証用

データモデル


MF
公的個人認証AP
券面事項
確認AP
券面事項
人力補助AP
住基AP
EF0001
署名証明書
EF0002
署名用CA
IEF001A
署名用鍵
IEF001B
署名用PIN
EF000A
認証証明書
EF000B
認証用CA
IEF0017
認証用鍵
IEF0018
認証用PIN
EF0001
マイナンバー
EF0002
4属性

APDU(ISO 7816-4)



APDU通信例 - SELECT FILE

```
> 00 A4 02 0C 02 00 01 # SELECT FILEコマンド
> 90 00
< 00 B0 00 00 10       # READ BINARYコマンド
> FF 10 0C XX XX XX XX XX XX XX XX XX XX XX FF

> 00 A4 02 0C 02 00 02 # SELECT FILEコマンド
< 00 B0 00 00 FF       # READ BINARYコマンド
> FF 20 82 00 83 DF 21 08 00 10 00 1F 00 79 00 84
> DF 22 0C E6 BF B1 E9 87 8E E3 80 80 E5 8F B8 DF
> 23 57 E6 9D B1 E4 BA AC E9 83 BD E5 A4 A7 E7 94
> B0 E5 8C BA XX XX XX XX XX XX XX XX XX XX XX XX
> XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
> XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
> XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
> XX XX XX XX XX XX XX XX XX DF 24 08 YY YY YY YY
> MM MM DD DD DF 25 01 31 FF FF FF FF FF FF FF FF
```

APDU通信例 - PIN入力

```
< 00 20 00 80 04 XX XX XX XX
> 90 00 # 成功
```

```
> 63 C2 # 失敗
```

APDU通信例 - 署名

秘密鍵 IEF を SELECT FILE

```
< 00 A4 02 0C 02 00 17
> 90 00
```

署名

```
< 80 2A 00 80 [PKCS1 DigestInfo]
> [署名データ]
```

PKCS1 DigestInfo

```
SEQUENCE {
   SEQUENCE {
      OBJECT IDENTIFIER \
         sha1(1 3 14 3 2 26)
      NULL
   }
   OCTET STRING XX XX .. XX XX
}
```

