Secure Token Container

Gerald Madlmayr

Hagenberg R&D Competence Center

NFC FORUM

Introduction

- **NFC Research Project in Austria**

fh OBERÖSTERREICH   mobilkom austria   PHILIPS   voestalpine   SKIDATA access unlimited

- **Different Areas of Interests**
  - Hardware Development
  - Use cases and Software/Application Development
  - Usability
  - Security
- **Infrastructure for NFC Trials on University Campus**

Goal

- **Standardize Ticketing/Access/SmartCard Solution for NFC Devices**
  - Definition of a Public Key Infrastructure & Participants
  - Secure Protocol whole token flow
  - Different card type should be abstracted in the JCOP Container (Software)
- **Integration of NFC Token Software into the Operating System of mobile Devices**

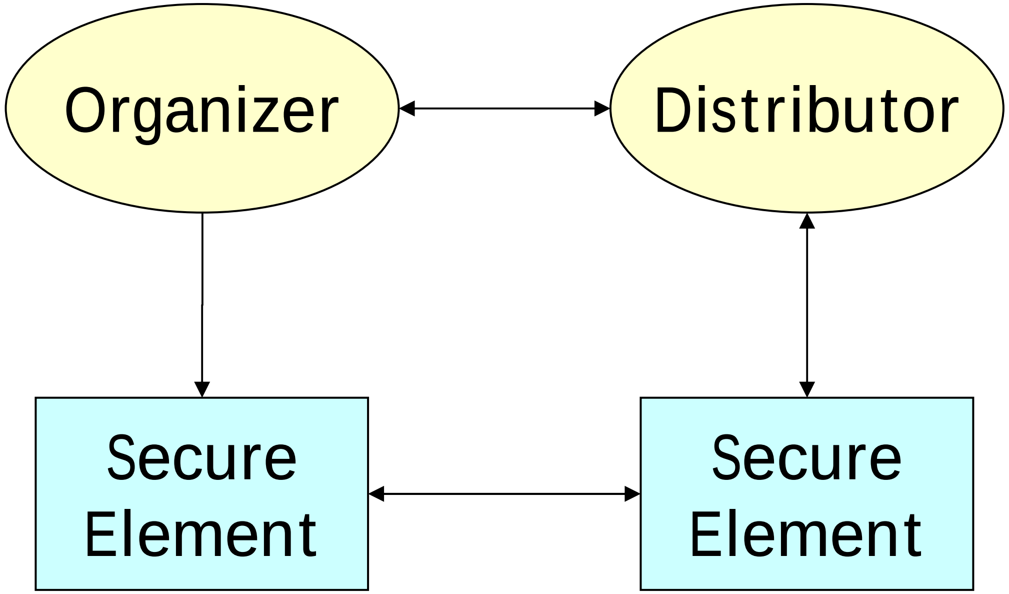
Requirements

- Target Platform: Mobile Phone
- Tokens Delivery via SMS/WAP Push (~ MMS)
- Secure Communication from Token Issuer to Access Gate
- Multiple Token/Token-Applications possible within one device
- Good Performance for En/Decryption
- No User interaction required at Gate or when Token is received

Idea

- Usage of Java Card Applet as Token Container
- Dynamic allocation of Token (not limited to Memory-Blocks in the view of size & amount)
- Secure-Element holder responsible for Applications on Java Card

