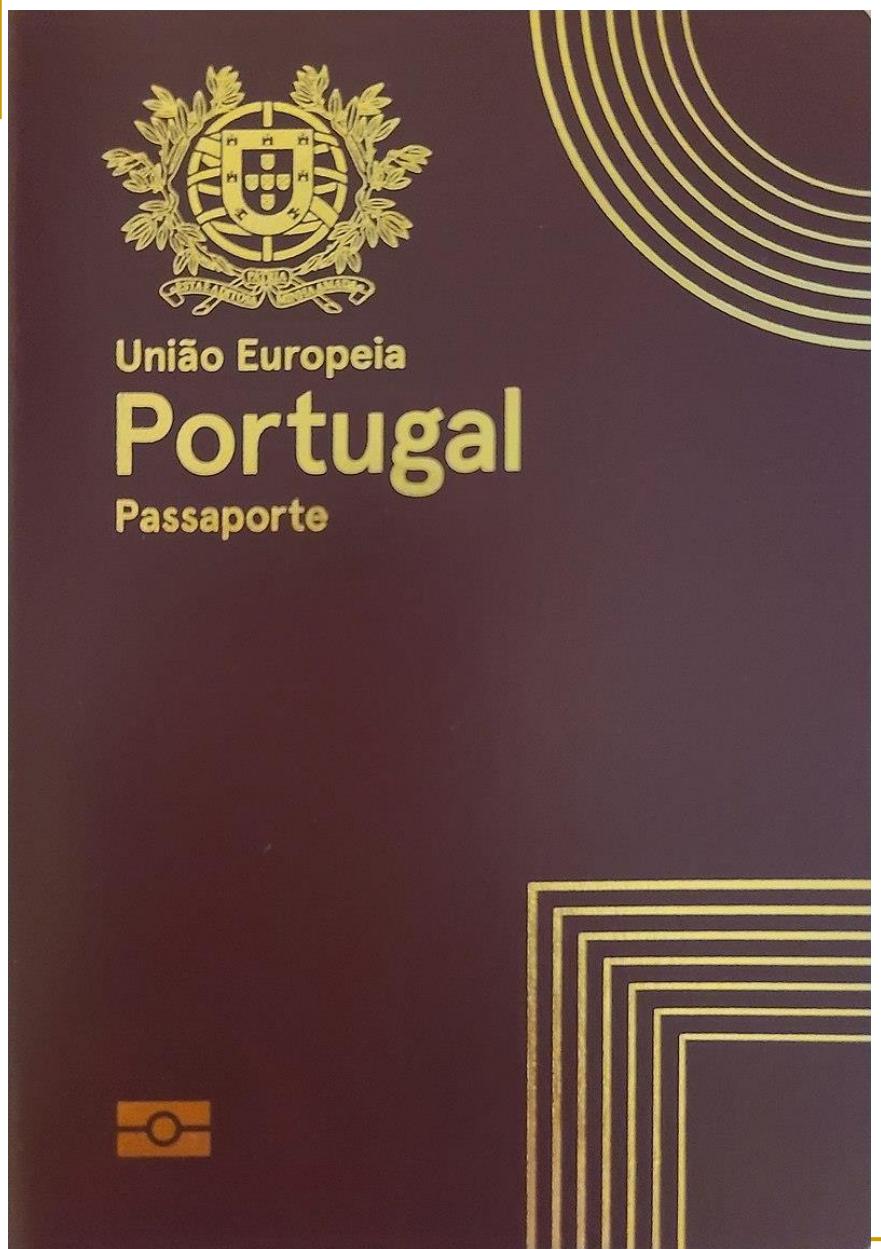
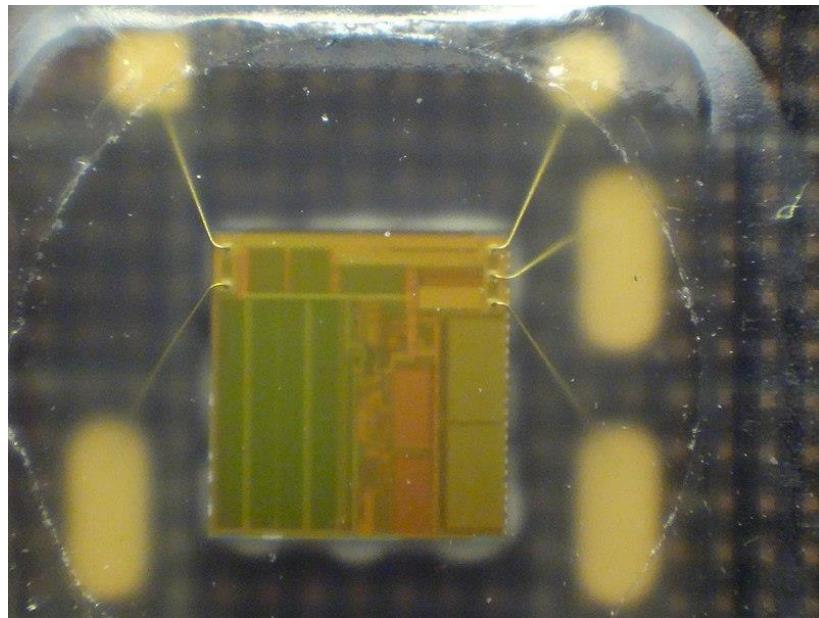
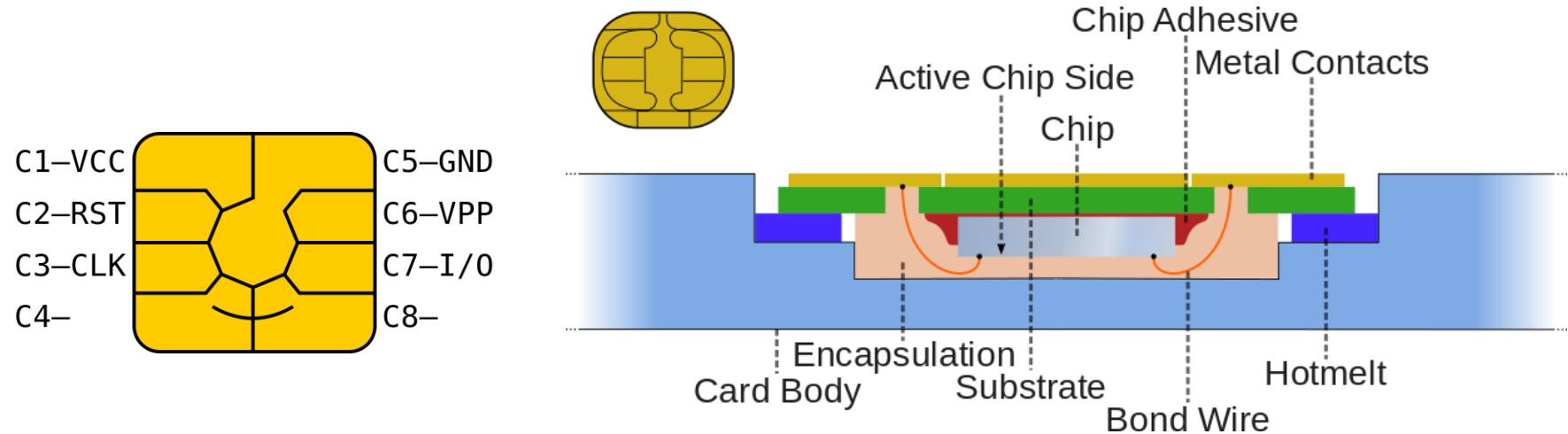