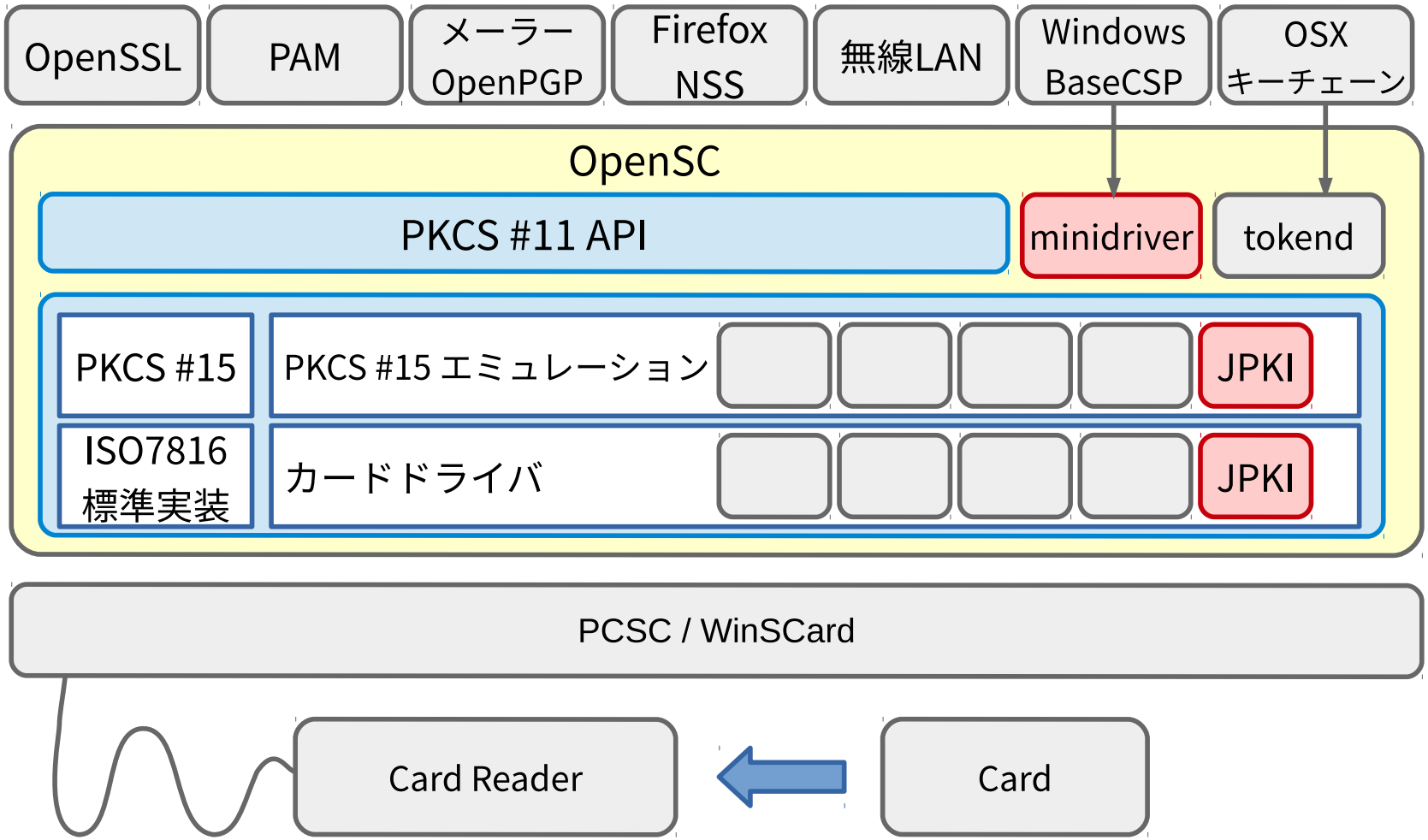
ロードマップ

- カードエッジ仕様の解析
- OpenSC カードドライバ開発
- オープンな PKCS#11 実装
- PKCS#15 カードエミュレーション
- 公的個人認証の普及

仕様を隠さないで

- 安全性の為に
- 普及の為に

OpenSCスタック


OpenSSL
PAM
メーラー
OpenPGP
Firefox
NSS
無線LAN
Windows
BaseCSP
OSX
キーチェーン
OpenSC
PKCS #11 API
minidriver
tokend
PKCS #15
PKCS #15 エミュレーション
JPKI
ISO7816
標準実装
カードドライバ
JPKI
PCSC / WinSCard
Card Reader
Card

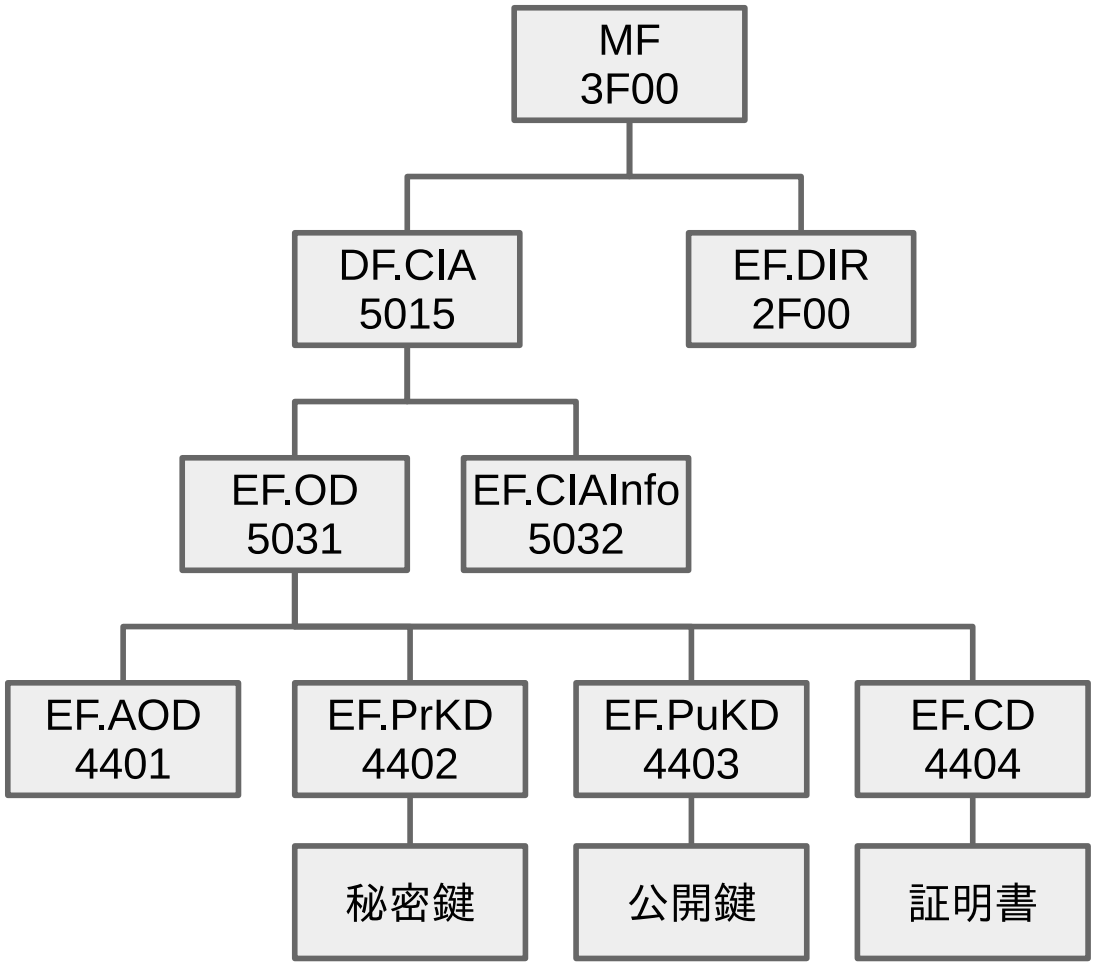
OpenSCカードドライバ

- esteid(エストニア)
- Belpic(ベルギー)
- DNIe(スペイン)
- CNS(イタリア)
- pteID(ポルトガル)
- PIV(米国)

