



Smart Cards: A Bridge Between Users And Trusted Platforms

Patrick George
Gemplus

Overall Presentation Goal

- Propose a “smart card-and-TPM” cooperative model for Trusted Architecture
 - Introduce to TCG user authentication model
 - Demonstrate how smart cards can enhance the existing mechanisms
 - Show increasingly secure implementations of TCG authentication protocols with smart cards
 - Open up to other user-related issues that can be addressed by smart cards in the context of TCG

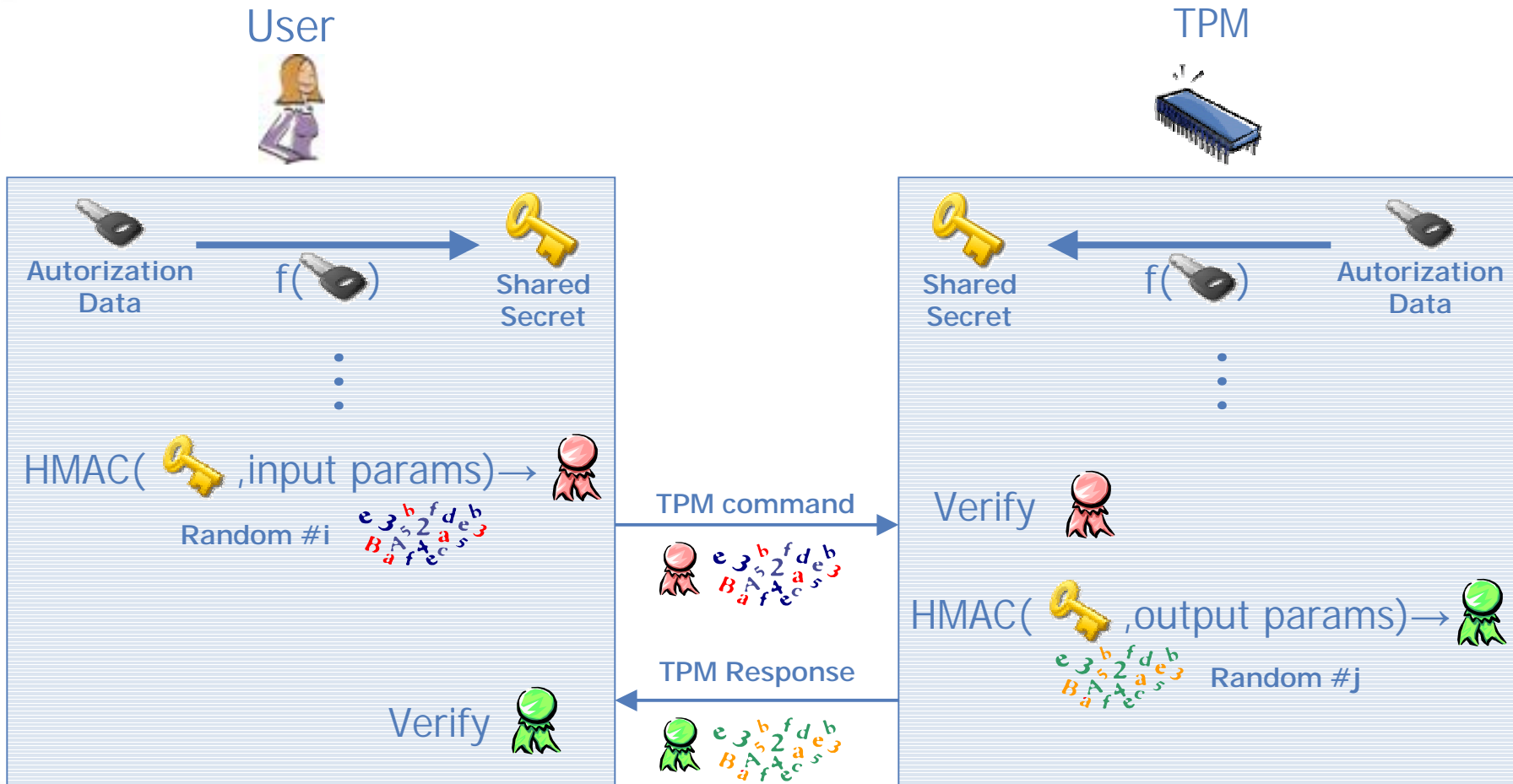
User Authentication in TCG

- TCG defines ways to “authenticate an owner and to authorize use of a TPM object” using TPM command validation mechanisms
- Simply rely on a secret data shared between the TPM and authorized users
- TCG refers equally to authentication and authorization when these mechanisms are discussed