# Smartcards



<https://pplware.sapo.pt/informacao/saiba-como-renovar-online-o-seu-cartao-de-cidadao/>  
<https://knowtechie.com/security-matters-5-benefits-of-contactless-smart-cards/>





# Why smartcards

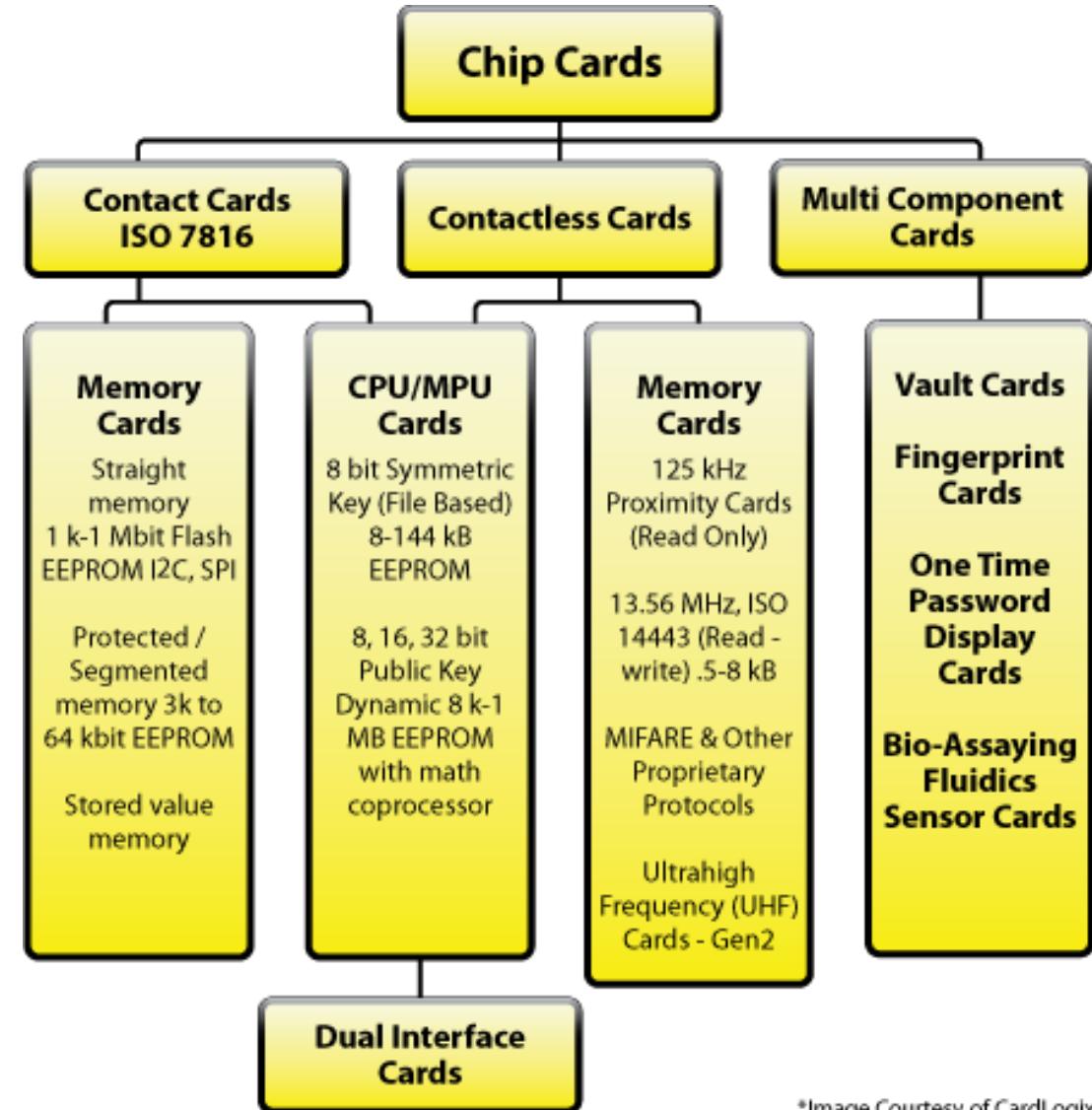
- ▷ Interoperability
  - ◆ Stack, interfaces and infrastructures are shared over many industries
- ▷ Multi-application support
- ▷ Secure transactions
- ▷ User acceptance
- ▷ Accessibility