PKCS#11 APIs

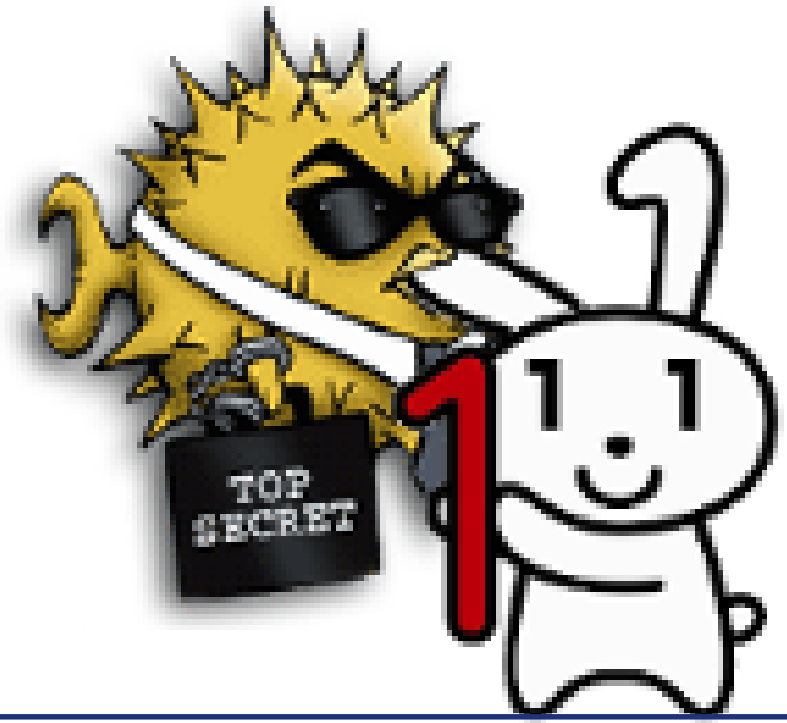
- C_Initialize()
- C_FindObjects()
- C_Login()
- C_Sign()
- C_Finalize()