Authorization Data And Protocols

- Authorization Data
 - 20-byte value shared between the TPM and authorized users
 - Knowledge of Authorization Data is complete proof of ownership of a TPM-protected object
- Authorization protocols
 - Provide proof of knowledge of the Authorization Data
 - Handle the confidential creation of the Authorization Data
 - Allow the secure update of Authorization Data

TCG Authentication Protocols



Authentication values protect TPM commands

Threats

- Authorization Data is the cornerstone of TCG authentication model
- A key issue is Authorization Data storage protection
 - Must not be accessed by unauthorized entities
 - Must not be duplicated
- Vulnerable to well-known attacks if stored on the platform itself
 - Static attacks: attack the container
 - Dynamic attacks: during the processing

Implementation Threats

- It is impossible for a user to remember a 20-byte random value
- Most of the products implement a password-based mechanism
 - Authorization Data=SHA-1(password)
- Password well-known weaknesses apply
 - One-factor only
 - Vulnerable to dictionary attacks
 - Can be snooped when keyed in or transmitted
 - Easy to lose and forget

Smart Cards Role

- The first role of smart cards will be to secure the card holder Authorization Data storage
- Countermeasures provided by smart cards
 - Two-factor authentication
 - Tamper-resistant storage
 - Cannot be cloned
 - Isolation of security-critical computations involving the Authorization Data from other parts of the system

