安全・安心を設計する

SSE Project: 安全・安心なソフトウェアを構築するための研究プロジェクト

吉岡信和、総研大学、早稲田大学、電通大学、東工大学、信州大学、立命館大学、JAIST、
富士通研究所、みずほ情報総研、オープン大学、フロリダアトランティック大学、イーストロンドン大学ほか

なにができる？

安全・安心なソフトウェアの構築をサポートするための
ソフトウェア工学技術を確立し、その普及を目指します。
具体的には下記を開発します。
- 体系的な方法論の確立
- モデリングをサポートするツールの開発
- セキュリティソフトウェア工学の教育教材開発

どんな研究？

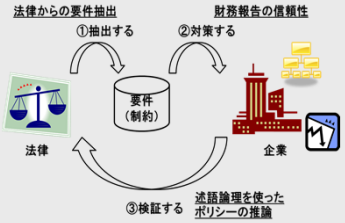
近年、個人情報流出や不正アクセスの危険性など、情報システムのセキュリティは社会問題となってきています。その中で、アドホックなセキュリティパッチや運用強化などの対応は限界があります。本プロジェクトでは、システムの**要求時から運用まで一貫したセキュリティ**を考慮した開発を実現します。

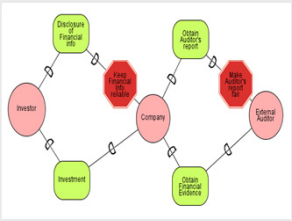
プロジェクトの目的

安全・安心なシステムを構築するための
セキュリティ・セーフティのモデル化・インテグレーション技術の開発

要求技術：法律を順守するためのフレームワークの研究

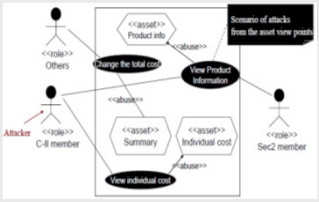
＜背景＞：個人情報保護法、内部統制、国際会計基準などの
新しい法律や基準への迅速な対応が必要.
＜方法＞：法律から要件を抽出、要件に沿った対策を設計、
妥当性を検証. 要件は、ゴール指向言語でモデル化.

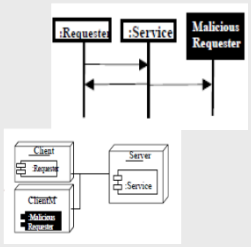




設計技術：セキュリティパターンの研究

- アタックのモデリング・パターン化
- パターンの形式化
- パターンの摘出・検索

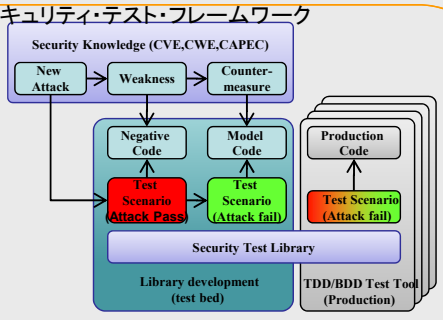




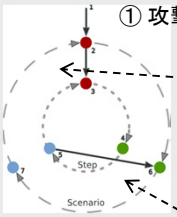
実装技術：テスト駆動開発におけるセキュリティ対策の研究

＜目的＞：セキュリティ対策のコスト低減と品質向上
＜課題＞：セキュリティテストの難しさ
様々な脅威と脆弱性への対応が必要
テスト対象となる環境そのものが変化
定量化手法の欠如
＜解決＞：テスト駆動開発に適した、新しいセキュリティ・テストの実現
–テスト駆動設計とセキュリティ・テストを統合
–モデル化による対象アプリケーションの振る舞いの網羅的な把握
–セキュリティ・テストの自動生成と、カバレッジの自動確認
–セキュリティ・テストのライブラリ化によるテスト記述の簡素化

セキュリティ・テスト・フレームワーク



テスト駆動設計によるセキュリティ対策の実施例



① 攻撃を想定

② ③ 攻撃をテストケースで表現（自動生成）

例）CSRF対策

```
When /^(?:|I )press "([^¥"]*)" with CSRF$/ do |button|
begin
    set_hidden_field 'authenticity_token', :to => "577adb5c-b8c5-11df-a45c-
080027fe0165"
    @result = click_button(button)
  rescue => @exception
  end
End
Then /^(?:|I )should have CSRF error$/ do
  @exception.should be_a_kind_of(ActionController::InvalidAuthenticityToken)
end
```

④ ⑥ テスト実行 ⇒ 攻撃に対するセキュリティ対策を実装

```
Scenario: Add task description, emulate CSRF(CWE-352)        #
features/task_description.feature:41
 Given I am on the new task page                              #
features/step_definitions/web_steps.rb:14
 When I fill in all required fields                           #
```

⑤ ⑦ リファクタリング