

# Smartcard Jungle

Using security devices should be easy.

Jean-Michel Pouré  
[Http://www.gooze.eu](http://www.gooze.eu)

# Hands-up!

- How many of your friends use:
  - smartcards,
  - crypto sticks,
  - or one-time passwords (OTPs)?
- How many of your friends use credit cards with a chip?

# Preliminary question

- Can you explain why:
  - **so many** people use credit cards ?
  - And **so few** people use security devices?

# Possible answers

- Denial: we don't need security devices.
- Patents: destroyed the market.
- Budgets and size: only large companies.
  
- We may have to admit ...

# Possible answers

- It should be possible to improve the integration and usability of security devices.
- This is one of the reason that we are all here in Brussels to discuss about security at FOSDEM.

# Plan

- Part 1: Hardware and standards.
- Part 2: Operating systems.
- Part 3: Applications.
- Conclusion: get a free smartcard.

# Part 1

Hardware and standards

# Hardware

- Smartcards and tokens



+



=



# Hardware

- Smartcards and tokens:
  - May preserve a secret (RSA key).
  - May compute secrets without displaying them.
  - Are inside your wallet (you know you have them).
  - May be destroyed if opened.
- Standards:
  - PCSC: smartcards.
  - PKCS#11: interface.
  - PKCS#15: information formats.

# Hardware

- One-time password (OTP) generators.



# Hardware

- OATH (Open AuTHentication)
  - HOTP: event-base
    - RFC 4226
  - TOTP: time-based
    - RFC 4226 extension
- Several implementations.

# Part 2

## Operating Systems

# Operating systems

We will focus on libraries  
and take the example of single-sign-on  
using smartcards.

# Windows Vista/7

- One single API: WinScard
  - CSP + CAPI interfaces.
- No PKCS#11 direct interface
  - Each vendor provides a PKCS#11 interface.
  - OpenSC PKCS#11 is available.

# Windows Vista/7

- Smartcard logon can be implemented with Windows 2003 server + CSP interface.
  - => This kills the market for smartcards.
- You may use the free software alternative MySmartLogon.
  - => Excellent interface.

Control Panel > Security > Search

- Control Panel Home
- System and Maintenance
- Security**
- Network and Internet
- Hardware and Sound
- Programs
- User Accounts and Family Safety
- Appearance and Personalization
- Clock, Language, and Region
- Ease of Access
- Additional Options
- Classic View

**Recent Tasks**

- Allow a program through Windows Firewall
- Uninstall a program
- Change how Windows searches

**Security Center**  
Check for updates | Check this computer's security status | Turn automatic updating on or off | Check firewall status | Require a password when the computer wakes