# Smartcard: Definition

- ▶ Card with computing processing capabilities

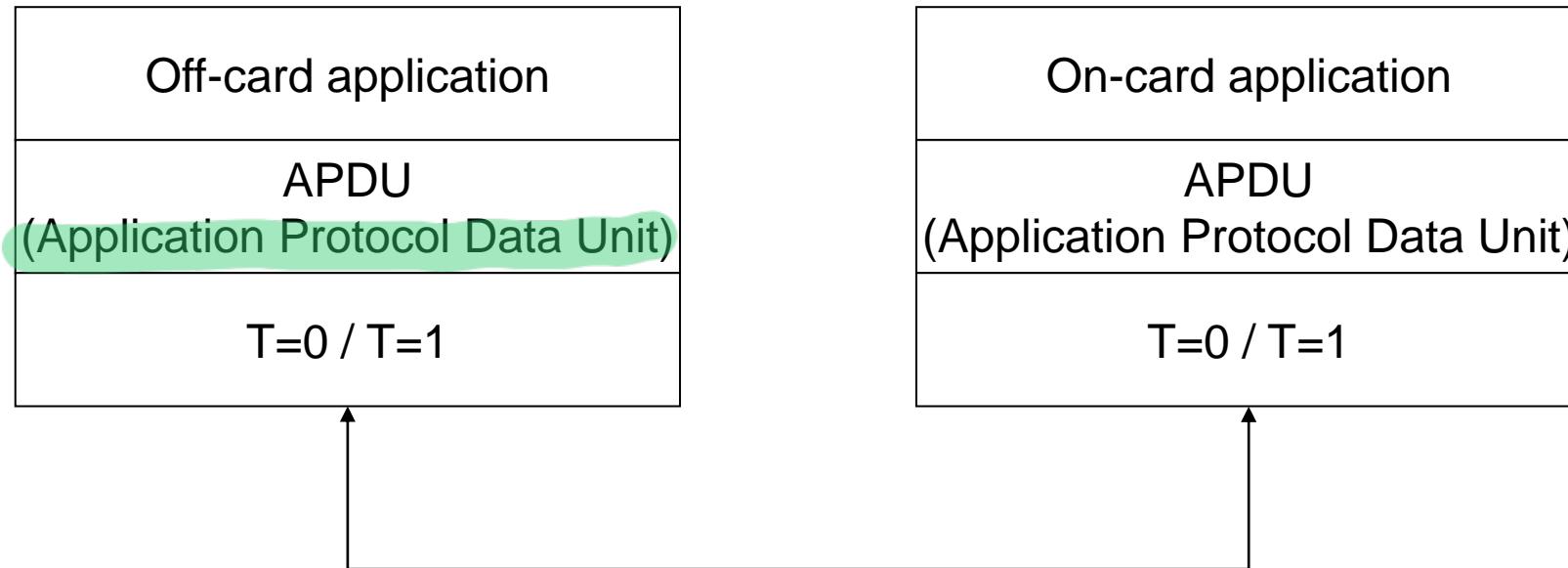
- ◆ CPU
- ◆ ROM
- ◆ EEPROM
- ◆ RAM

- ▶ Interface
  - ◆ With contact
  - ◆ Contactless



\*Image Courtesy of CardLogix

# Smartcard applications: Communication protocol stack



# T=0 and T=1

- ▷ T=0
  - ◆ Each byte transmitted separately
  - ◆ Slower
- ▷ T=1
  - ◆ Blocks of bytes transmitted
  - ◆ Faster
- ▷ ATR (ISO 7816-3)
  - ◆ Response of the card to a reset operation
  - ◆ Reports the protocol expected by the card

ATR: 3B 7D 95 00 00 80 31 80 65 B0 83 11 00 C8 83 00 90 00

- + TS = 3B --> Direct Convention
- + T0 = 7D, Y(1): 0111, K: 13 (historical bytes)
- TA(1) = 95 --> Fi=512, Di=16, 32 cycles/ETU  
    125000 bits/s at 4 MHz, fMax for Fi = 5 MHz => 156250 bits/s
- TB(1) = 00 --> VPP is not electrically connected
- TC(1) = 00 --> Extra guard time: 0
- + Historical bytes: 80 31 80 65 B0 83 11 00 C8 83 00 90 00
- Category indicator byte: 80 (compact TLV data object)
  - Tag: 3, len: 1 (card service data byte)
    - Card service data byte: 80
      - Application selection: by full DF name
      - EF.DIR and EF.ATR access services: by GET RECORD(s) command
      - Card with MF
  - Tag: 6, len: 5 (pre-issuing data)
    - Data: B0 83 11 00 C8
  - Tag: 8, len: 3 (status indicator)
    - LCS (life card cycle): 00 (No information given)
    - SW: 9000 (Normal processing.)

Possibly identified card (using /usr/share/pcsc/smartcard\_list.txt):

3B 7D 95 00 00 80 31 80 65 B0 83 11 00 C8 83 00 90 00

3B 7D 95 00 00 80 31 80 65 B0 83 11 .. . 83 00 90 00

Portuguese ID Card (eID)

<http://www.cartaoeconomia.pt/>

# APDU (ISO 7816-4)



## ▷ Command APDU

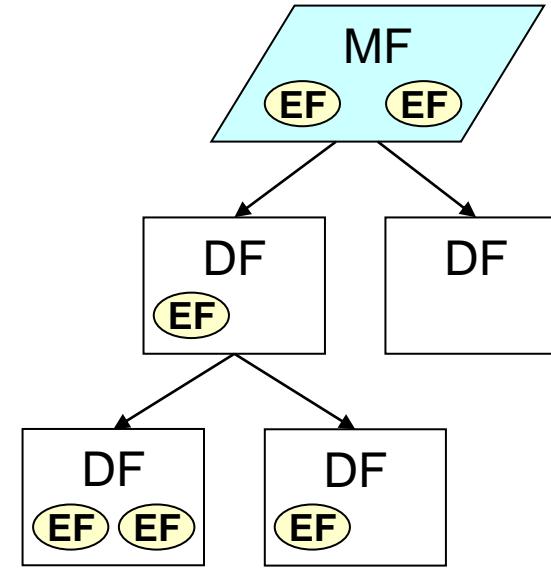
- ♦ CLA (1 byte)
  - Class of the instruction
- ♦ INS (1 byte)
  - Command
- ♦ P1 and P2 (2 bytes)
  - Command-specific parameters
- ♦ Lc
  - Length of the optional command data
- ♦ Le
  - Length of data expected in subsequent Response APDU
  - Zero (0) means all data available

