Security Evaluation of Service-oriented Systems with an Extensible Knowledge Base

Manuel Rudolph
manuel.rudolph@iese.fraunhofer.de

Authors:
Christian Jung (IESE)
Manuel Rudolph (IESE)
Dr. Reinhard Schwarz (IESE)
AGENDA

- Motivation
- Project Overview
- SiSOA Method
- Exemplary Security Evaluation
- Future Work
- Conclusion
Motivation

- Security problems can be distributed all over the system, especially in SOA

- Aggregate security related information on a more abstract level
  - e.g. on architectural level ➔ big picture
Motivation

What you expect

What you may have
Project Overview

- SiSOA project
  - security evaluation of SOA applications on architectural level

- Goals
  - displaying static security features and issues on architectural level
  - gather and reuse security knowledge
  - tool supported security analysis

- Non-goals
  - automatic security assessment: detailed manual inspection indispensable
The SiSOA-Method
Overview

Extraction Phase

System under Evaluation

Identification Phase

System Model

Enriched System Model

Security Analysis Phase

Architectural Views with Security Tags

Knowledge Base

Security Expert

Security Analyst

Fraunhofer IESE
The SiSOA-Method
Extraction Phase

- **Artifact:** Element that can be aggregated to a security feature or issue
- Parse source code and configuration files
- Put system structure into system model (SAVE tool)
- Fill system model with artifacts:
  - imports
  - call relations
  - exceptions
  - annotations
  - …
The SiSOA-Method Knowledge Base

- **Security Tag**: Marker for security features or issues
- Tree-like structure
- Security Tags / Security Tag Rules
  - Aggregate artifacts
  - Boolean operators
  - Weightings
- Security Goals and Indicators
  - Indicate security features/issues
  - Categorize security tag rules
  - Help to find adequate tag rules in the knowledge base
The SiSOA-Method
Identification Phase

Extraction Phase ➔ Identification Phase ➔ Security Analysis Phase

Knowledge Base

System Model

Security Tag Rule
The SiSOA-Method
Identification Phase

- Mapping of artifacts and security tag rules
- Enriching the system model with tags

- Use of metrics to calculate
  - Credibility of tags
    - Probability that a security tag is placed correctly
  - Severity of tags
  - Quality of the security tag rule
The SiSOA-Method

Analysis Phase

- Views with assigned tags can be generated:
  - Class view
  - Service view

- Analysis features:
  - Adding or changing tags
    - In the model
    - In the knowledge base
  - Rerun identification (after adding or changing tags)
  - Different filters
The SiSOA-Method
Analysis Phase

- Class view:
  - classes
  - packages
  - relations
  - security tags
The SiSOA-Method
Analysis Phase

- Service view:
  - services
  - service interfaces
  - trust zones
  - security tags
Exemplary Security Evaluation

- Toy Example
  - Restaurant SOA application
- SOA Framework SCA (Service Component Architecture)
- Features:
  - order food
  - make reservations
  - pay bill with credit card
  - …
Exemplary Security Evaluation

- Problem: Information Disclosure
  - Services that process credit card information should not share them

- Several service scopes in SCA
  - **Stateless**: does not store data between two requests
  - Stateful: does store data between two requests
    - **Composite**: one instance for all users
    - **Conversation**: one instance per user

- We need stateless or conversation scopes
Exemplary Security Evaluation

```java
package credit.lib;

import credit.api.CreditCheckService;
import org.osoa.sca.annotations.ConversationAttributes;
import org.osoa.sca.annotations.ConversationID;
import org.osoa.sca.annotations.Scope;
import org.osoa.sca.annotations.Service;

@Service(CreditCheckService.class)
@Scope("CONVERSATION")
@ConversationAttributes(maxAge="30 minutes",maxIdleTime="20 minutes")
public class CreditCheckServiceImpl implements CreditCheckService {
    private int creditCardNumber;
    private int securityCode;

    private String ConversationID;

    public boolean checkCreditCard(int creditCardNumber, int securityCode) {
        this.creditCardNumber = creditCardNumber;
        this.securityCode = securityCode;
        if (creditCards.containsKey(creditCardNumber) &&
            creditCards.get(creditCardNumber) == securityCode)
            return true;
        else
            return false;
    }
}
```
Exemplary Security Evaluation

- Conversation Scope has been found in CreditCheckServiceImpl.java

  Stateless Scope SCA - Severity: (10, 1), Credibility: 0.4
  Conversation Scope SCA - Severity: (9, 1), Credibility: 1.0
Future Work

- Filling the knowledge base with more security related rules
- Gain practical experience with the SiSOA method
- Evaluation, especially of
  - the methodology itself
  - the used metrics and weightings
  - the gathered security knowledge in the Knowledge Base
- Adapt the prototype to more SOA technologies
Conclusion

- SiSOA method can aggregate security features/issues on architectural level
- Method supports experts during security analysis

- Method is as good as the provided information in the Knowledge Base
- Knowledge Base can never be exhaustive

- Method is also applicable on non-SOA applications
Thank you for your attention
Any questions ???

→ Manuel Rudolph

Fraunhofer IESE, Germany
Information Systems Quality Assurance (ISQ)
+49 631 6800 2289
manuel.rudolph@iese.fraunhofer.de