PV204 Security technologies

Labs: Project presentation, improvements

Petr Švenda svenda@fi.muni.cz
Faculty of Informatics, Masaryk University

CS
Centre for Research on Cryptography and Security

The plan

- Presentation of projects (Phase II.), discussion
  - 10-15 minutes presentation
- Collaborative work on google docs spreadsheet summarizing important observations
  - https://docs.google.com/spreadsheets/d/1VNj1WnrcaSotiO2aNT7lLGrwlCP1fbS7FqqTSay-cB0/edit?usp=sharing
  - Fill your findings into spreadsheet under your project
  - Use colour for issue seriousness
- Proposal how to fix detected issues
  - Plan for fixing, discuss (also with me)
  - Open issues on GitHub under corresponding milestone

HOMEWORK, PROJECT

Homework

- Homework: Password-Authenticated Key Exchange (PAKE)
  - 29.3.2018 24:00
  - Two options: simpler (DH-based) and advanced (RSA-based)
- Project
  - Phase III.: Improve code (GitHub), 10 points (19.4.2018)
  - Functionality and security tests (unit tests, integration)
  - Best practices, fix identified problems
  - Version + repo info specific command
  - Verify on simulator and on real card
  - Based on the performed analysis and discussion
    - Identified in Phase II and discussed today

Diffie-Hellman Encrypted Key Exchange

Step	Alice	Bob
1	Shared Secret: $S = H(password)$	
2	Parameters: p, g	
3	$A = \text{random}()$ $a = g^A \pmod p$	$\text{random}() = B$ $g^B \pmod p = b$
4a	$E_S(a) \longrightarrow$ $\longleftarrow E_S(b)$	
4b	$a \longrightarrow$ $\longleftarrow E_S(b)$	
4c	$E_S(a) \longrightarrow$ $\longleftarrow b$	
5	$K = g^{BA} \pmod p = b^A \pmod p$	$a^B \pmod p = g^{AB} \pmod p = K$
6	$\longleftarrow E_K(data) \longrightarrow$	

http://www.themccallums.org/nathaniel/2014/10/27/authenticated-key-exchange-with-speke-or-dh-eke/

Homework: PAKE

- Create implementation of Password-Authenticated Key Exchange
  - Shares authentication using PIN or short password (max. 6 characters)
    - Shared between card and PC (can be hardcoded)
  - Derive properly two AES128 session keys
  - Implement also PC-side code and demonstrate its functionality
    - Start with simulator, then attempt real card
  - Explain why attacker can't perform offline bruteforce after eavesdropping APDU-level communication
  - Remove all unnecessary code (no leftover from examples!)
- Produce short text description of your solution
- Option 1: DH or ECDH-based PAKE (max. 5 points)
- Option 2: RSA-based PAKE (max. 7 points)
- Submit **before 29.3. 23:59** into IS HW vault
  - Soft deadline: -1.5 points for every started 24 hours

Option 1: DH-based PAKE

- Use DH or ECDH for ephemeral keys
- Select suitable version
  - Make clear which one you selected
  - Make clear why you selected that one
- Relatively straightforward (maximum 5 points)

Option 2: RSA-based PAKE

- Use RSA for ephemeral keys
- Study existing RSA-based protocols
- Important: completely secure RSA-based PAKE is not easy straightforward task
- You can still get full number of points, even when your scheme will not be 100% secure against all attacks
  - But you must clearly describe the limitations of your design and implementation
- More demanding (maximum 7 points)