Token Flow

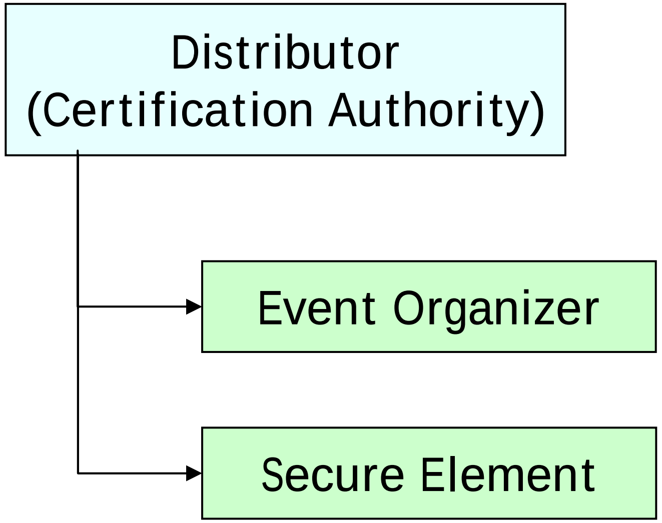


mobilkom austria

SKIDATA
access unlimited

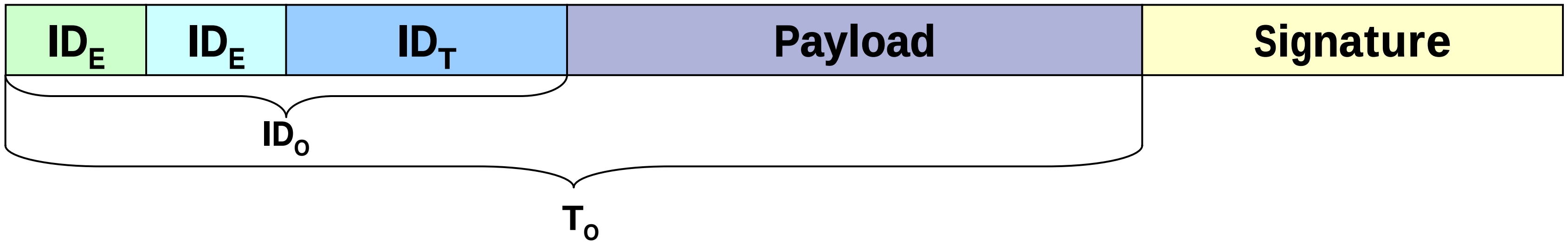
PKI Participants

- *Token Distributor*: **Self Signed Authority (CA), signs public keys of other instances with its private key.**
- *Secure Element*: **Contains signed public Key; needed for secure Communication to Reader and Token Distributor**
- *Event Organizer*: **holds signed ID for each token/event; also needed for authentication to mobile phone. Signed public key needed for authentication against token distributor**



