Don’t Repeat Yourself: Automatically Synthesizing Client-side Validation Code for Web Applications

Nazari Skrupsky, Maliheh Monshizadeh, Prithvi Bisht, Timothy Hinrichs, V.N. Venkatakrishnan, Lenore Zuck

UIC University of Illinois at Chicago
Department of Computer Science
Overview

- Introduction
- Goals, Challenges
- Our Approach
- WAVES Tool
- Results
- Conclusion
Introduction

- Web Application Development
  - Client-side
    - HTML, JavaScript, ...
  - Server-side
    - PHP, Java, ASP

⇒ Independent development is problematic
  • When the client and server share application logic
Input Validation

Browser

Web Application

Credit Card Information
Card Number: 1234
Expiry Date: Jul 21 1996

SUBMIT

Reject invalid inputs
Parameter Tampering

- Input validation must always occur at the server
WAVES
(Web Application Validation Extraction and Synthesis)

- Automatic synthesis of input validation for client-side

- Benefits:
  - Development Efficiency
  - Greater Compatibility
  - Code Efficiency
Automatic Synthesis

Challenges

• Inference of server-side constraints
  • Server-side: Variables
  • Client-side: Form Fields

• Preservation of application logic and security

• Validation involving the server state
WAVES Architecture

- Non-interactive Web Application
- Server Analysis
- Constraints
- Code Synthesis
- Integration
- Interactive Web Application

Success Input
WAVES

1- Server Analysis

\[ F_{\text{server}} = \text{All conditions on user inputs that must be satisfied to reach sensitive operations} \]

1. Submit benign inputs
2. Extract server formula
1- Server Analysis

Non-interactive Web Application

Instrumented Server

Execution Trace

Taint Analysis

Constraints: $C_i$

String Solver

inputs satisfying $\neg C_i$

Taint Analysis

Instrumented Server

Constraints representing Error Conditions
WAVES

2- Synthesis

• Static Constraints
  • pass1 \(==\) pass2

• Dynamic Constraints: **Dependent** on the server state
  • userID is UNIQUE
WA VES

2- Synthesis

WAVES - Code Synthesis

Constraints representing Error Conditions

Server Stub Generator

Dependency Analysis

Program Slicing

Client-side Code Generator

Dynamic Constraints

Static Constraints

Server code

Server Stub

JavaScript code
Results

- Three medium to large and popular PHP applications
  - B2Evolution
  - WeBid
  - WebSubRev

- Successfully synthesized 83% of the constraints

- Generated Stubs are much smaller (less than 26%)

- Improved RTT 43 to 164 ms (originally 65 to 633 ms)
Conclusion

- Code efficiency
- Interactive applications
- Improved performance
Related Papers


Questions?