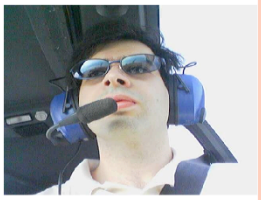
EXPLORING SMARTCARDS: AN INDEPENDENT LOOK TO TECHNOLOGIES AND MARKET

Giuseppe "Gippa" Paternò
gpaterno@gpaterno.com
June 2008

WHO AM I



- Experienced architect
  - Linux, Networking and Security
  - Focused on Telcos and Finance
- Special attitude on research projects and Deep market knowledge
- Currently employed in Red Hat as a Senior Architect and EMEA Security Specialist
- Previously Security Solution Architect in Sun and also in IBM
- Part of the italian security community *sikurezza.org*
- Teached wireless security on “master sulla sicurezza informatica”, Politecnico di Milano
- More on:
  - http://www.gpaterno.com/
  - http://www.linkedin.com/in/gpaterno

Overview

THE OBJECTIVE OF THE RESEARCH

- Explore the smartcard technology
- Understand market maturity
- Find a cross platform solution
  - Must work with Windows, Linux and Mac OS
- Make it working with common application
  - Mail, Web, VPN, Interactive login, …

WHAT IS A SMARTCARD

- "Smartcard" is an abused term
  - It's like saying "I use a computer", without specifying the type of architecture (x86/ppc/…), OS or the used application.
  - Smartcard is usually a card with an embedded chip
- Features of a smart card
  - Each smartcard has different OS/access method
  - Memory cards
    - to store/read information
  - Microprocessor card
    - to execute operations and/or simple applications
  - Can be contact or contactless

The use of the term "smart card" is quite general. It really means "something" to dumb, i.e. a card with a chip.
It's basically saying "I use a computer", without specifying which type of processor architecture (x86, ppc,..), nor OS or a given program.
There are two types of cards: memory cards and microprocessor cards.
The first is to read and store information, while the second one is quite complex and has a small microprocessor, with an operating system and an application loaded.

WHY SMARTCARDS?

- Keyloggers and ad-hoc worms are targeting identity thief
- Smartcards can protect from Man-in-the-middle attacks (password and content snooping)
- They provide tamper-proof storage
- It is based on the concept: "something you have" (the card) and "something you know" (the PIN)
- Need to be used on very sensitive environments
  - Where privacy and identity proof is a must
  - Ex: in government agencies, finance, …

We're facing some more targeted, smart attacks. Hackers are creating keyloggers and ad-hoc worms to steal someone's identity.
Smartcards can help solving identity thief, by providing a tamper-proof storage, as passwords and content cannot be sniffed on-the-wire.
It's based on the concept of "something you have and something you know", i.e. the card and the PIN. Of course can be implemented the "something you are", i.e. biometric authentication, but that's involve more sensitive environments.
Smartcards, in fact, should be used in that environments that privacy and identity proof is a must. For example, in government agencies and in the finance sector.

SMARTCARD USAGE

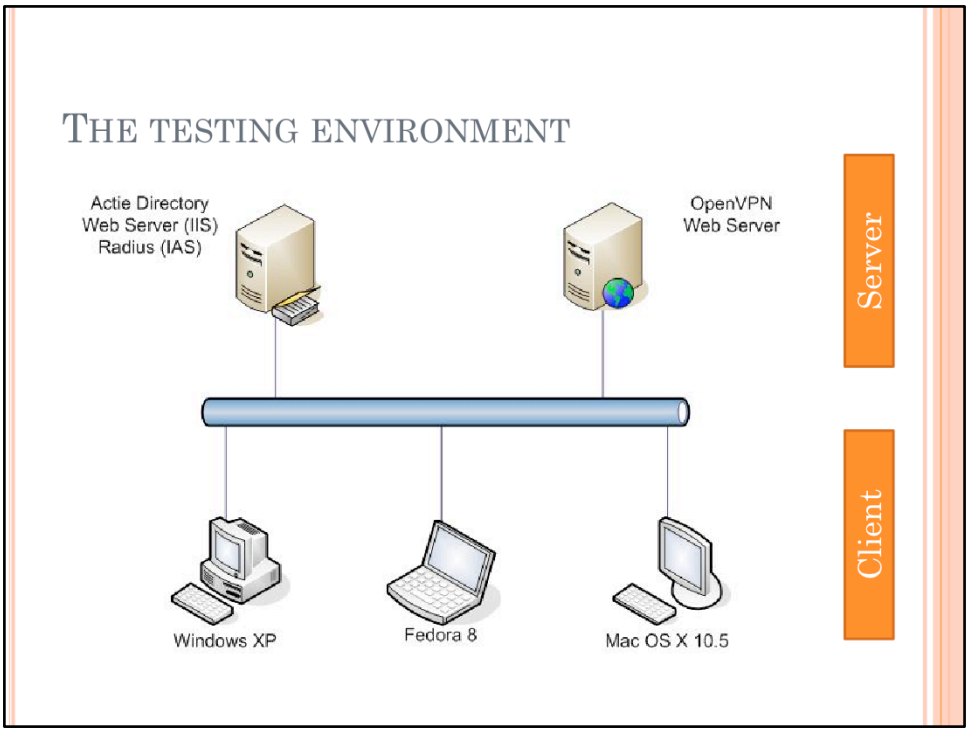
- As of today, very vertical use:
  - ATM cards
  - Loyalty cards
  - Access badges (proximity access cards)
  - Pay TV/Satellite TV cards (CAM)
- Not very much used as a certificate store for identity and encryption
  - Used in some government agencies such as US Dept. of Defense
  - Belgium introduced smartcard in the National ID