## ▷ Response APDU

- ♦ SW1 and SW2 (2 bytes)
  - Status bytes
  - 0x9000 means SUCCESS

# Smartcard: File system

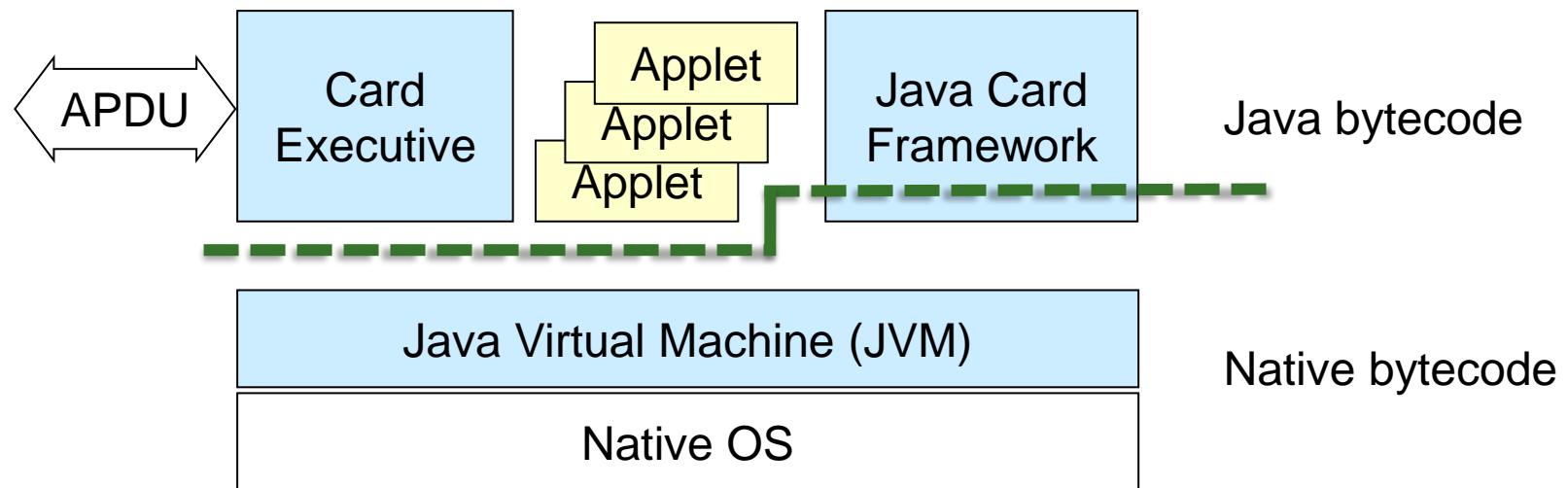
- ▷ File identification
  - ◆ Name or number
- ▷ File types
  - ◆ Master File (MF)
    - File system root, ID 0x3F00
  - ◆ Dedicated File (DF)
    - Similar to a directory
    - Can contain other EFs or DF
  - ◆ Elementary File (EF)
    - Ordinary data file
    - File size fixed and determined when created



# Java cards

- ▷ Smartcards that run Java Applets
  - ◆ That use the JCRE
  - ◆ The JCRE runs on top of a native OS
- ▷ JCRE (Java Card Runtime Environment)
  - ◆ Java Virtual Machine
  - ◆ Card Executive
    - Card management
    - Communications
  - ◆ Java Card Framework
    - Library functions

# Java cards



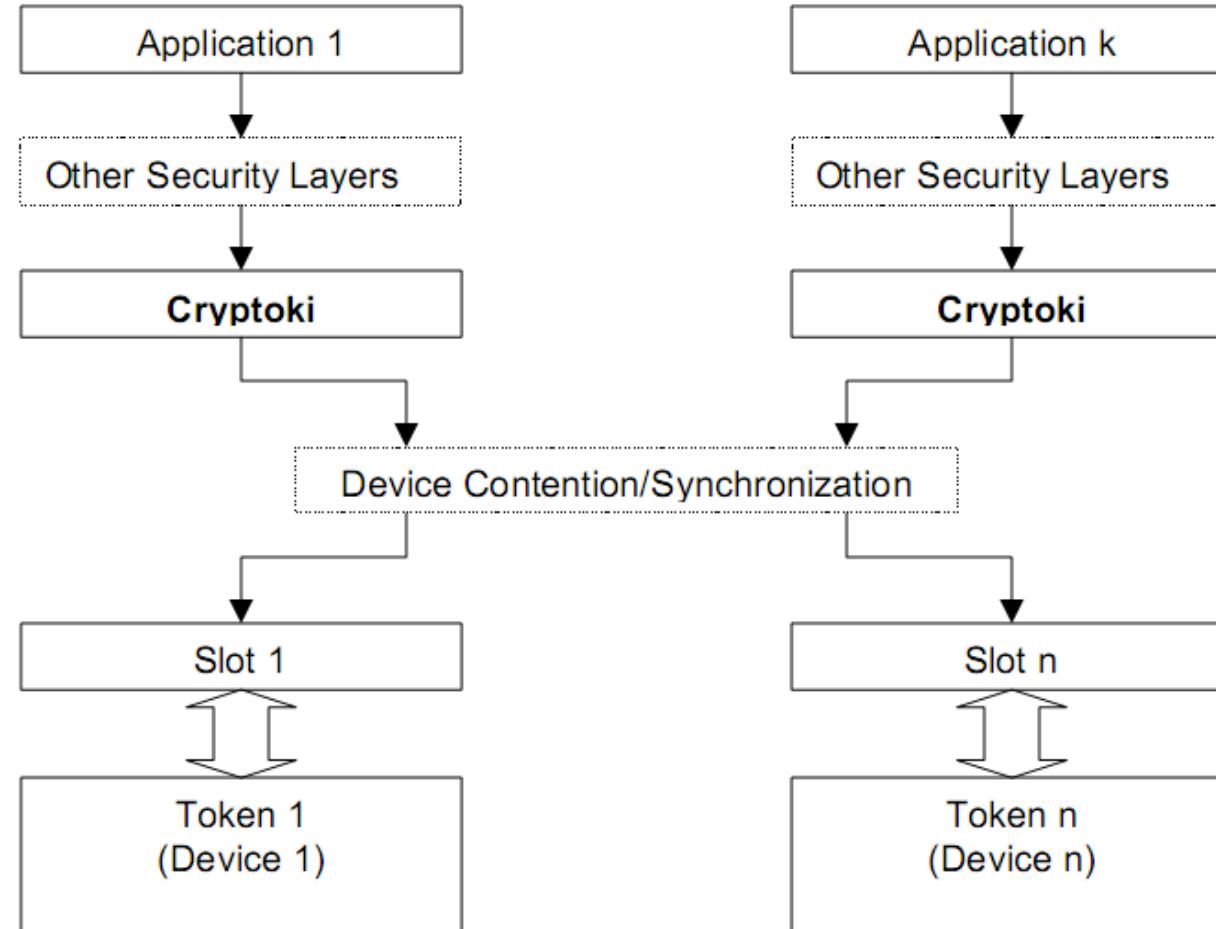
# Cryptographic services

- ▷ Ciphers
- ▷ Digest functions
- ▷ Key generation
- ▷ Key management
  - ◆ Key import
  - ◆ Key export
- ▷ Digital signatures
  - ◆ Generation
  - ◆ Verification
- ▷ Management of public key certificates
  - ◆ Generation
  - ◆ Verification