**Windows Firewall**  
Turn Windows Firewall on or off | Allow a program through Windows Firewall

**Windows Update**  
Turn automatic updating on or off | Check for updates | View installed updates

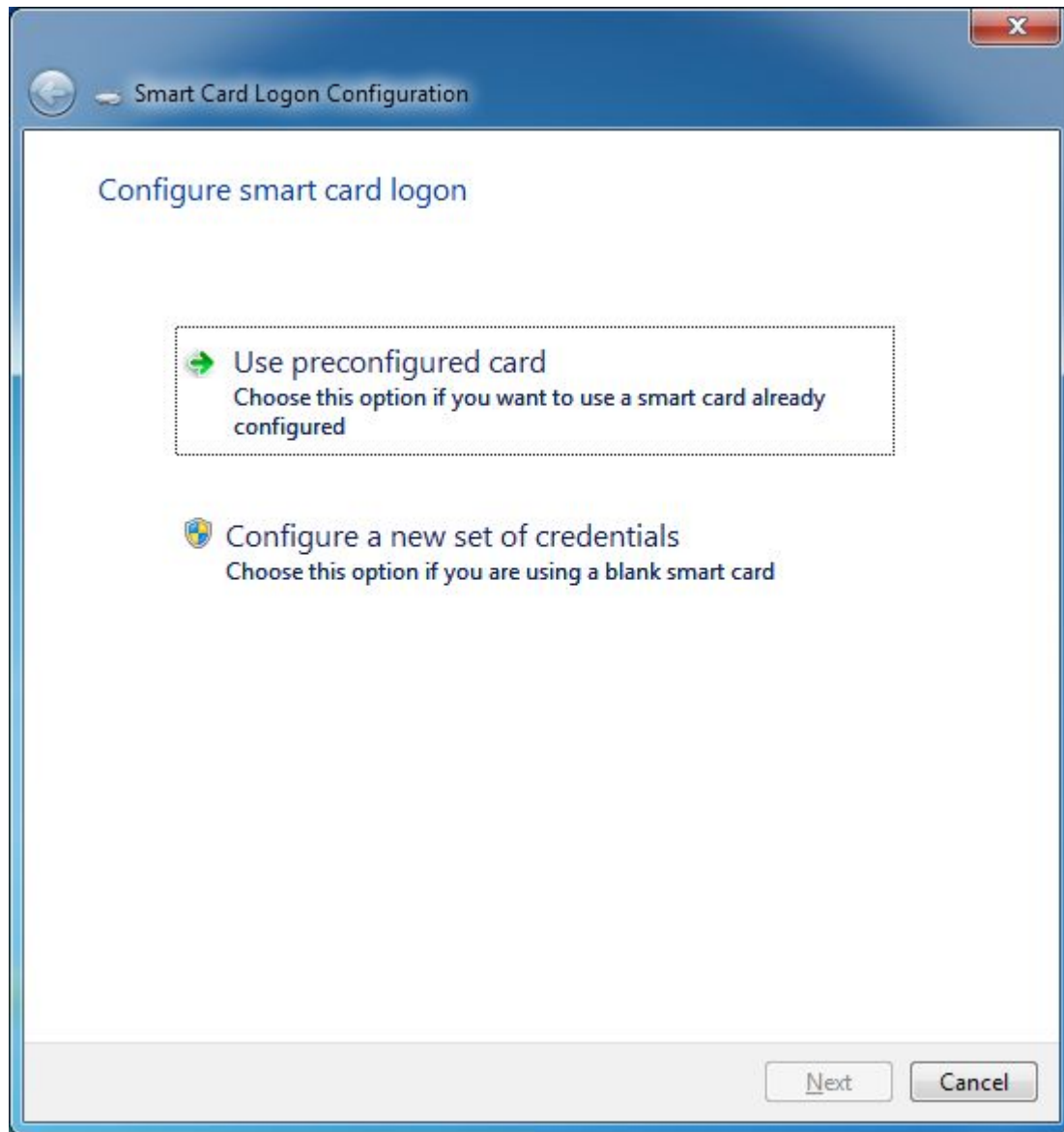
**Windows Defender**  
Scan for spyware and other potentially unwanted software

**Internet Options**  
Change security settings | Delete browsing history and cookies | Manage browser add-ons

**Parental Controls**  
Set up parental controls for any user | View activity reports

**BitLocker Drive Encryption**  
Protect your computer by encrypting data on your disk | Manage BitLocker keys

**Smart Card Logon**



Smart Card Logon Configuration

### Configure a smart card

This wizard will create a certificate on the smart card based on a certification authority. Please select a certification authority

Create a new certification authority

Use this certification authority

Selected authority :

Object : ADIANT-PCTEST Delivered : Wednesday, December 30, 2009 5:20:25 PM Expires : Monday, December 30, 2019 5:20:25 PM	Show the certificate
	Select an authority