Smartcard usage is today very vertical.
We use several smartcards in today's life, for example ATM cards, loyalty cards in supermarkets, access badges, satellite's Conditional Access Modules (CAM). However, there's small usage as a certificate store for identity and encryption in the IT. There are some exceptions in some governments agencies such as US Dept. of Defense. A positive note is Belgium National ID, as every Belgium citizen has a smartcard with an X.509 certificate on-board.

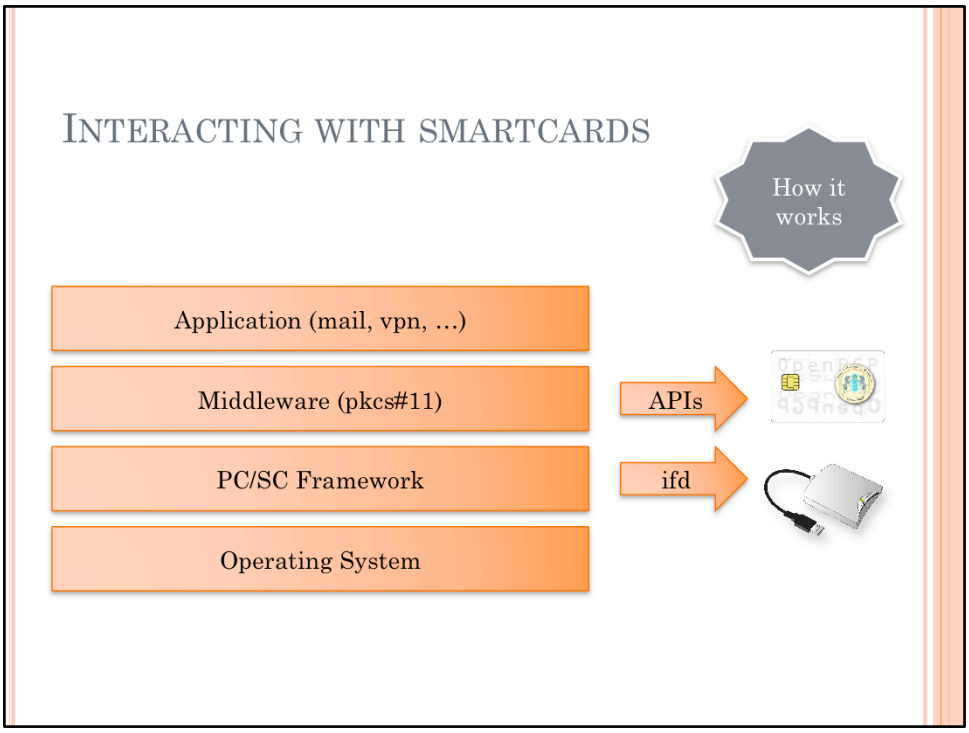
SMARTCARDS VS. OTP TOKENS

- OTP tokens are getting more popular
  - No need to install clients
  - Can be used in any browser and with every client
  - Provide good method to identify a person
  - Cannot be used for confidentiality (encryption)

We're facing a increase usage of OTP tokens in the IT industry, especially in current account self-service login for banks or VPN Access.
That's good as OTP tokens can increase security in the IT environment. There are some advantages using this methodology, for example there's no need to install any client software, and can be used anywhere and with any computer. However OTPs cannot be used for encrypting data, and this can be a limitation if such feature is needed.



The objective was to test extensively a customer environment. As such we setup an heterogeneous environment such as Windows Active Directory, ….