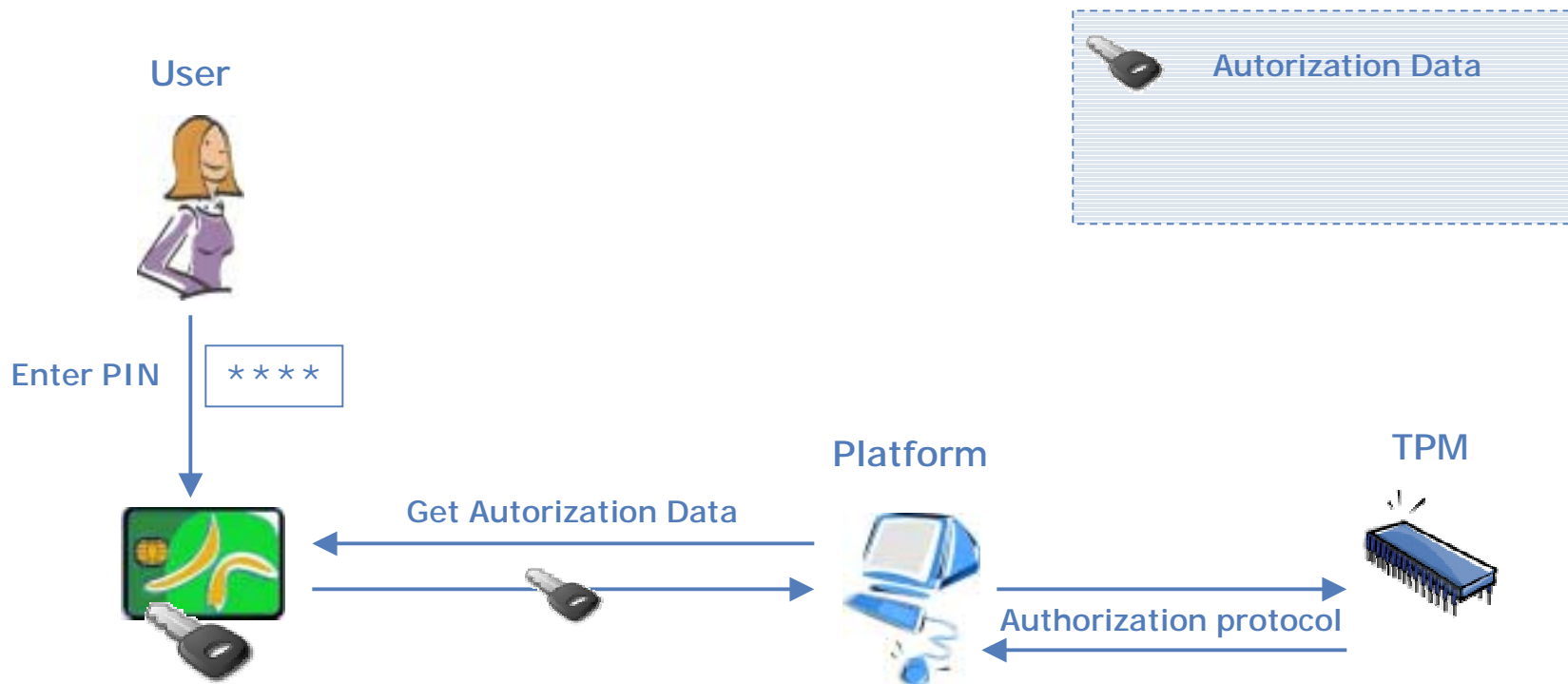
Smart Cards Capabilities

- Storage
 - Store 20-byte values
- Processing
 - Generate random values (RNG)
 - Generate Authorization Data (SHA-1)
 - Compute a shared secret according to TCG specifications (HMAC)
 - Compute and verify authentication values (HMAC)
 - Secure new Authorization Data transmission (XOR-enc)

Possible Solutions

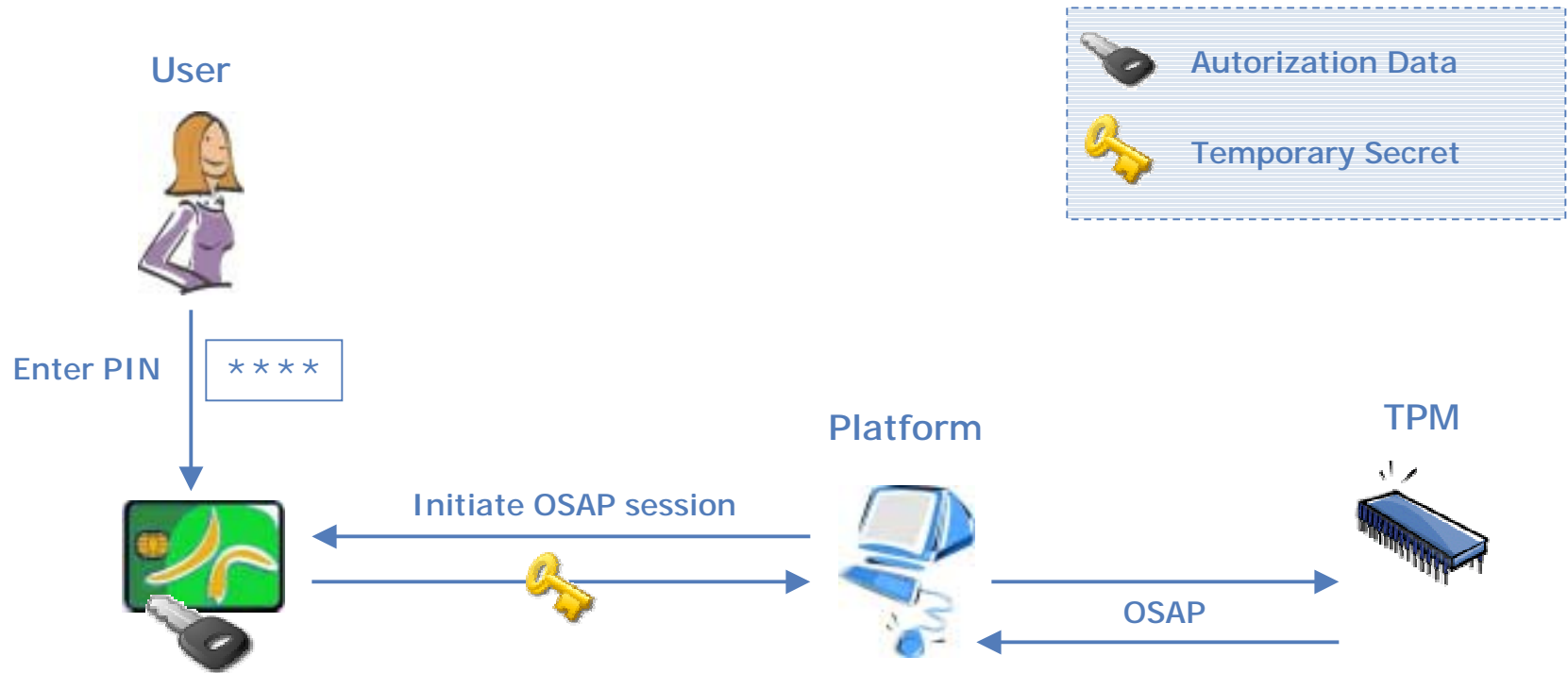
- Several approaches are possible, depending on
 - Security level required
 - Smart card capabilities
- Increasingly secure implementations
 1. Simple storage
 2. Compute OSAP shared secret
 3. Compute the authentication values generation and verification
 4. Encrypt the new Authorization Data (ADIP)