PKCS#15エミュレーション

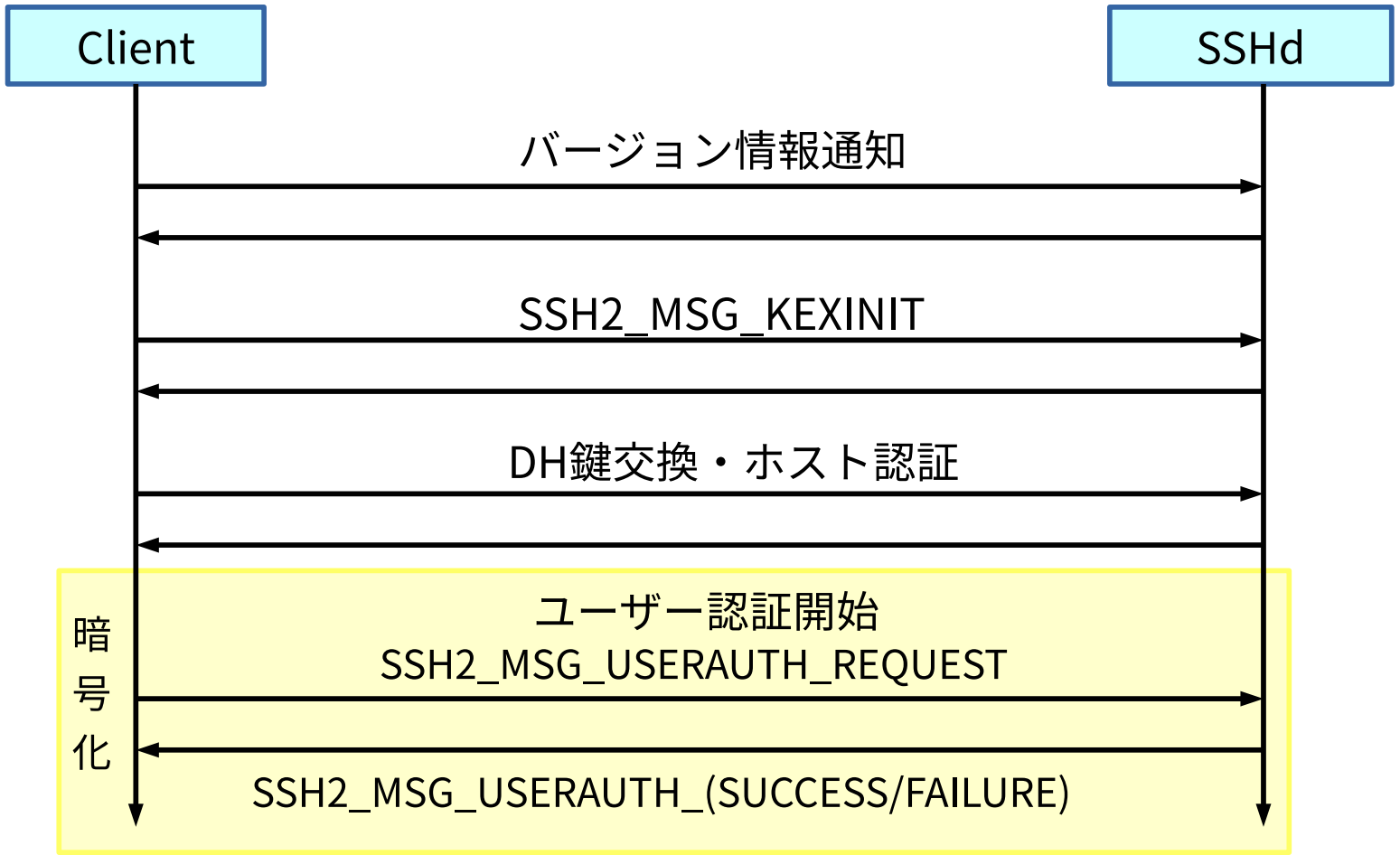


マイナンバーカードでSSH

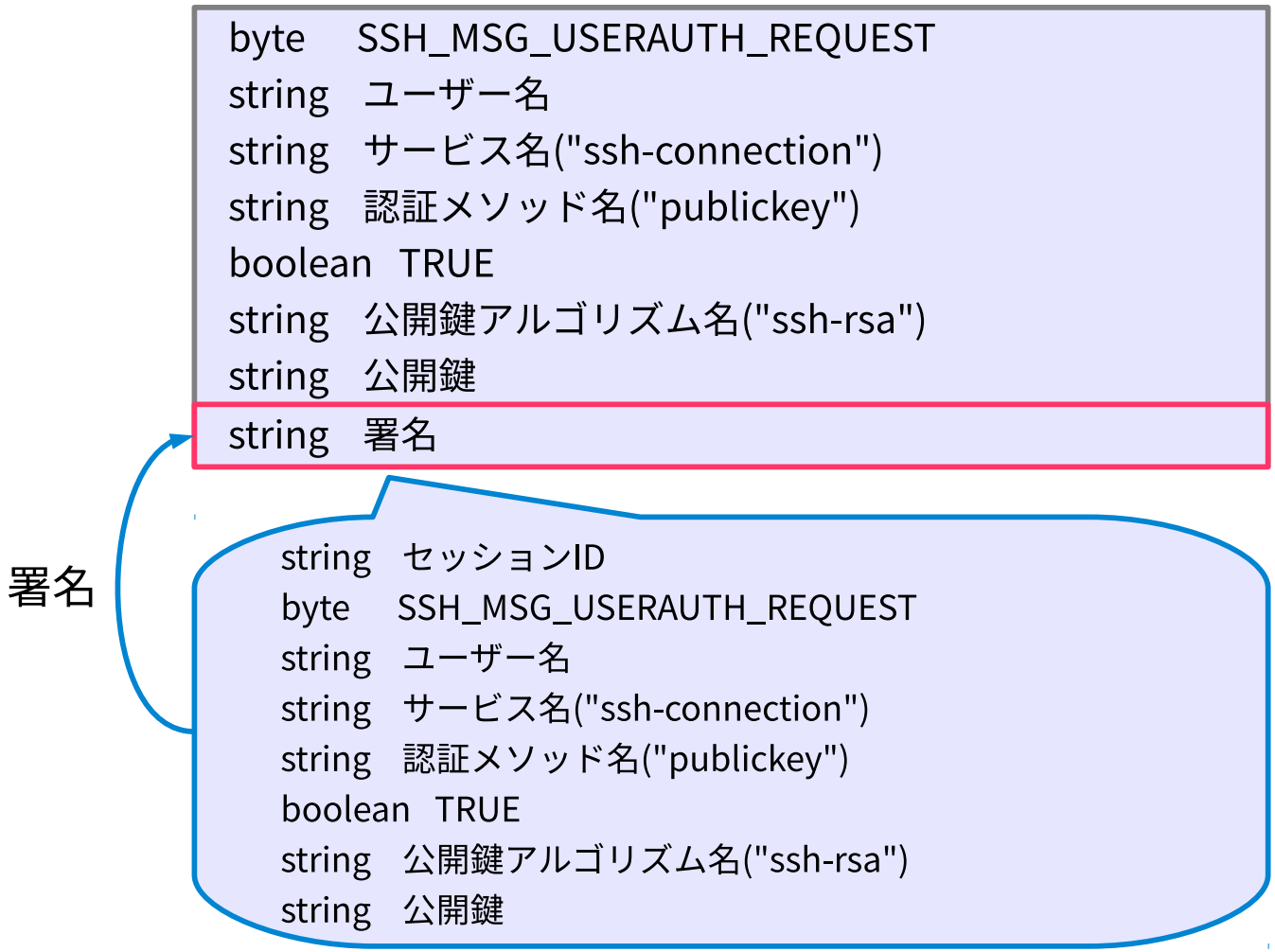
https://www.osstech.co.jp/~hamano/posts/jpki-ssh/