Talking to the smartcard IS NOT the same as talking to the reader
You have to make that both to work
PCSC is framework in the Operating System to provide a common access methodology

- Implemented into the os (pcsc-lite)
- Provide a common access to the reader through drivers (interface drivers or ifd)
- Gives a common APIs to talk to the card

The application will talk to the smartcard

- Each smartcard implementation has a library to access it
- This library is referred as "middleware"
- Since there's not a standard "application" on the card, there is no standard "middleware"

OS Support

OS SUPPORT: WINDOWS

- Microsoft implemented it via Cryptographic Service Providers (CSP)
  - CSP is proprietary
- PKCS#11 support is vendor-based
  - Each vendor provides its own pkcs11 library and a CSP plug-in
- Status of applications
  - All Microsoft applications uses CSP keystore
  - Most of the applications written using Microsoft tools uses CSP
  - Cross-platform applications uses pkcs11 (such as Mozilla, …)

OS Support: Mac OS

- Smartcard middleware implemented through "tokend"
- Integrated into keychain, that provides support for native applications
- Application support for pkcs#11 is vendor-based
  - Similar to Microsoft Windows
- Overall: good implementation

OS Support: Linux

- Linux smartcard support is painful
  - Not a single keystore
  - Might be better with Gnome 2.22 (smartcard support for the gnome-keyring)
- Each application has is own keystore
  - Difficult to maintain list of trusted CA
  - Difficult to maintain list of CRLs
- Not a common SSL engine
  - OpenSSL
  - NSS (from Netscape and Mozilla foundation)
  - Each distribution implements both in different way
- Smartcard support must be configured in each application!!

http://live.gnome.org/GnomeKeyring/Cryptoki

The cards

MOST ACCEPTED CARDS

- CAC: Common Access Cards
  - Issued by US DoD
  - Based on Java cards specifications
- BELPIC, the Belgium National ID
- JPKI, the Japanese PKI
- PIV: Personal Identity Verification
  - New standard coming out
  - For US federal employees or contractors
- Considerations
  - Good support across all the platforms
  - Public specifications
  - Public access to Root CA & CRLs
  - Cannot be used if your not in the above categories

OpenPGP

- Smartcard designed for PGP/GPG operation
- Seems the only open and cross platform working solution
- Supported natively from GPG version 2
- Public card specifications, anyone can produce cards
- Works great for mail and file encryption
- Is not usable for two factor authentication (VPN, Login, …)
  - A pkcs11 access is being developed

JAVA CARD: WHAT IS IT

- Smart card capable of running java applications (applets)
  - The term "formatting a card" means uploading an java applet into a card
- Can be any "java" application
  - Loyalty systems
  - Prepaid card
  - Access control badge
  - Cryptographic cards
  - …
- Several applets to manage PKI certificates
  - Most popular are Musclecard and Coolkeys

JAVA CARD: MUSCLE CARD

- Public specification
- Good community behind
- Difficult to "format" the card
  - Identity Alliance has a tool for Windows (ID Ally)
  - GPShell is a command line tool, but is complicated
  - Vendors might provide pre-loaded cards
- Operating Systems support:
  - ID Ally PKCS#11 and MS CSP support for Windows
  - Linux has libmusclepkcs11
  - No compiled packages for Mac OS
    - Can be compiled, but it's not easy

JAVA CARD: COOLKEY

- Developed by Red Hat
- Real cross platform
  - has support Windows, Linux, Mac OS
- Derived from CAC specifications
- Need a specific PKI infrastructure
  - Red Hat Certificate System
  - Fedora Project's Dogtag
- Is not possible to add certificates from other certificate systems (ex: Thawte, CA Cert)

OPEN SC

- Community effort to build a "swiss-knife" of middleware
- Support several cards and USB tokens
  - National IDs
  - Common smartcards
  - Common USB tokens
  - Complete list on: *http://www.opensc-project.org/opensc/*
- It has a great cross-platform support
  - Windows bundle available
  - Mac OS bundle available
    - No OpenCT support, therefore no USB tokens
  - Linux packages in most Distributions (Fedora, Ubuntu, ...)
- Not 100% compatible with cards
  - Usually created through reverse-engineering

ALLADIN ETOKEN

- Comes in two flavors:
  - USB token, easy to carry
  - ISO 7816 form-factor smart card
- Alladin's middleware is really cross platform
  - Works on Linux, Windows, Mac
- It's well documented
  - Good provided documentation
  - Lots of sites refer to it
- Very popular with lot of users, especially in the educational space
- Proprietary implementation:
  - Middleware client has to be paid extra
  - OpenSC implementation works good only on windows and Linux. Not working on Mac OS. Not compatible with Alladin's Middleware

PROPRIETARY CARDS/TOKENS

- Other proprietary solutions available
  - Mainly USB tokens
  - Good example: Eutronsec
- Usually works fine only on Windows
- Some "rare cases" for Linux and/or Mac OS

Conclusions

The smart card world is a mess!