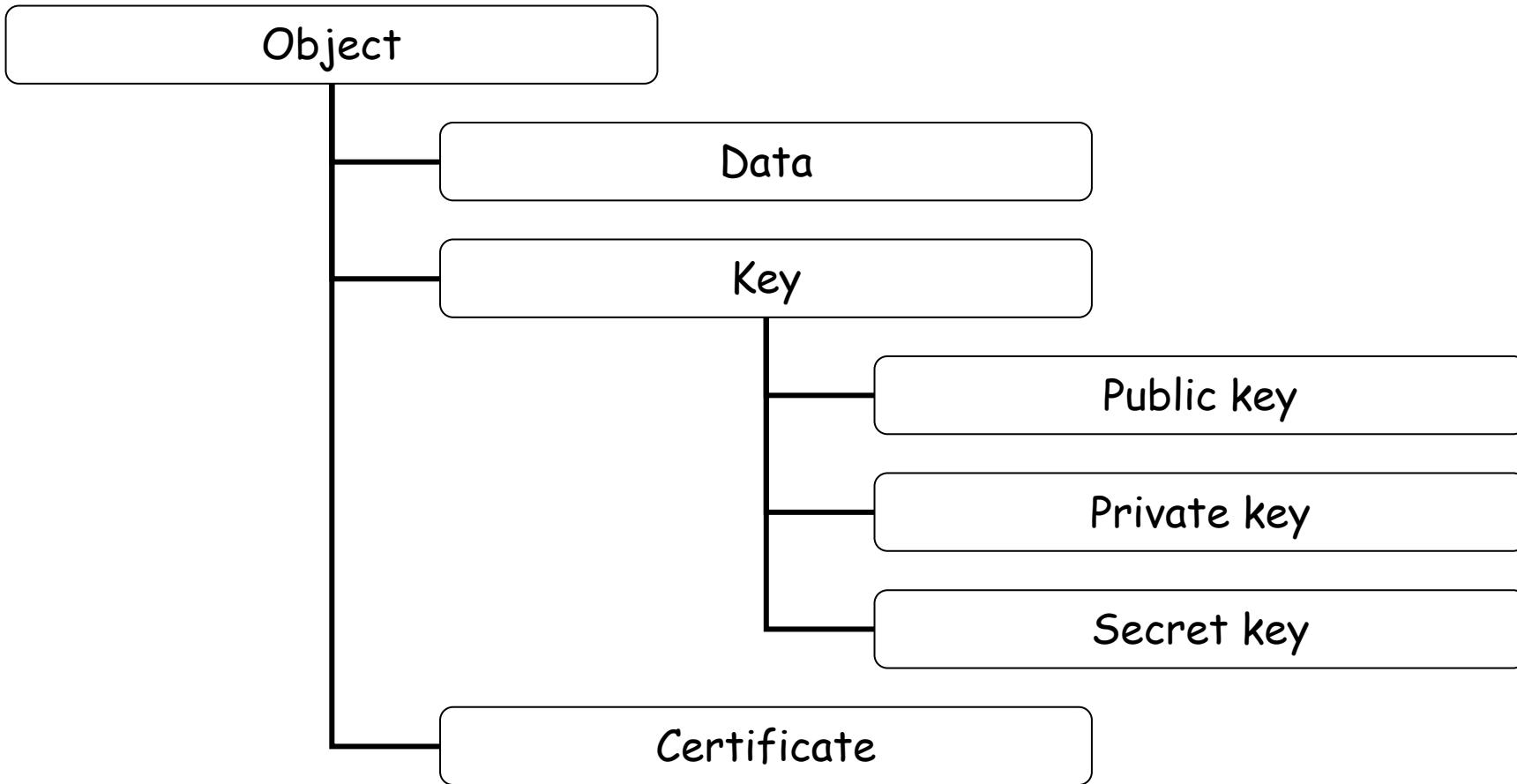
# Cryptographic services: Middleware

- ▷ Libraries that bridge the gap between functionalities of smartcards and high-level applications
- ▷ Some standard approaches:
  - ◆ PKCS #11
    - Cryptographic Token Interface Standard (Cryptoki)
    - Defined by RSA Security Inc.
  - ◆ PKCS #15
    - Cryptographic Token Information Format Standard
    - Defined by RSA Security Inc.
  - ◆ CAPI CSP
    - CryptoAPI Cryptographic Service Provider
    - Defined by Microsoft for Windows systems
  - ◆ PC/SC
    - Personal computer/smartcard
    - Standard framework for smartcard access on Windows systems