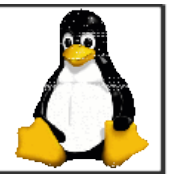
SSHプロトコル


Client
SSHd
バージョン情報通知
SSH2_MSG_KEXINIT
DH鍵交換・ホスト認証
暗号化
ユーザー認証開始
SSH2_MSG_USERAUTH_REQUEST
SSH2_MSG_USERAUTH_(SUCCESS/FAILURE)

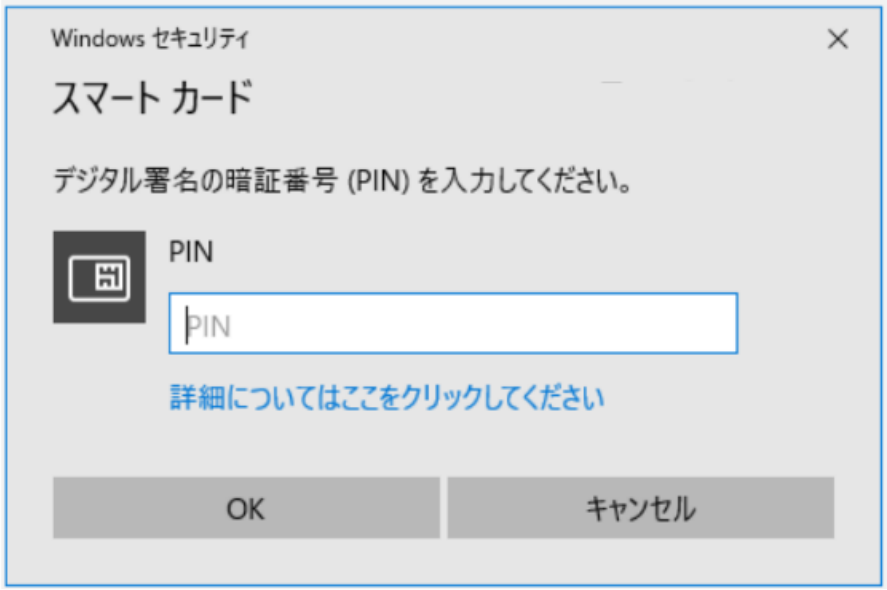
SSH公開鍵認証


byte SSH_MSG_USERAUTH_REQUEST
string ユーザー名
string サービス名("ssh-connection")
string 認証メソッド名("publickey")
boolean TRUE
string 公開鍵アルゴリズム名("ssh-rsa")
string 公開鍵
string 署名
署名
string セッションID
byte SSH_MSG_USERAUTH_REQUEST
string ユーザー名
string サービス名("ssh-connection")
string 認証メソッド名("publickey")
boolean TRUE
string 公開鍵アルゴリズム名("ssh-rsa")
string 公開鍵