Token

- Transferred via binary SMS/MMS



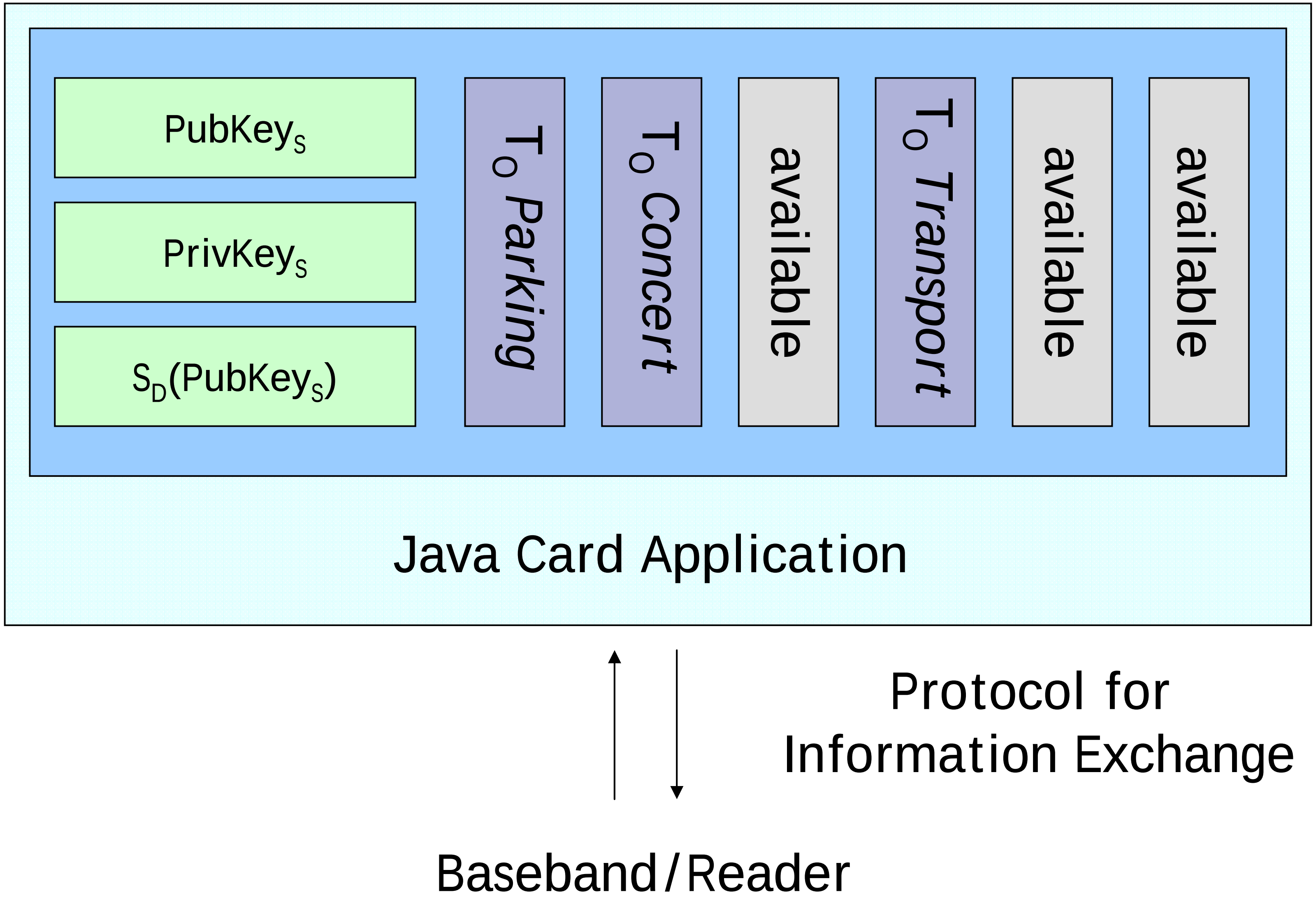
ID_E ID_E ID_T Payload Signature

ID_O

T_O

- ID_E for Event (split between Organizer and Distributor)
- ID_T for Token (set by Organizer)
- Combination of ID_E and ID_T necessary for valid token.
- Payload (Counter, Name, Period of Validity, ...)
- Signature

