gemalto
security to be free

How can webapps benefit from automotive environment, with safety?

Web and automotive
W3C workshop

Pierre.Girard@gemalto.com

Rome, November 14, 2012

Agenda

- Gemalto introduction
- Car as a programming platform
- Safety, security and privacy requirements
- Recommendations

Gemalto at a glance

Customers

50 Government programs & customers worldwide

490 telecoms with services for 2.5 billion subscribers

300 financial institutions serving more than 500 million cardholders

Employees

10,000 employees

90 nationalities

40 countries

Shareholders

↑ 2B € Revenue

↑ PFO up by 15% at 239M €

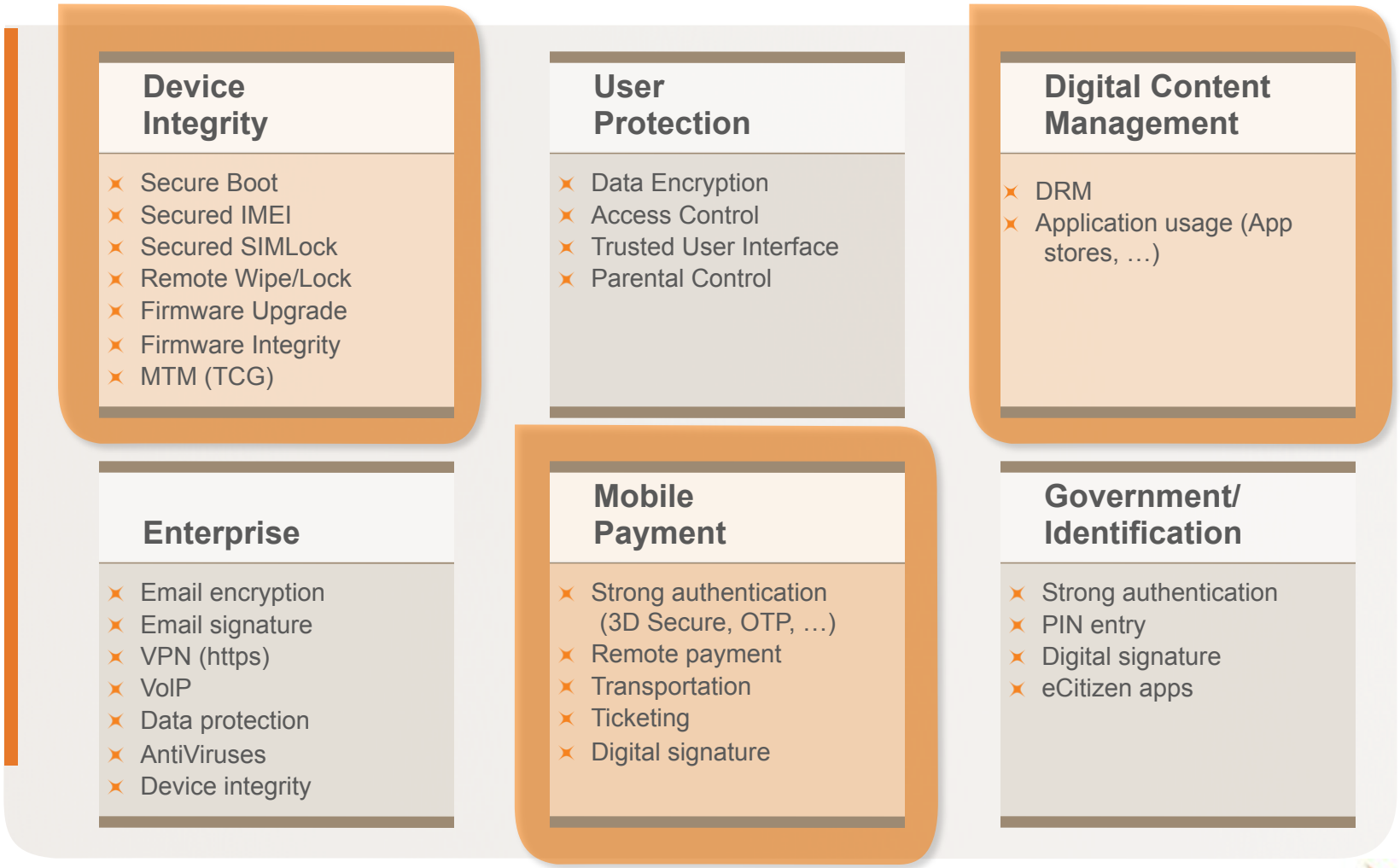
Society

Eco friendly design & manufacturing practices

Developing local markets

Sponsored community service projects

The need for digital security and trust is booming…



… and it has to come with ***convenience***

Machine to Machine Communications

How our M2M solutions are making a difference

Mobile health

We allow patients to be treated at home and alert healthcare providers if necessary

Smart energy

We help power smart grids, balance loads, reduce home energy consumption & speedily charge electric vehicles

Track & trace

We ensure goods can always be located by their owners, logistics companies but not the bad guys

Automotive

We have announced our partnership with Deutsche Telekom & BMW for eCall wide-scale deployment

Our customers

PHILIPS RESPIRONICS

BOSCH Invented for life

TZ Medical Sparked by your ideas

Aerotel Medical Systems

T

RWE The energy to lead

Landis+Gyr manage energy better

Trimble

BMW

MASSTIME

VeriFone THE WAY TO PAY

Hardware factorization in cars

Navigation

Speed radar locator

Ecodriving

Multimedia

Car as a programming platform

- Services are provided as apps
- The car needs to provide a rich API in order to be an attractive platform for developers
  - Case study: RelayRides app on OnStar

- Can we avoid the native app fragmentation problem ?

How to protect …

- Safety
  - How to prevent access to CAN bus by malicious in-car apps ?
  - How to prevent malicious firmware upgrade ?
- Privacy
  - How to selectively disclose location, driving patterns, …
  - Big Data or local aggregation and inference ?
  - Anonymous authentication and payment
- Security
  - How to prevent car stealing by hacking ?