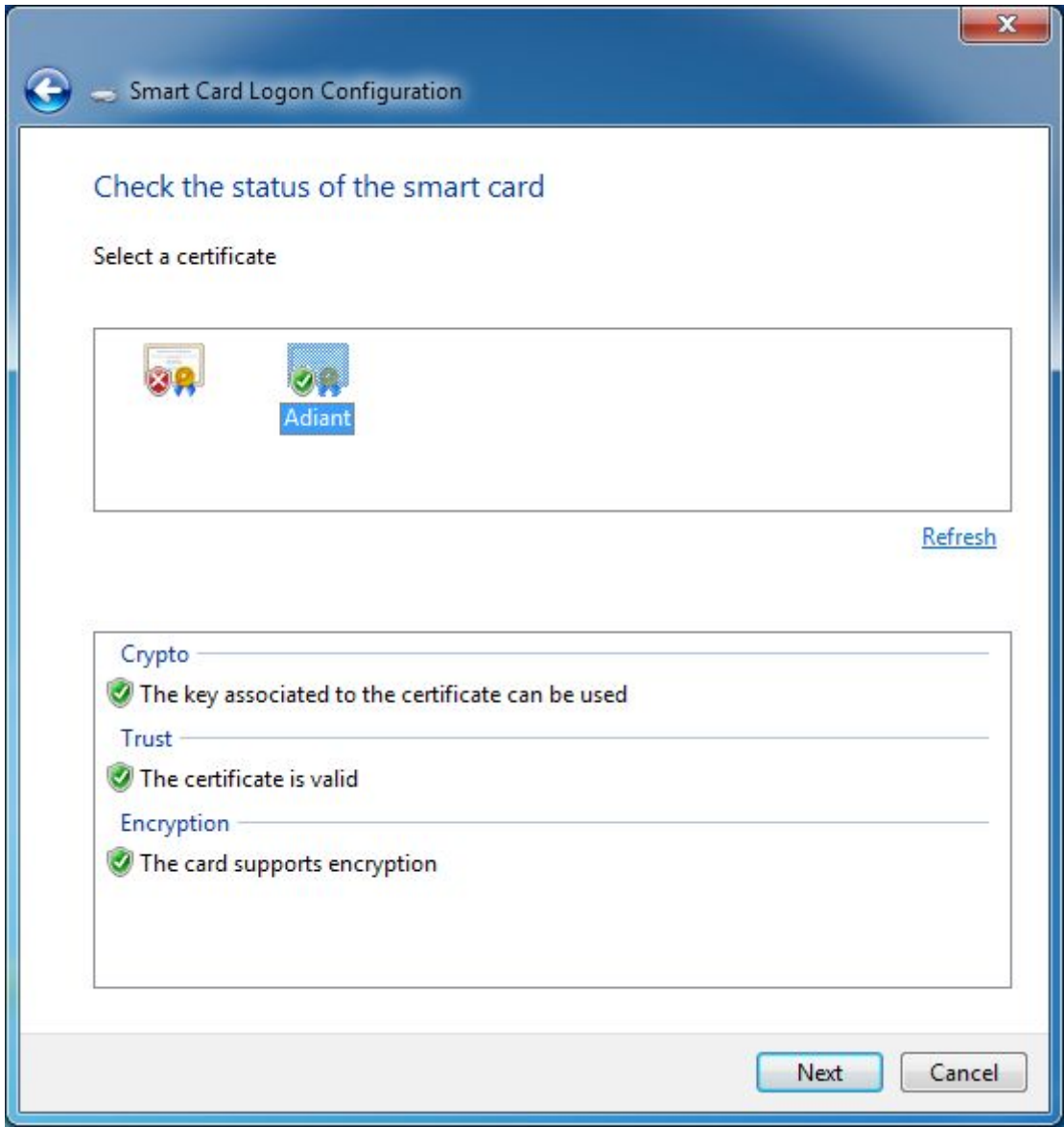
Import a p12 file into the smart card

File :  ...

Password :

Delete all certificates on the smart card before processing

Next Cancel





### Smart card PIN change

SCM Microsystems Inc. SCR33x USB Smart Card Reader 0

Enter your old PIN and your new PIN.

PIN

New PIN

New PIN confirmation



# GNU/Linux and Unixes

- PCSC muscle framework:
  - CCID subsystem (standard).
- OpenSC
  - PKCS#11 library.
  - OpenSC utilities.