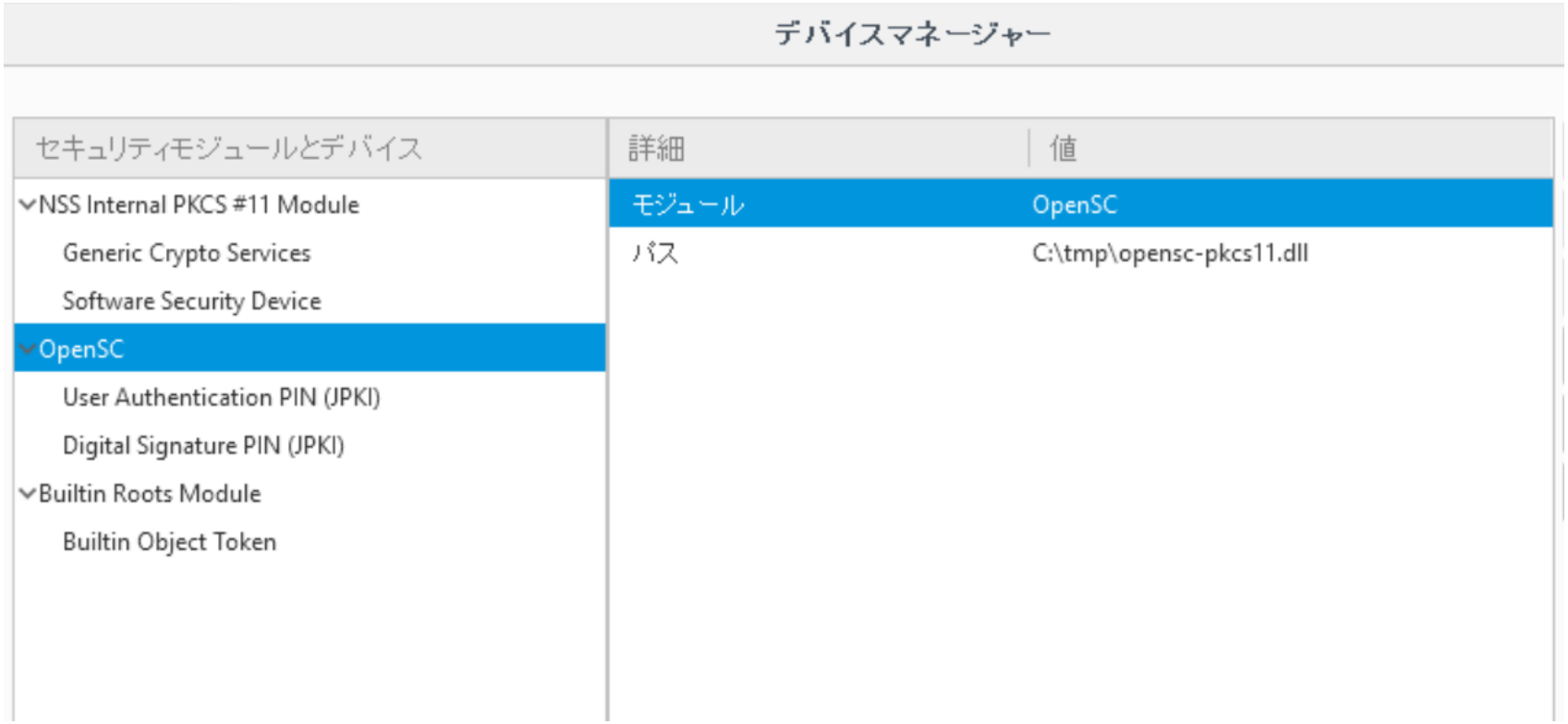
ブラウザ対応状況

			
 Microsoft Edge	✓		
 Internet Explorer 11	✗		
 Safari		✓	
 Google Chrome	✓	✗	✓
 Mozilla Firefox	✓	✓	✓

