

언어를 이용하여 정보 흐름 파악하기

박 창 희

카이스트 프로그래밍 언어 연구실

2011년 1월 7일

Previous Work

Presentation

- ▶ “Adding Pattern Matching to Existing Object-Oriented Languages”, joint work with Sukyoung Ryu and Guy L. Steele Jr.


Achievement

- ▶ Presentation at FOOL'10 workshop, Reno, Nevada, USA, 17. Oct. 2010


Introduction

- ▶ "Web is tomorrow's operating system" by Andrei Sabelfeld
- ▶ JavaScript - the most dominant language in web applications
- ▶ But still vulnerable to security problems such as confidential information leakage

Motivating Example

[Return to eBay.com](#)[Return to eBay.ca](#)

New to eBay?
[start here](#)



Freight Resource Center

Your solution for moving heavy items.

Powered by
FREIGHTQUOTE.COM

Choose A Topic

- [Home](#)
- [Add a Freight Calculator](#)
- [Rate & Schedule](#)
- [Trace Shipments](#)
- [My Account](#)
- [FAQ](#)


Helpful Links



- [View Demo](#)
- [Packaging Tips](#)
- [About freightquote.com](#)
- [Glossary & Definitions](#)

Payment information

Please provide payment information to confirm your shipment.

☐ Apply charges to my Freightquote.com account.

☐ PayPal 


☐ I would like to pay by credit card.  

Card name:

Card number:

Expiration date:

Name on card:

Pay for shipment 

```
<!-- Input validation -->
<form name="cform" action="script.cgi"
method="post" onsubmit="return
checkform();">

<script type="text/javascript">
function checkform () {...}
</script>
```

From "Language-based security" by Andrei Sabelfeld

Former Approaches

Non-language based approaches

- ▶ Dynamic monitoring : covers only single execution path

Language based approach

- ▶ Security-typed language : no runtime overhead but inapplicable to dynamically-typed languages such as JavaScript

Interesting Approach

Language based dynamic analysis

- ▶ "Efficient Purely-Dynamic Information Flow Analysis" by Thomas H. Austin and Cormac Flanagan, 2009
- ▶ "Permissive Dynamic Information Flow Analysis" by Thomas H. Austin and Cormac Flanagan, 2010
- ▶ Introduces a small dynamically-typed language with security labels
- ▶ Proves that the language guarantees non-interference property

Future Directions

- ▶ Improving expressiveness of the language
- ▶ Minimizing performance overhead
- ▶ Considering various channels such as timing channels