# GNU/Linux and Unixes

- Single-logon:
  - Pam-p11
    - SSH mapper.
  - Pam-pkcs11
    - LDAP, SSH, Kerberos mappers.
- No graphical interface for setting up single-logon

# Mac OS X

- PCSC muscle framework.
- Tokend framework.

# Mac OS X

- Single log-on
  - Terrible to set-up using XML files.
  - A lot of people tried, very few succeeded.

Mac OS X 10.4: Enabling smart card login - Iceweasel

Eichier Édition Affichage Historique Marque-pages Outils Aide

http://support.apple.com/kb/TA24244?viewlocale=en\_US

Admin Shipping quotes Conditional actions Modules Recent log entries Available updates Traduction de l'interf... Translations Langues

Mac OS X 10.4: Enabling smart ...

Store Mac iPod iPhone iPad iTunes Support

## Mac OS X 10.4: Enabling smart card login

Last Modified: October 04, 2008 Old Article: 304035  
Article: TA24244

Products Affected  
Mac OS X 10.4

Mac OS X 10.4 Tiger greatly enhances integration for smart cards, as described in this advanced article. The configuration required is much simpler than in previous Mac OS X versions.

**Important:** The format of the configuration file and the scripts used to configure smart card login have changed from Mac OS X 10.3.x. Do **not** copy or use these files directly from a previous version of Mac OS X, or it may make the system unusable. It is no longer necessary to manually run pcsd, as this daemon is started automatically by Mac OS X when it detects a smart card reader.

It is safe to install a successfully modified /etc/authorization to enable smart card login on any client system, even those without smart card readers. If no reader or card is present, the user will continue to see the default login window, and there will be no performance impact.

To support login with a smart card on Mac OS X 10.4, the card must support signing with a public key. In addition, the card itself must have a login\_knows-as-a-token, that can communicate with securityd and the card itself.

Terminé

Mac OS X 10.4: Enabling smart card login - Iceweasel

Eichier Édition Affichage Historique Marque-pages Outils Aide

http://support.apple.com/kb/TA24244?viewlocale=en\_US

Admin Shipping quotes Conditional actions Modules Recent log entries Available updates Traduction de l'interf... Translations Langues

Mac OS X 10.4: Enabling smart ...

to support login with a smart card on Mac OS X 10.4, the card must support signing with a public key. In addition, the card itself must have a plugin, known as a tokend, that can communicate with securityd and the card itself.

### Tokens installed with Mac OS X 10.4

There are three tokens installed as part of the base Mac OS X installation.

Token	Smart card specification
"CAC"	US Federal Government: CAC: Common Access Card GSC-IS: Government Smart Card Interoperability Specification v2.1
"BELPIC"	Belgian Personal Identity Card
"JPKI"	Japanese Public Key Infrastructure card

Smart cards that are not compliant with the specifications listed above would require a supporting tokend be installed for the smart card to be recognized and available to applications.

### Enabling smart card login

Enabling smart card login is controlled through entries in the `/etc/authorization` file. Perform the following steps in Terminal to modify the authorization file.

1. Execute this command: `sudo -s`
2. Execute this command: `cd /etc`
3. Execute this command: `cp authorization authorization.orig`
4. Execute this command: `cp authorization /tmp/authorization.mod`  
Don't quit Terminal.
5. Next, edit the `/tmp/authorization.mod` file in pico, or your favorite text editor, or the plist editor that comes with the Apple Developer Kit.

Terminé

Mac OS X 10.4: Enabling smart card login - Iceweasel

Fichier Édition Affichage Historique Marque-pages Outils Aide

http://support.apple.com/kb/TA24244?viewlocale=en\_US

Admin Shipping quotes Conditional actions Modules Recent log entries Available updates Traduction de l'interf... Translations Langues

Mac OS X 10.4: Enabling smart ...