# PKCS #11: Cryptoki middleware integration



# PKCS #11: Cryptoki object hierarchy

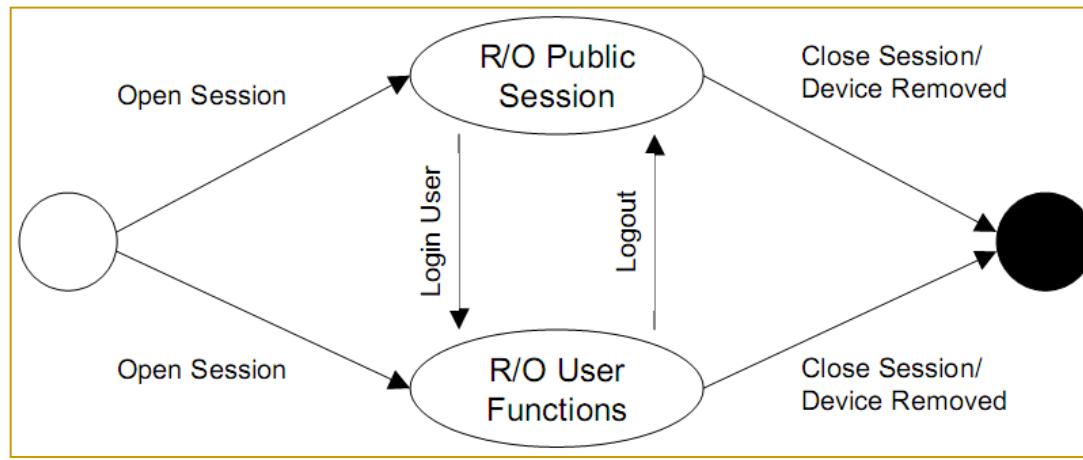


# PKCS #11: Cryptoki sessions

- ▷ Logical connections between applications and tokens
  - ◆ R/O and R/W sessions
  - ◆ Session owners
    - Public
    - User
    - Security Officer (SO)
- ▷ Lifetime of sessions
  - ◆ Usually for a single operation on the token
- ▷ Operations on open sessions
  - ◆ Administrative
    - Login/logout
  - ◆ Object management
    - Create / destroy an object on the token
  - ◆ Cryptographic
- ▷ Session objects
  - ◆ Transient objects created during sessions

# PKCS #11:

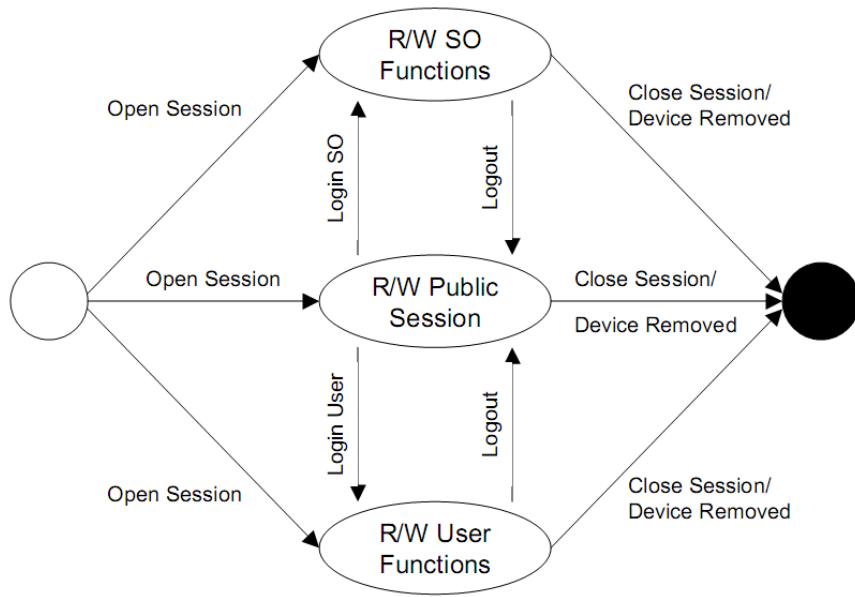
## Cryptoki R/O sessions login/logout



- ▷ R/O public session
  - ◆ Read-only access to public token objects
  - ◆ Read/write access to public session objects
- ▷ R/O user functions
  - ◆ Read-only access to all token objects (public or private)
  - ◆ Read/write access to all session objects (public or private)

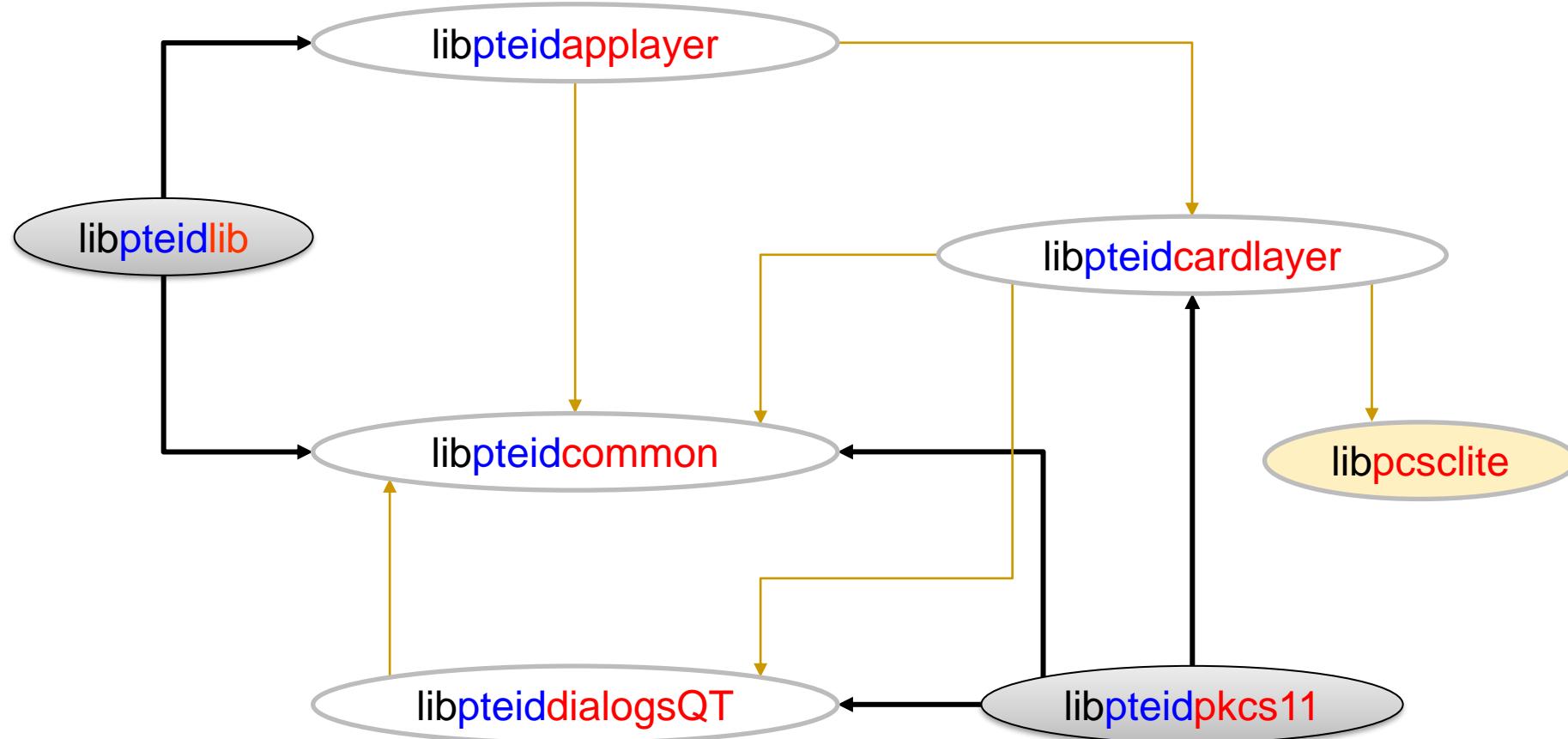
# PKCS #11:

## Cryptoki R/W sessions login/logout

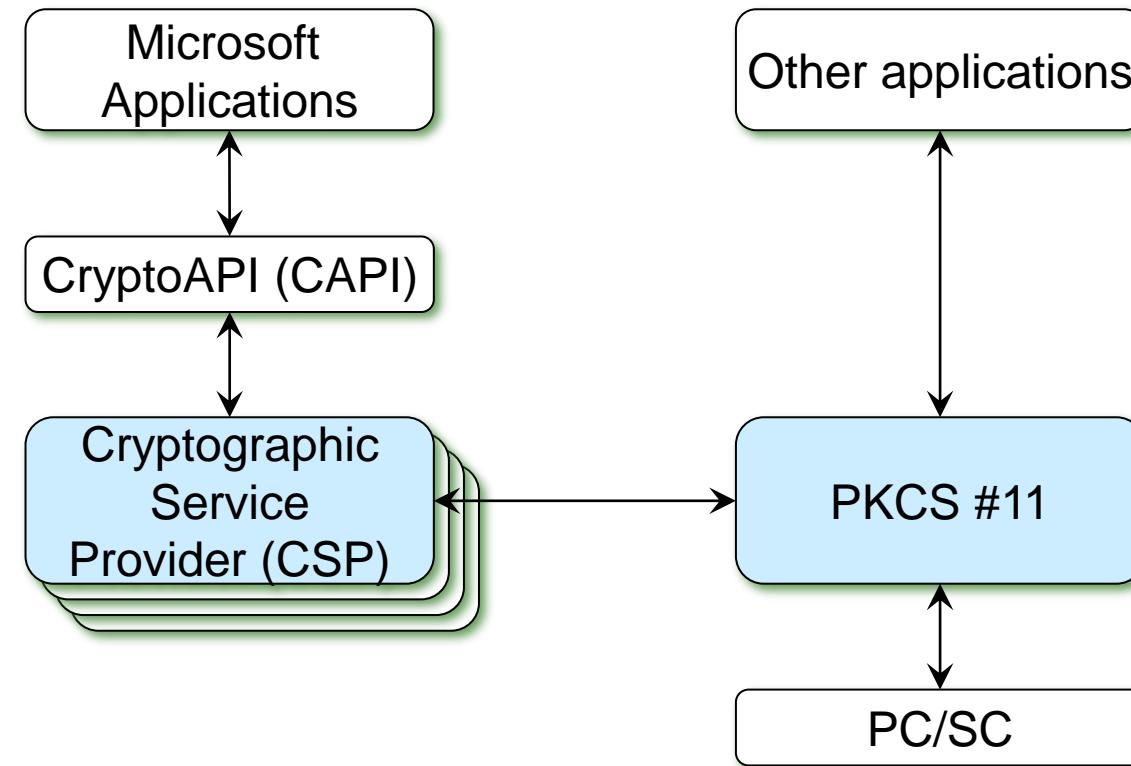


- ▷ R/W public session
  - Read/write access to all public objects
- ▷ R/W SO functions
  - Read/write access only to public objects on the token
    - Not to private objects
  - The SO can set the normal user's PIN
- ▷ R/W user functions
  - Read/write access to all objects

# Cartão de Cidadão: Middleware for Unix (Linux/MacOS)



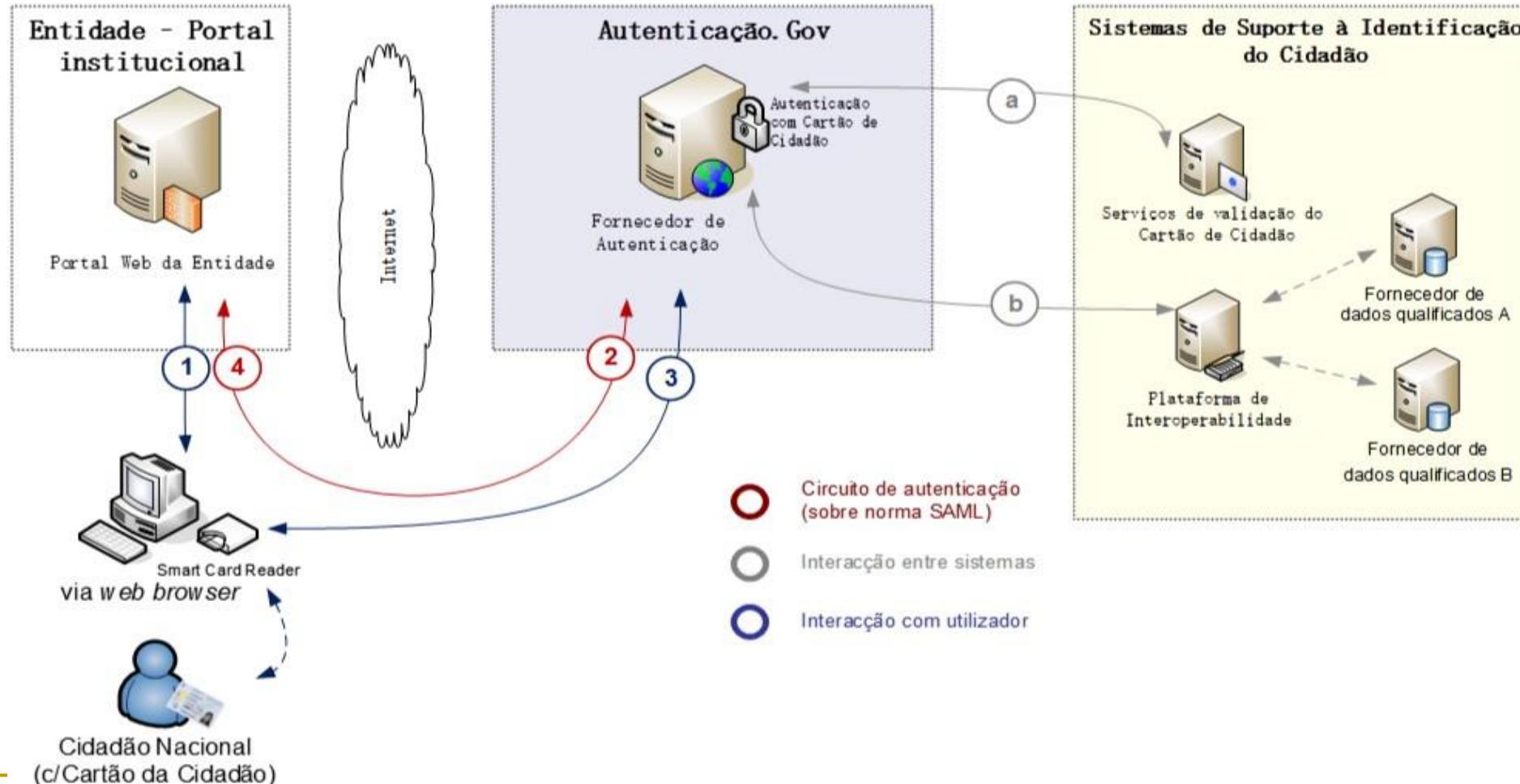
# Cartão de Cidadão: Middleware for Windows