  - How to prevent mileage modification ?
  - How to prevent Denial Of Service ?

Which threat model ?

- The car use cases and lifecycle is more complex than a electronic appliance
- Who would be the attacker ?
  - Driver(s), passengers, owner, car dealer, maintenance operator, thieves, remote hacker
- Both remote and physical attacks will be faced
- The car life cycle need to be considered
  - Wiping personal data when reselling the car, locking when in maintenance …
- Various use cases
  - Renting, sharing, company fleet

Software security

- Protected environment
- Trusted users
- Direct access to data

Hardware security

- Unprotected environment
- Non trusted users
- No direct access to data
- **Tamper resistant devices**

What about cars ?

A security framework will be needed

- Of course we need permissions on API
  - But it's not so simple
  - Avoid the "Click I accept" syndrome
- Permissions need to be managed based on
  - Service provider / developer identity
  - Certification status
  - User authentication
  - Car life cycle state (e.g. in maintenance)
  - Real time context (e.g. speed)
- Apps and services will also need
  - Users and car authentication
  - Billing framework

Identification and authentication

- Management of identities and roles
  - Roles = owner, driver, passenger, shift manager, fleet manager, maintainer, …
- Flexible authentication methods
  - Biometrics
  - Cryptography
  - Hardware based
- Flexible security levels
  - Not the same level needed for kids screen skinning and door opening
- Various form factors
  - USB tokens, SD cards, mobile phone, key fob, driving license, ….

App life cycle management

- Actors
  - Developer
  - Service provider
  - Car platform manager
  - Evaluation and certification entity
- App life cycle
  - Development
  - Evaluation and certification
  - Loading and installation
  - Usage
  - Upgrade
  - Uninstall

Recommendations

- Technical
  - Standardize a powerful and attractive car API
  - Design a safety / security / privacy model
    - Permission based
    - Role based
    - With a flexible authentication framework
- Method
  - Encourage automotive industry and service providers to participate
  - Connect with other W3C workgroups (sysapp, deviceAPI)
  - Reuse from existing specifications (e.g. OMTP Bondi)
  - Connect with other organizations (Genivi, OneM2M …)

Thank you !