5. Next, edit the "/tmp/authorization.mod" file in pico, or your favorite text editor, or the plist editor that comes with the Apple Developer Kit.

6. Make the following changes to the "mechanisms" Array inside the "system.login.console" rights:

- After the string "<string>builtin:auto-login,privileged</string>" add the string "<string>builtin:smartcard-sniffer,privileged</string>"
- After the string "<string>builtin:reset-password,privileged</string>" remove the string "<string>authinternal</string>" then add string "<string>builtin:authenticate,privileged</string>"

7. Make the following changes to the "mechanisms" Array inside the "authenticate" rules:

- add the following string to the beginning of the array "<string>builtin:smartcard-sniffer,privileged</string>"
- after the string "<string>builtin:authenticate</string>" remove the string "<string>authinternal</string>" then add the string "<string>builtin:authenticate,privileged</string>"

8. After saving the changes, go back to Terminal and execute the following command (while still in the sudo -s session):

```
cp /tmp/authorization.mod /etc/authorization
```

This final step will replace the system's active authorization file with your edited version. These changes take place immediately so you do not need to restart.

Terminé

Mac OS X 10.4: Enabling smart card login - Iceweasel

Eichier Édition Affichage Historique Marque-pages Outils Aide

http://support.apple.com/kb/TA24244?viewlocale=en\_US

Admin Shipping quotes Conditional actions Modules Recent log entries Available updates Traduction de l'interf... Translations Langues

Mac OS X 10.4: Enabling smart ...

Developer Kit.

6. Make the following changes to the "mechanisms" Array inside the "system.login.console" rights:

- After the string "<string>builtin:auto-login,privileged</string>" add the string "<string>builtin:smartcard-sniffer,privileged</string>"
- After the string "<string>builtin:reset-password,privileged</string>" remove the string "<string>authinternal</string>" then add string "<string>builtin:authenticate,privileged</string>"

7. Make the following changes to the "mechanisms" Array inside the "authenticate" rules:

- add the following string to the beginning of the array "<string>builtin:smartcard-sniffer,privileged</string>"
- after the string "<string>builtin:authenticate</string>" remove the string "<string>authinternal</string>" then add the string "<string>builtin:authenticate,privileged</string>"

8. After saving the changes, go back to Terminal and execute the following command (while still in the sudo -s session):

```
cp /tmp/authorization.mod /etc/authorization
```

This final step will replace the system's active authorization file with your edited version. These changes take place immediately so you do not need to restart.

**Smart cards and Directory Services**

Part of the login process is to do a lookup for the expected user in a directory service such as Open Directory, LDAP, or Active

Terminé

Mac OS X 10.4: Enabling smart card login - Iceweasel

Eichier Édition Affichage Historique Marque-pages Outils Aide

http://support.apple.com/kb/TA24244?viewlocale=en\_US

Admin Shipping quotes Conditional actions Modules Recent log entries Available updates Traduction de l'interf... Translations Langues

Mac OS X 10.4: Enabling smart ...

Part of the login process is to do a lookup for the expected user in a directory service such as Open Directory, LDAP, or Active Directory. The first and recommended method to link a smart card user with a record in a directory service is to add the hash of the public key to the user's directory record. This is the most convenient and most secure way of identifying a smart card user. The second method is to lookup the user based on values drawn from the email signing certificate as required for the US Federal Government smart card use.

A script is preinstalled to assist you in binding a smart card to a user's local directory domain record. This is /usr/sbin/sc\_auth:

```
myhostname# /usr/sbin/sc_auth -h
Usage:  sc_auth accept [-v] [-u user] [-k keyname] # by key on inserted card(s)
        sc_auth accept [-v] [-u user] -h hash # by known pubkey hash
        sc_auth remove [-v] [-u user] # remove all public keys for this user
        sc_auth hash [-k keyname] # print hashes for keys on inserted card(s)
```