Secure Element - Token Container



$PubKey_S$

$PrivKey_S$

$S_D(PubKey_S)$

T_O *Parking*

T_O *Concert*

available

T_O *Transport*

available

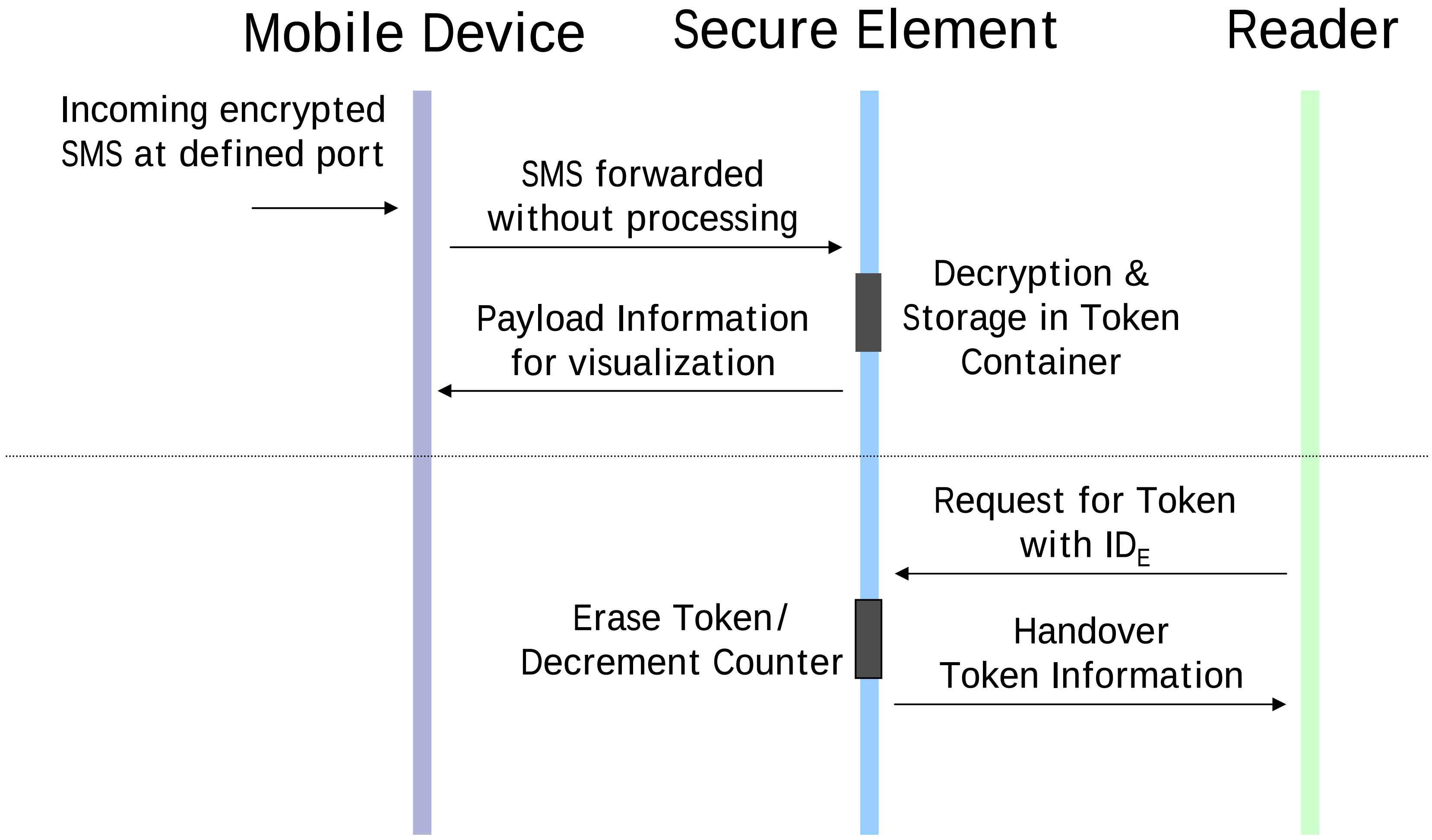
available

Java Card Application

Protocol for Information Exchange

Baseband/Reader

Prototype



Mobile Device

Secure Element

Reader

Incoming encrypted SMS at defined port

SMS forwarded without processing

Decryption & Storage in Token Container

Payload Information for visualization

Request for Token with ID_E

Erase Token / Decrement Counter

Handover Token Information

Protocol

- **Key Exchange & Installation of Application**
  - Certification of Event Organizer
  - Certification of mobile Device
- **Preparation of Token**
  - Request for Event ID
  - Initialization of Reader
- **Booking of Token**
  - Transfer of Token from Organizer to Distributor
  - Delivery of Token to mobile Device
- **Verification of Token**

Conclusion

- **For Secure Token System a trusted Instance (“Distributor”) needs to be introduced.**
- **Owner of Secure Element needs to be defined.**
- **Token Container, Security Protocol and Token Format need to be defined.**
- **NFC Devices should be delivered with pre installed Token Management Software.**