CHALLENGES IN THE RESEARCH

- Lack of standards
  - Readers are quite standardized (CCID), yet some exceptions
  - No given standard for accessing the card
- Global Platform
  - An effort from manufacturers to bring a standard way to access PKI smartcards
  - Based on javacards (derived from the MuscleCard applet)
  - Still in the beginning phase
  - Web Site: http://www.globalplatform.org/
- Next generation, smartcards over mobile devices
  - Bluetooth readers for RIM/Blackberry

TECHNOLOGIES COMPARED

	OpenPGP	OpenSC	Alladin eToken	Muscle Card	Coolkey	CAC/PIV	BelPIC/JPKI
Mail	Yes	Yes	Yes	Yes	Yes	Yes	Yes
File encryption	Yes	Yes (2)	Yes (2)	Yes (2)	Yes (2)	Yes (2)	Yes (2)
VPN	No	Yes	Yes	Yes	Yes	Yes	Yes
Web	No	Yes	Yes	Yes	Yes	Yes	Yes
Interactive logon	No	Yes (3)	Yes (3)	Yes (3)	Yes (3)	Yes (3)	Yes (3)
Remote logon	Yes (1)	No (4)	No (4)	No (4)	No (4)	No (4)	No (4)
IEEE 802.1x	No	Yes	Yes	Yes	Yes	Yes	Yes
USB form factor	No	No	Yes	No	No	No	No
ISO 7816	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Can be multifunction	No	No	Only ISO	Yes	Yes	Yes (5)	No
Can load any cert	No	Yes	Yes	Yes	No	No	No
Need a specific CA	No, P2P	No	No	No	Yes (6)	Yes (7)	Yes (7)

Tests done with built-in capabilities, with supported software or with no-modification to OSS software

(1) Using the gpg-agent that emulates ssh keys
(2) A separate software is required for file encryption. The resulted encrypted file is not standard, but proprietary to the solution
(3) For windows, active directory is required to do an interactive logon using smartcards
(4) Remote logon via smarcard is somehow archievable by recompiling the openssh. RDC has it standard. It can be archieved using Kerberos and pkinit
(5) Born to be a multifunction card
(6) Can be used only with Red Hat Certificate System or Fedora's DogTag
(7) The cards are released by given CAs, such as DoD, or national Ids

If mail and file encryption are enough, go for OpenPGP
If you need also two factor authentication (VPN, Logon, Web, ...)
    if you're part of US DoD, or have a National ID with PKI, you're OK
    If you need a multifunction card (i.e. a badge)
        Musclecard, if you have few users
        Coolkey, if you can sustain a CA
Use Axalto eToken
    easy to carry
    really cross platform
    Stick to a proprietary stuff isn't that bad

USERS' BAD HABITS

- Most of the smartcard users leave the card in the reader
  - This is a security risk
  - It makes smartcard ineffective
  - Better use a software token instead
- Be a "smart" administrator, use tricks instead
  - Use a multifunction smartcard: the user must carry the card to open the doors
  - Use an USB token to carry in the keychain
  - Most of the time, enforcing a unique, good-strength password or an OTP token is sufficient

THE IDEAL WORLD

- Each nation should provide a multifunction smartcard
  - The best example is in Belgium
  - Other nations are doing the same (Finland, …)
  - In Italy: Regione Lombardia provided a smartcard with a certificate on-board, but drivers only for Windows (experimental drivers for OpenSC)
- A user should not have a key pair for each organization
  - Each organization is providing X.509 certificates
  - It should be a unique certificate to logon to systems and enter sites, independent of a given organization
  - Only the authorization should be demanded for each organization

GOING TO DEPLOY? SOME SUGGESTIONS…

- If you're small and mail and file encryption are enough: go for OpenPGP
- If you've a supported National ID or CaC/PIV
  - You're in the best position: avoid creating a CA
  - Configure your infrastructure to support these cards
- If you can sustain a Certification Authority (in terms of maintenance): go for the CoolKey
  - Java Cards are easy to find
  - Make sure it's a multifunction card, be aware of bad habits!
- Otherwise: use Axalto eToken
  - easy to carry
  - really cross platform
  - Stick to a proprietary stuff isn't that bad

THANK YOU!

Giuseppe Paternò
gpaterno@gpaterno.com
Web Site: http://www.gpaterno.com/