An example of the output from this for a US Department of Defense Common Access Card is:

```
myhostname% sc_auth hash
01C2F20D8964BE7701B57B63B0A1795B8F2604C1 Identity Private Key
443F30C356E676F447CD4DA89F46CC0CCE019737 Email Signing Private Key
4845564C1F8C6B378C19B8F262CE422933CF1FD1 Email Encryption Private Key
```

To add a user to the local directory

```
myhostname% sudo sc_auth accept -u myuser -h 01C2F20D8964BE7701B57B63B0A1795B8F2604C1
```

...where "01C2F20D8964BE7701B57B63B0A1795B8F2604C1" is the hash for the key associated with the Identity Private Key. Refer to the script for further usage instructions. You will need to run this as a user authorized to modify the directory. In this example, any of the hash entries listed could have been used for associating the card to the account. If desired, more than one smart card can be associated with a single user account by running the script again with the hash from the additional card(s).

Terminé

# Mac OS X

It takes 8 pages to reach the end.

# End of part 2

## Single sign-on:

- Windows: buy Windows \$erver .
- GNU/Linux: satisfactory, but not GUI.
- Mac OS X: impossible to set-up.

As a result ... « peut-mieux faire ».

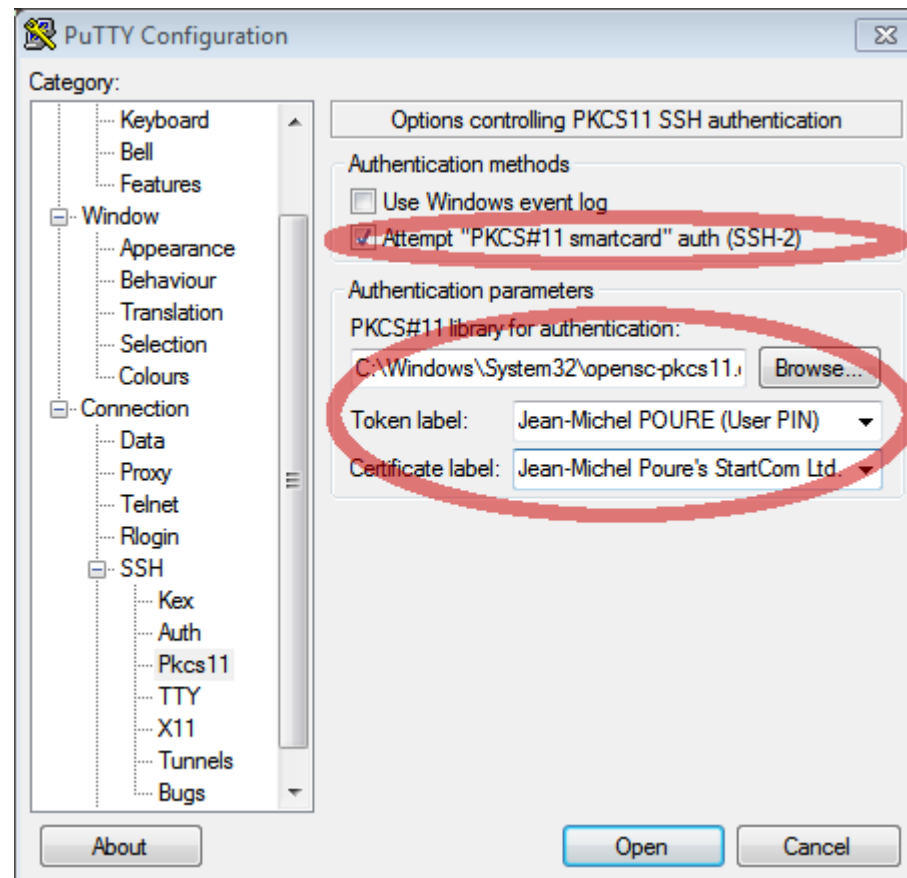
# Part 3: applications

The rule:

Smartcard features are  
Always hidden  
and/or difficult to set up.