TLSクライアント認証 with Microsoft Edge



TLSクライアント認証 with Mozilla Firefox



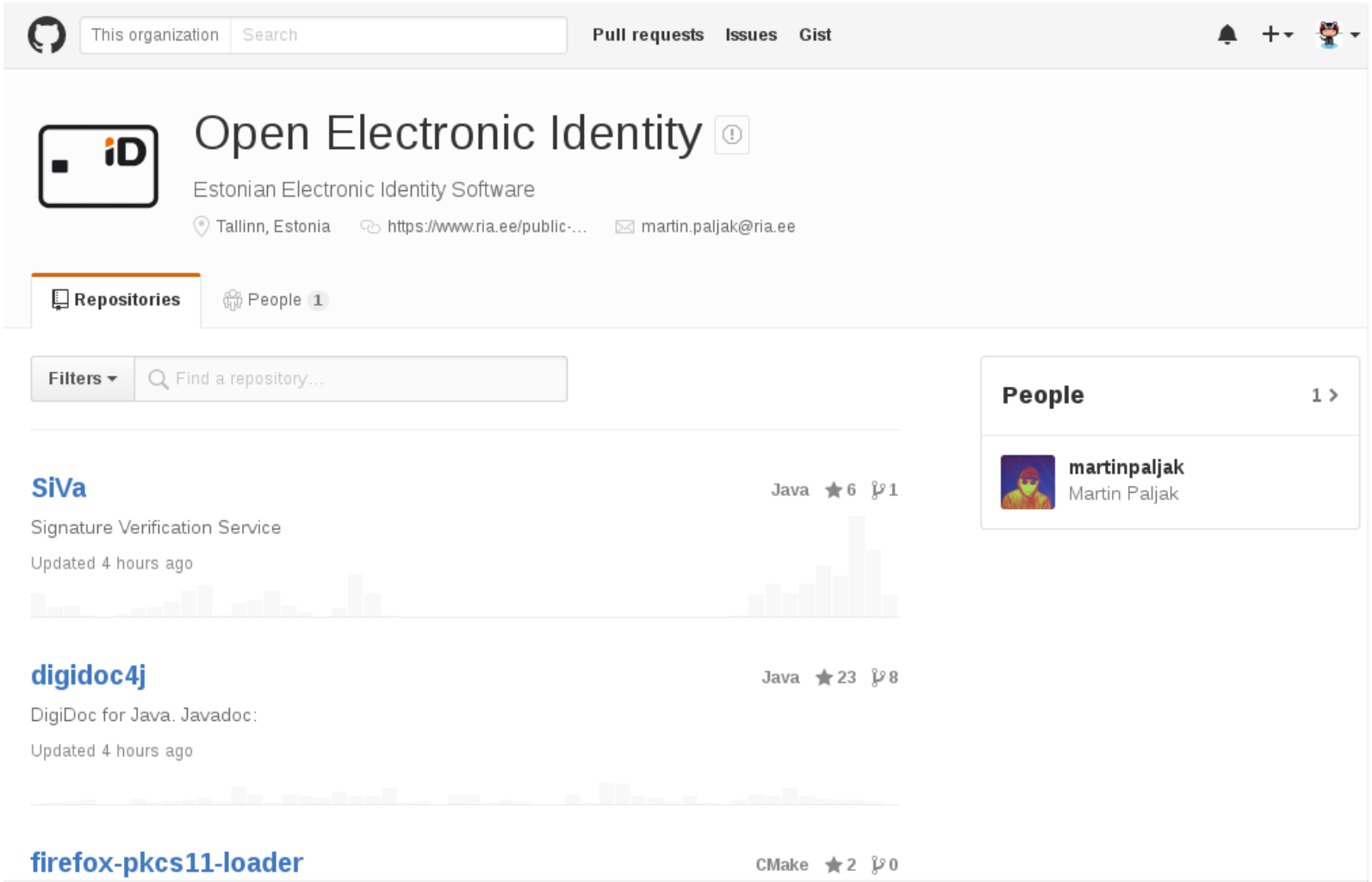
目指すべき社会

- 電子先進国 (エストニア・ベルギー)
  - オープン・スタンダード
  - オープン・ソース
  - OS・ブラウザ非依存
  - 高い普及率と利用率
- 電子後進国 (日本)
  - 仕様が非公開
  - クローズドソース
  - Java Applet(笑)

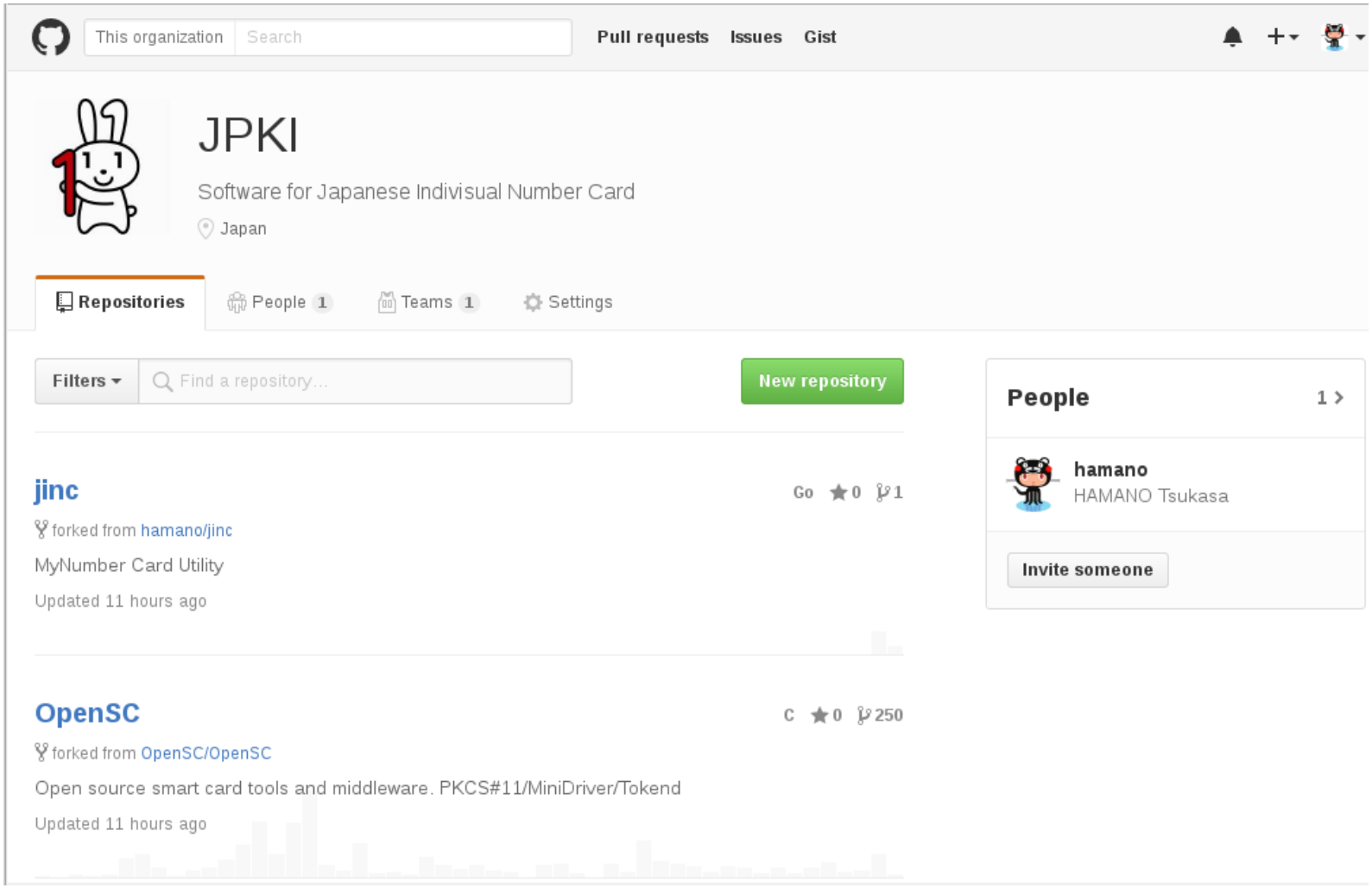
課題

- 運用に対する不安
  - マイナンバーに対する不信
  - 鍵生成の仕組み
  - 仕様公開されないと安心できない
- 失効情報の検証
  - 総務大臣の認可
- 名寄せの問題
  - 信頼できる ID Provider が必要
  - 複数の鍵管理

github.com/open-eid



github.com/JPKI



OpenSCソースレポジトリ

- github.com/OpenSC
  - OpenSC/master
- github.com/JPKI
  - OpenSC/jpki = 0.16.0 + JPKI patch

