

Toward Hybrid Enforcement of Security Policies in JavaScript Applications

Scull Pupo, Angel Luis; Nicolay, Jens; Gonzalez Boix, Elisa

Publication date:
2018

License:
GNU GPL

Document Version:
Final published version

[Link to publication](#)

Citation for published version (APA):
Scull Pupo, A. L., Nicolay, J., & Gonzalez Boix, E. (2018). *Toward Hybrid Enforcement of Security Policies in JavaScript Applications*. Poster session presented at 15th International Conference on Managed Languages & Runtimes, Linz, Austria.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Context



- **Static** policy verification
- **Dynamic** policy enforcement
- **Selective** source code instrumentation

Objectives

- **Hybrid** program verification
- **Reduce** the runtime overhead introduced by monitoring the application

Implementation & Results

Input Program

```
G.onCall(document.write).deny();
document.write("foo"); //Disallow
document.createElement("div"); //Allow
```

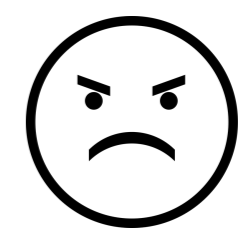
Instrumenter

Monitor

Full Instrumentation

```
ADVICE.invoke(ADVICE.invoke(G, "onCall",
[ADVICE.get(document, "write", 12)], 10), "deny", [], 9);
ADVICE.invoke(document, "write", ["foo"], 15);
ADVICE.invoke(document, "createElement", ["div"], 19);
```

- Redundant instrumented code
- Redundant policy verifications
- Increased parse and load application time



Selective Instrumentation

```
G.onCall(document.write).deny();
ADVICE.invoke(document, "write", ["foo"], 10);
document.createElement("div");
```

- Less redundant policy checks
- Less instrumented code