Here are a few examples:

# Putty

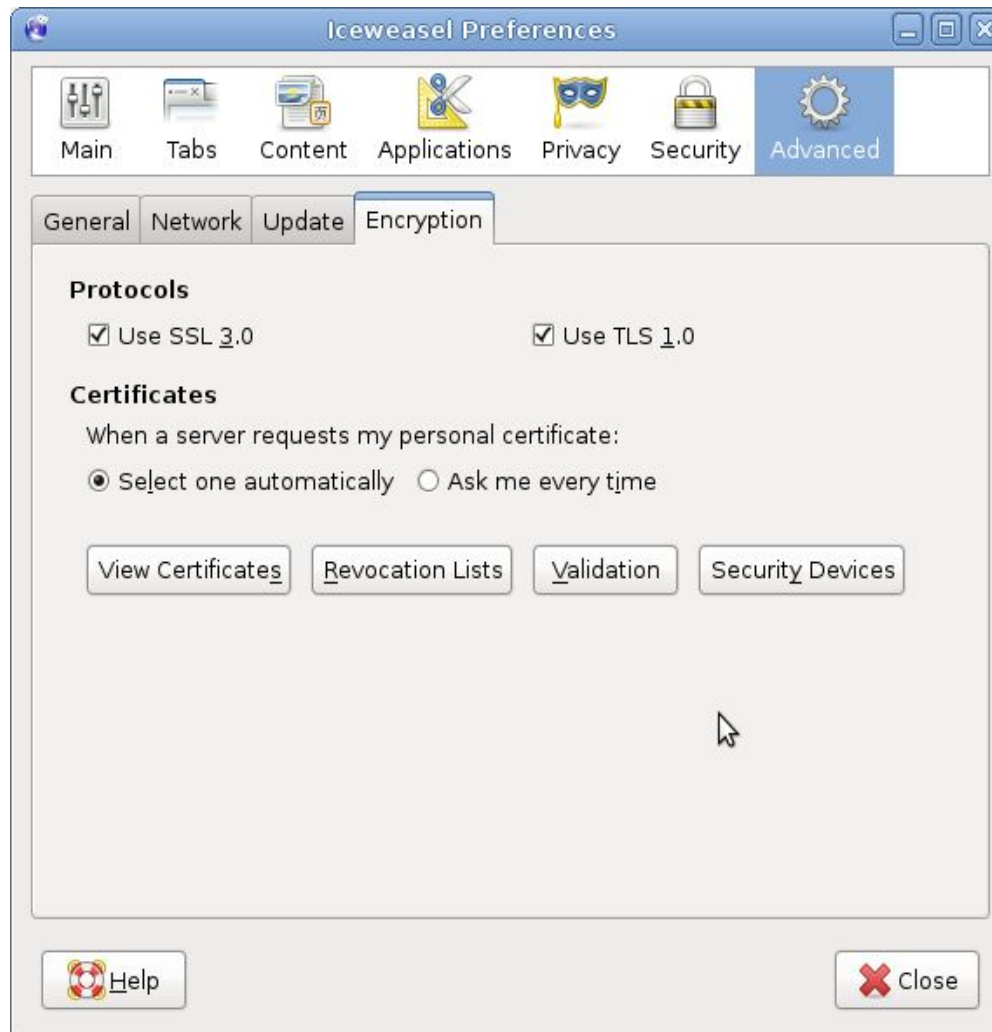


# OpenSSH client

- Adding the key:
  - `ssh-add -s /usr/lib/pkcs11.so` to add
- Remove the key:
  - `ssh-add -d /usr/lib/pkcs11.so` to remove

Patches available since 2006. Never implemented using OpenSC. OpenSSH finally used its own PKCS#11 library. Implementation not complete.

