Design and Evaluation of PEARS
Privacy Enhancing ARchitectures

Antonio Kung
Outline

- PEARs: Privacy Enhancing Architectures
- Using ATAM for Privacy-by-design
  - Architecture Trade-off Analysis Method
- Using CBAM for Privacy-by-design
  - Cost Benefit Analysis Method
- Conclusion
Possible Definitions of Architecture

- High level structure of a system
- Documentation of this high level structure
  - e.g. ISO/IEC/IEEE 42010 Systems and software engineering — Architecture description (replaces IEEE 1471)
- Discipline of creating such a high level structure
PEAR: Privacy Enhancing ARchitecture

- Discipline of creating a high level structure integrating privacy

- Includes the following phases
  - Architecture analysis
  - Architecture design
  - Architecture evaluation
Architecture Tactics Examples

Send Message → pseudonymization → Pseudonym Based message → Anonymity Quality measure

Data collected At vehicle level for billing → Zero-knowledge Based proof → Data not Collected at Information System level → Minimization Quality measure
Tactics for Architecture Design

Tactics for Privacy Quality Attribute

Minimization tactics

- Anonymize credentials (e.g. Direct anonymous attestation)
- Limit processing perimeter (e.g. client processing, P2P processing)

Enforcement tactics

- Enforce data protection policies (collection, access and usage, collection, retention)
- Protect processing (e.g. storage, communication, execution, resources)

Accountability tactics

- Log data transaction
- Log modifications (policies, crypto, protection)
- Protect log data

Modifiability tactics

- Change Policy
- Change Crypto Strength and method
- Change Protection Strength

Event that creates potential Privacy loophole

System Resists, Traces or Recovers
Examples of Solutions (Patterns)

- **user data confinement pattern**
  - Minimisation

- **hippocratic management**
  - Enforcement

- **Isolation**
  - Enforcement
Architecture tactics can change functional requirements

- Functional Requirements
- Architecture Analysis
- Architecture Design
- Architecture Evaluation

Examples:
- e.g. Birth date
- e.g. Data minimisation
- e.g. Proof > 21 year old
ATAM

- Architecture Tradeoff Analysis Method
  - Relies on **quality models**
  - Relies on **utility trees**
  - Relies on **attribute driven design method**
Quality Models

- Functions to predict the response measure

- Two approaches
  - **Analytic models (support quantitative analysis)**
    - Availability: markov models/statistical models.
    - Performance: queuing theory/scheduling theory
  - **Check lists/Guidelines**
    - Security: e.g. common criteria, TVRA
    - Safety: e.g. safety integrity level
Utility Trees

- Level 1: Quality attributes
  - e.g. minimisation
- Level 2: Attribute refinements
  - e.g. amount of data revealed
- Level 3: Definition of Architecturally Significant Requirements
  - Scenario
    - system can provide a proof that user is above 21 instead of transmitting birthdate.
  - Business value (high, medium, low)
  - Architecture impact (high, medium, low)
Attribute Driven Design Method

- Iteration-based
  - Choose an element of the system to design
  - Identify ASRs for that part
  - Generate a design solution (architectural views)
  - Inventory of remaining requirements and selection of input for the next iterations
Example of architectural views

- Module views

- Allocation views
ATAM Output

- Concise presentation of architecture
  - Views (e.g. static, dynamic, deployment)
  - Quality attributes and Tactics (e.g. minimization and attribute based credential)
- Business goals (e.g. financial, market position, legal, competitive, quality, ...)
- Utility trees
- Risks (e.g. privacy leaks)
- Sensitivity points
  - architectural decisions affecting some quality attribute measure (e.g. data better protected)
- Tradeoff points
  - architectural decisions affecting several quality attributes measures, some positively and some negatively (e.g. data better protected but response time is not as good)
ATAM

- Participants
  - Evaluation team
  - Project decision makers
  - Architecture stakeholders
    - Do not participate to entire exercise

- Phases
  - Evaluation phase 1
    - Involves project decision makers and evaluation team
  - Evaluation phase 2
    - + Architect stakeholders
ATAM Steps

- **Phase 1**
  - Step 1 - Present ATAM
  - Step 2 - Present Business Drivers
  - Step 3 - Present Architecture
  - Step 4 - Identify Architectural Approaches (list patterns and tactics)
  - Step 5 - Generate Quality Attribute Utility Tree
  - Step 6 - Analyze Architectural Approaches
    - Focus on higher ranked scenarios

- **Phase 2**
  - Step 7 - Brainstorm and Prioritize Scenarios
  - Step 8 - Analyze Architectural Approaches
  - Step 9 – Present results
### Scenario

**Analysis of Architectural Approach**

| Scenario #: 1 | **In-house non authorised access** |
| Attribute(s): Protection enforcement |
| Environment: **SMS data base of mobile phone operator** |
| Stimulus: **Attempt to access SMS log of a celebrity** |
| Response: **Access denial** |

<table>
<thead>
<tr>
<th>Architectural Decisions</th>
<th>Sensitivity</th>
<th>Tradeoff</th>
<th>Risk</th>
<th>Nonrisk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access control</td>
<td>S1</td>
<td>T1</td>
<td>R1</td>
<td>N1</td>
</tr>
</tbody>
</table>

**Reasoning:** Non authorised accesses are detected.

<table>
<thead>
<tr>
<th>Sensitivity Points</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td><strong>SMS log better protected</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tradeoff Points</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td><strong>Affects flexibility of access and sometimes customer services</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Risks</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td><strong>Access control may be attacked</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Non-risks</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td><strong>SMS data base encryption means slower access</strong></td>
</tr>
</tbody>
</table>
CBAM (Cost Benefit Analysis Method)

- Takes place after ATAM
- Maximize difference between
  - benefit derived from system
  - and cost of implementing the design
CBAM: Utility-Response Curves

- For each quality create a utility-response curve
- Example anonymity quality:
  - Case a
    - Response: K-anonymity 1
    - Utility 0
  - Case b
    - Response: K-anonymity 3
    - Utility 50
  - Case c
    - Response: K-anonymity 6
    - Utility 90
CBAM: Some Metrics

- Overall Benefit of an Architectural Strategy
  - $B_i$: benefit of architectural Strategy $i$
  - Strategy $i$ described through $j$ scenarios
  - $W_j$: weight of scenario $j$
  - $b_{ij}$: change in utility caused by scenario $j$: $U_{\text{expected}} - U_{\text{current}}$
  - $B_i = \sum_j (b_{ij} \times W_j)$

- Value for cost (VFC)
  - $C_i$: cost of implementing architecture Strategy $i$
  - $VFC = B_i / C_i$
CBAM Steps

- Step 1: collate scenario
  - Choose the top third
- Step 2: refine scenario
  - Worst case, current, desired, best case QA response level for each scenario
- Step 3: prioritise scenarios
- Step 4: assign utility for step 3 scenarios
  - Worst case, Current, Desired, Best Case
- Step 5: identify architectural strategies and associated scenarios. Determine their expected QA response level
- Step 6: Determine the utility of the expected QA response levels by interpolation
- Step 7: Calculate total benefit obtained from an architectural strategy
- Step 8: Select architectural strategy base on VFC (compatible with cost and schedule constraints)
- Step 9: confirm results with intuition
For more information, visit the PRIPARE website: http://www.pripareproject.eu

Thank you for your attention

QUESTIONS?

Coordinator
Antonio Kung (Trialog)

Technical Coordinator:
Christophe Jouvray (Trialog)