1- Storage



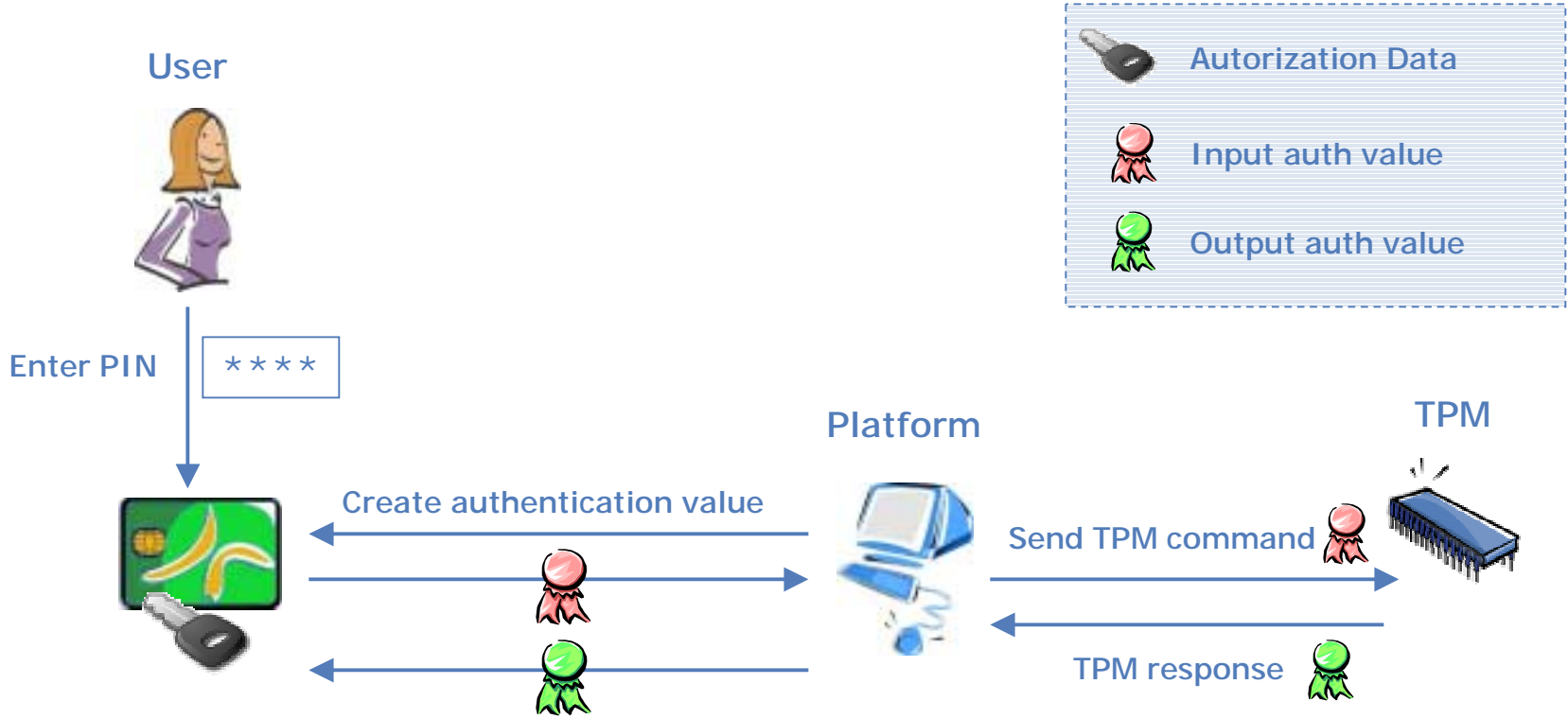
Very simple, but Authorization Data is exposed