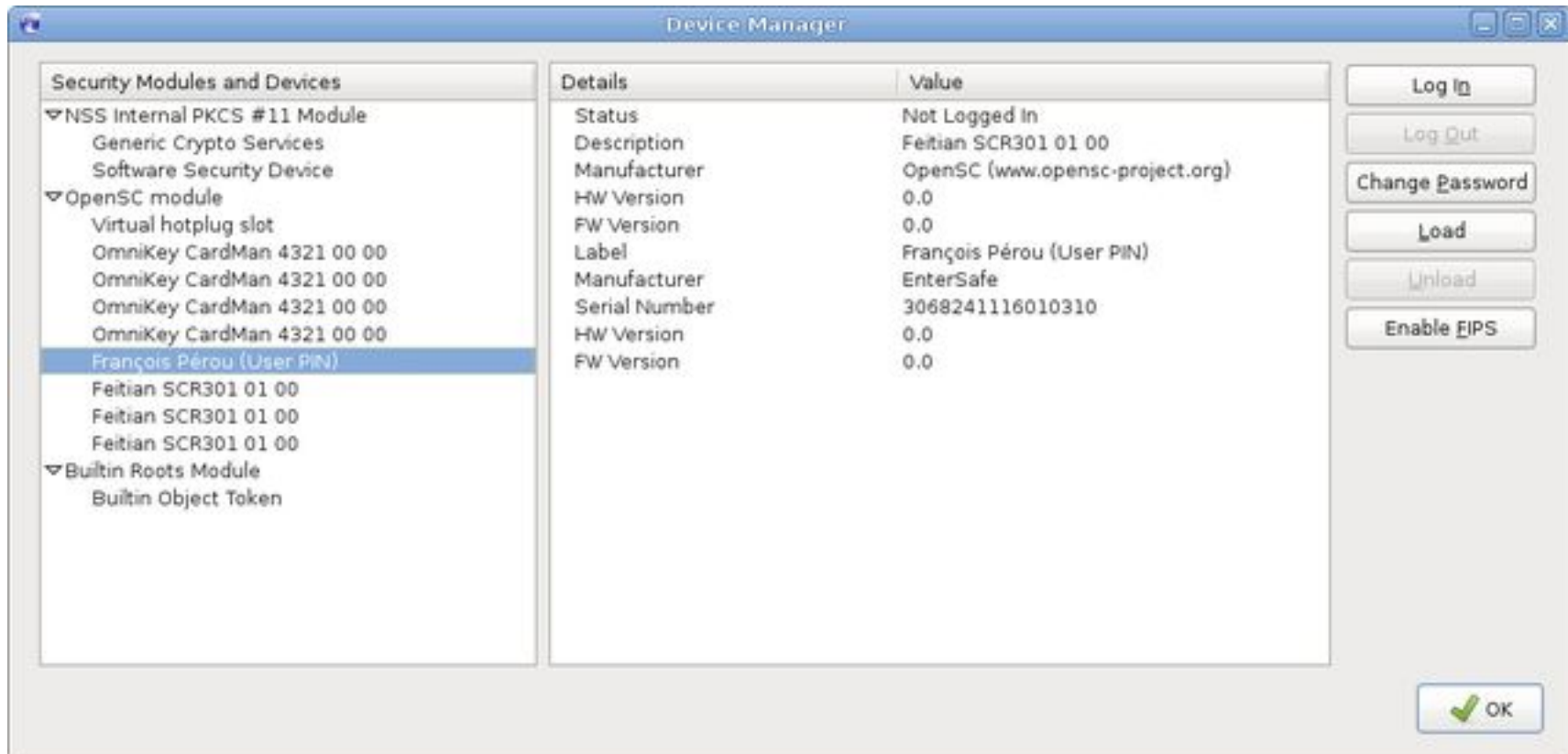
# Firefox / Icedoveasel



# Firefox / Icedweasel



# Firefox / Iceweasel



# Conclusion

- Muscle and OpenSC provide a complete solution. Soon available a new CSP driver for Windows. All this seems very exciting.
- Integration of frameworks in OS (logon) and applications is poor and can be enhanced.
- Credit cards are a success because you simply need to insert, enter PIN code and it works.
- Other conclusions will come during the day.

# Get a free smartcard

And start contributing to OpenSC.