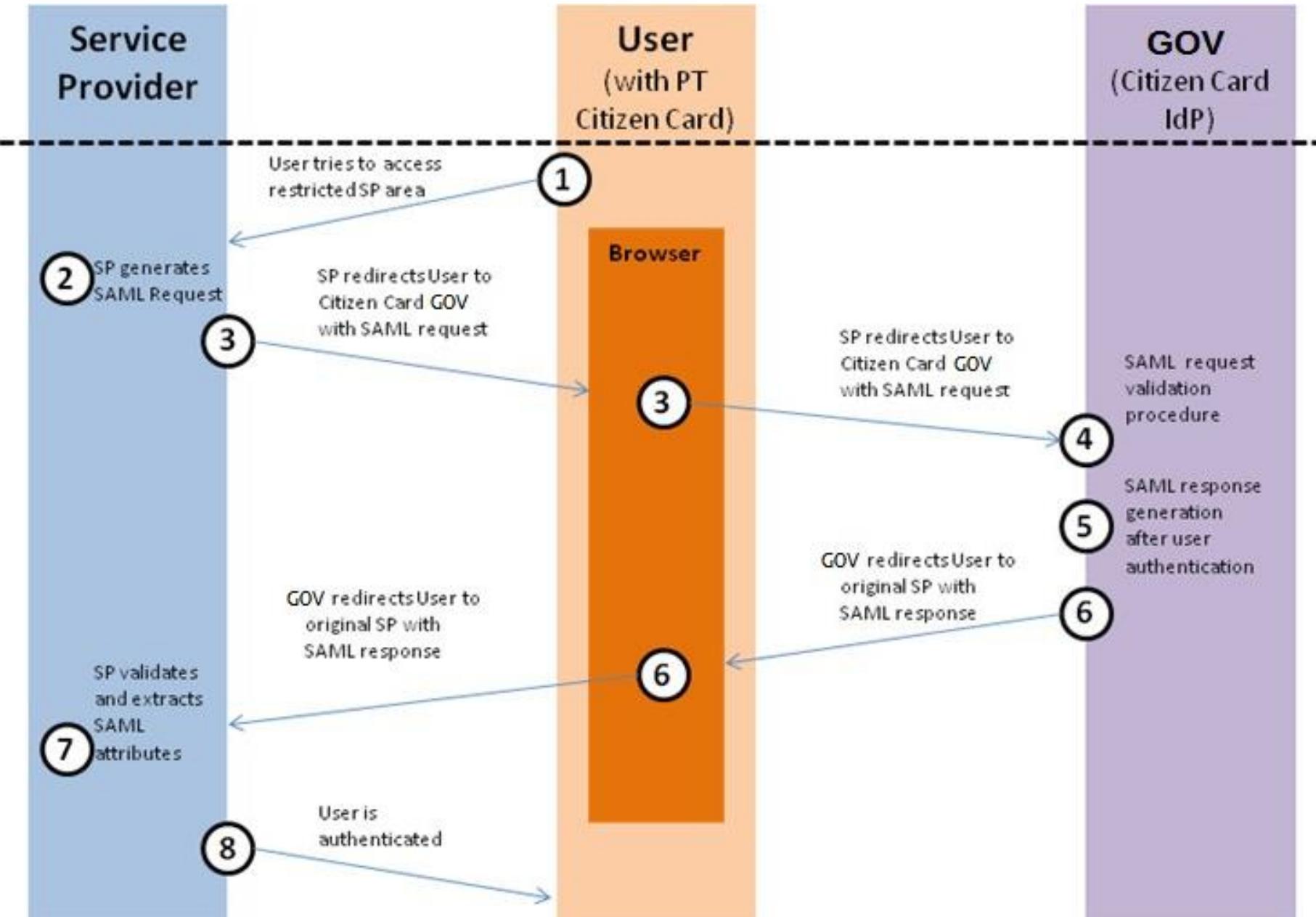


# Authentication with the PTEID

- ▷ Authenticator sends NONCE to the CC to be signed with the private key
- ▷ Issue: Browser do not have direct access to the CC
  - ◆ Possible to configure libpteidpkcs11.so, limited to the PKCS#11 API
  - ◆ Possible to use a java applet (obsolete)
- ▷ Solution: Use a plugin installed in the user computer
  - ◆ Exposes a web server to the localhost
  - ◆ Allows access to the card through the web server
    - Only to authenticated requests through the CC infrastructure
  - ◆ Required the previous approval of each new integration

# PT Authentication Plugin





# Mobile Digital Key (CMD)/Virtual Smart Card

- ▷ Objective: allow authentication/signature even without the smartcard
  - ◆ But with a similar level of security
- ▷ Operation principles
  - ◆ Requires a smartcard to authenticate the request of a CMD
  - ◆ Users can authenticate themselves/sign documents using the CMD
  - ◆ Doesn't require any plugin installed
  - ◆ Doesn't require the card in future uses
  - ◆ Uses 2FA: PIN code in the site + code through another channel
    - E.g: SMS or Twitter