GNU Lesser General Public License

- アプリ + LGPL ライブラリ別配布
- アプリ + LGPL ライブラリ同梱
- アプリ + LGPL ライブラリ static link

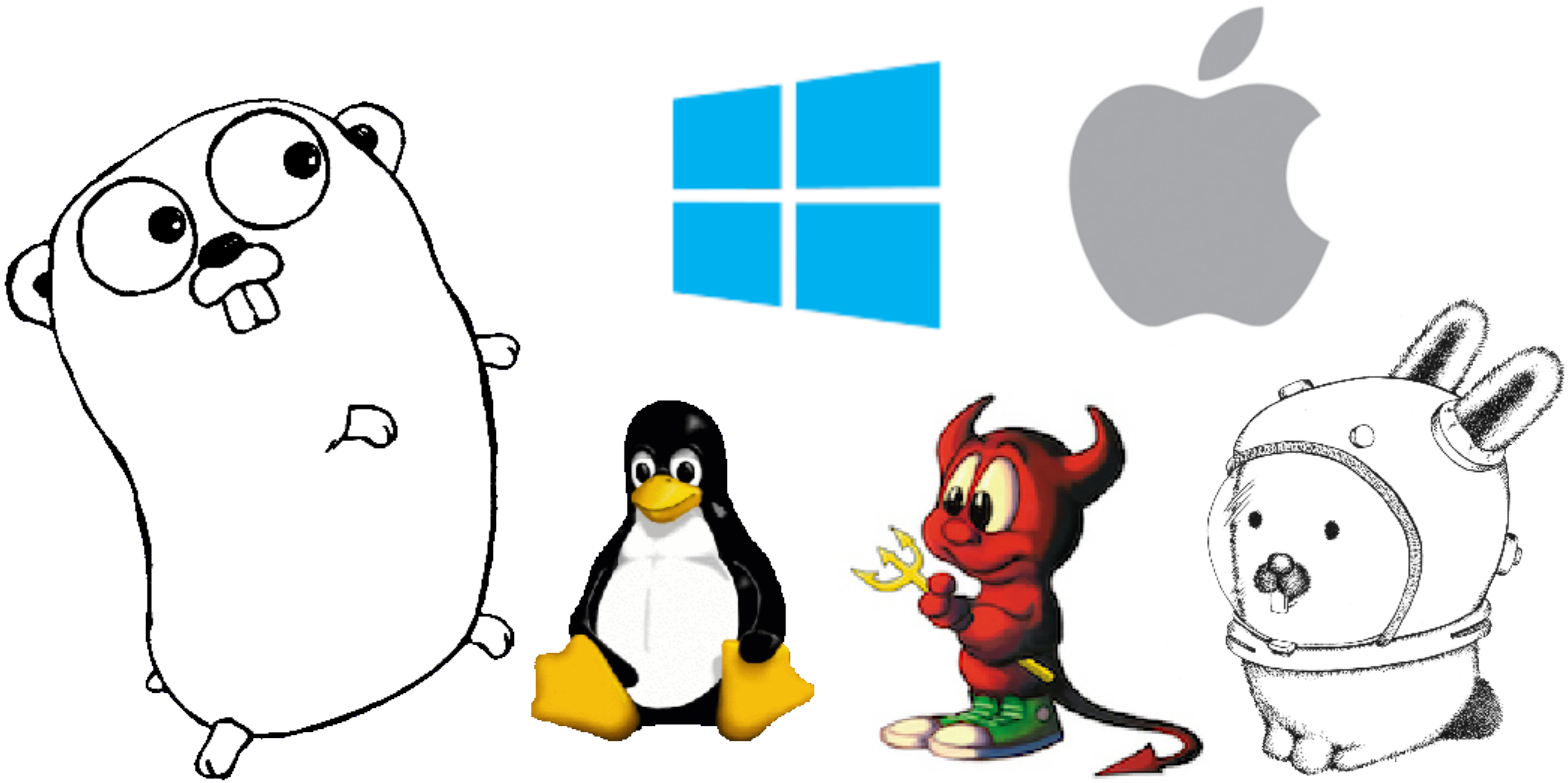
適用例: glibc, GTK+

mynaコマンド

https://github.com/jpki/myna

```
$ myna --help
COMMANDS:
    card          券面事項を表示
    cert          証明書を表示
    sign          CMS署名
    pin_status    PINステータス
```

5種類のOSで動く！



CMS署名(RFC 5652)

```
$ myna sign \
    -i 文書ファイル \
    -o 署名付きファイル
```

検証

```
$ openssl cms -verify \
    -CAfile CA.pem \
    -inform der -in 署名ファイル
```

GUI版もある