# PA193 - Secure coding principles and practices

## Designing good and secure API
## Automata-based programming

Petr Švenda svenda@fi.muni.cz

CROCS
Centre for Research on Cryptography and Security

# Security module – informal description

*"Hardware token (module) allows to import two master symmetric cryptography keys and set initial user PIN value in the trusted environment during the personalization. Once token is personalized and issued to a user, a user can submit its own data, that are then encrypted (or decrypted) and integrity protected (or integrity verified, MAC). This operations are available only after successful verification of the user PIN. User can change his/her PIN to new value after successful verification of the current PIN. Cryptographic algorithms used are set and fixed during personalization, but should be easily modifiable for the newer tokens, if required."*

# Design security API, make FSM model

- Design security API for security module adhering to the best practices described during the lecture
  - gather requirements, make use cases, create API, document...
  - provide declaration of API in C/C++
  - add SAL annotations
- Create FSM states model with transitions
  - use Graphviz's .dot format
  - visualize, add to documentation

# Homework

- Deadline: 26.11.2015 23:59 (Groups 1-3), 1.12.(G4)
- Finish API design and FSM
- Submit:
  - use cases (plaintext or UML)
  - API written in source code (C/C++)
  - JavaDoc-like documentation inside source code
  - SAL annotations inside source code
  - generate html from documentation (Doxygen)
  - include FSM visualization (Graphviz)
  - (Pseudo)code for client using API