2- Shared Secret (OSAP)



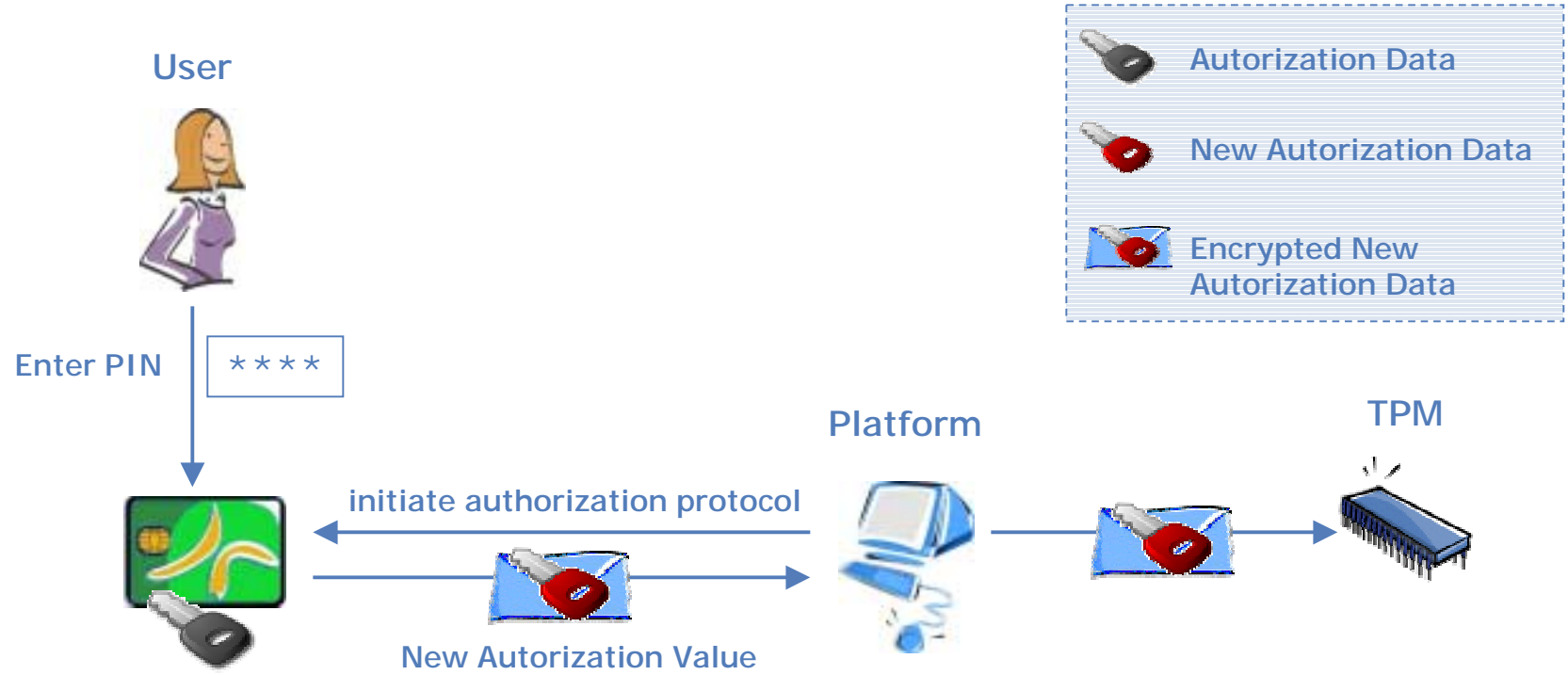
Per session unique shared secret

3- Authentication Values



Authorization Data are never exposed

4- ADIP



Secure generation and insertion of Authorization Data

Beyond User Authentication

- Does one security device fit all?
 - Same device for platform and user secrets?
- Separate credentials
 - User credential portability
 - User administration simplification
 - Protection level adequacy
 - User privacy
- Leverage from corporate deployments
 - Logical access to computers
 - Physical access control badges too
- Towards a smartcard-and-TPM cooperative model

Conclusion

- Smart card build more secure implementation of TCG authentication protocols
- Beyond this usage, smart card can help tackle platform user related issues in TCG
 - User mobility
 - Privacy
- A cooperative model combining smart card and TPM can bring Trusted Computing Architecture
 - More security
 - User benefits
 - Easier deployment and management