Gatekeeper

Mostly Static Enforcement of
Security & Reliability Policies for JavaScript Code

Salvatore Guarnieri, University of Washington

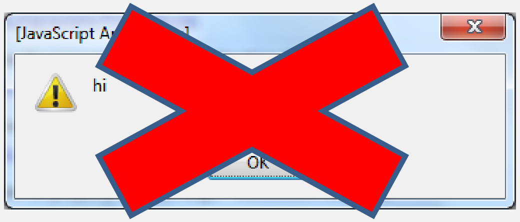
Ben Livshits, Microsoft Research

Catch me if you can

```
alert('hi');
```

program

malicious



[JavaScript A]

hi

don't want to allow alert box

?

can we figure this out statically?

```
alert(‘hi’);
```

```
document.write(
“<script>alert(‘hi’);</script>”);
```

```
var d = document;
var w = d.write;
w(“<script>alert(‘hi’);”);
```

```
eval(“do”+”cu”+”ment.write(”+…
```

```
var e = window.eval;
e(“do”+”cu”+”ment.write(”…”);
var e = new Function(“eval”);
e.call(
    “do”+”cu”+”ment.write(”…”);
```

```
var e = new
  Function(unescape(“%65%76%61%6C”));
  e.call(“do”+”cu”+”ment.write(”…”);
```

Gatekeeper

Static analysis for JavaScript

- General technology we developed for JavaScript
- Can use for performance optimizations, etc.

This paper

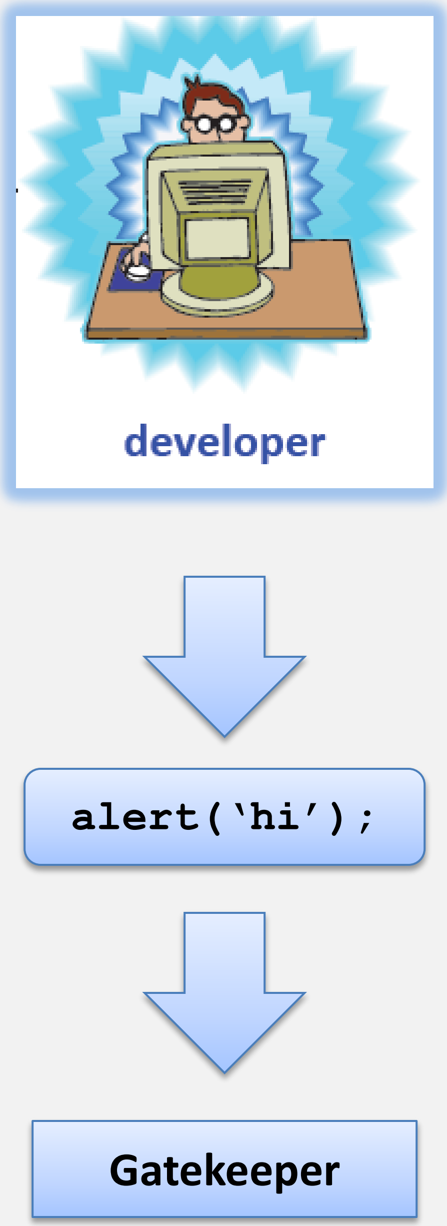
- Use to enforce security and reliability policies
- Analyze Web widgets

Focus on *whole program analysis*. Contrast with:

- JavaScript language subsets (do a little of)
- JavaScript code rewriting (do a little of)

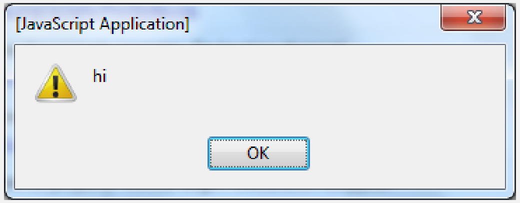
Goal of Gatekeeper:

Reason about JavaScript code statically



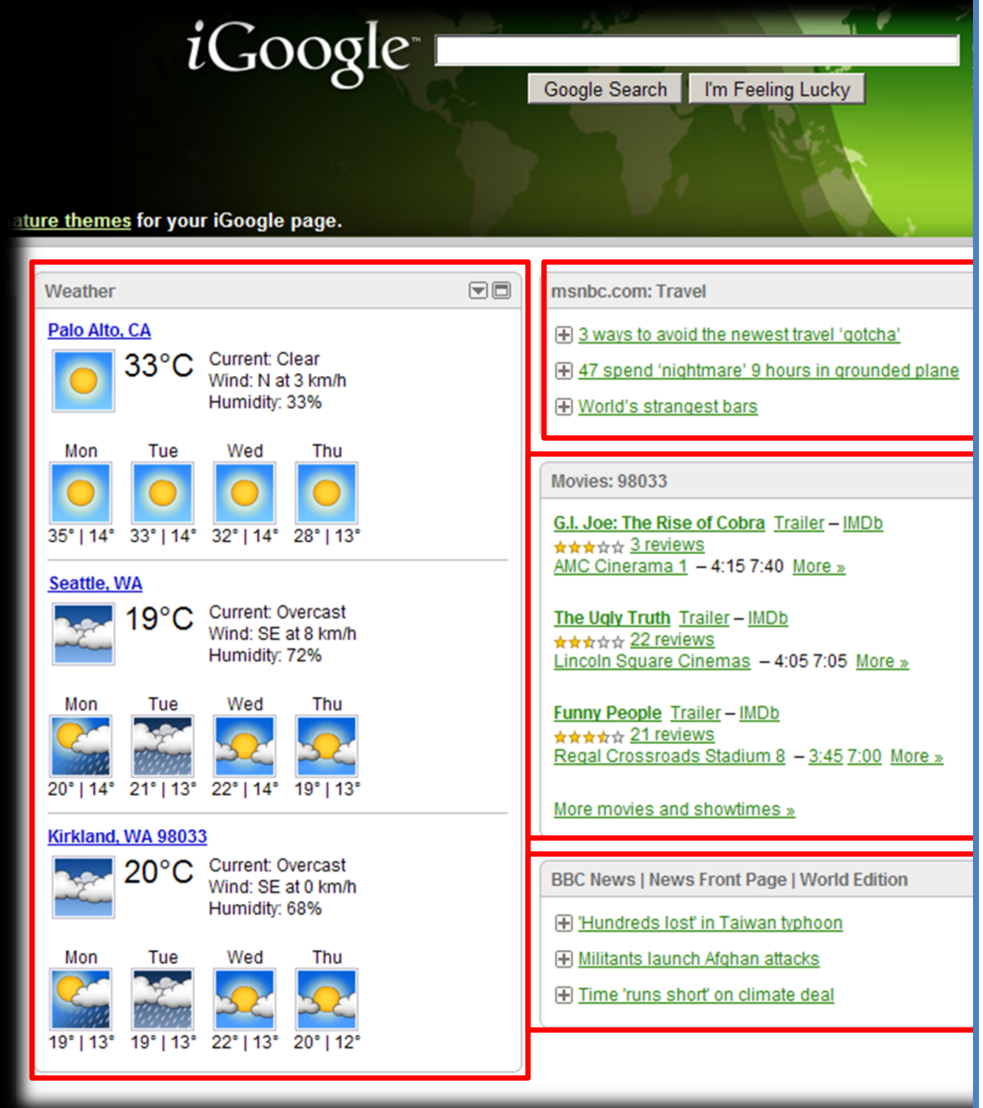
```
alert(‘hi’);
```

Gatekeeper


[JavaScript Application]

hi

JavaScript Widgets



iGoogle

Google Search | I'm Feeling Lucky

ature themes for your iGoogle page.

Weather

Palo Alto, CA

33°C Current: Clear
Wind: N at 3 km/h
Humidity: 33%

Mon	Tue	Wed	Thu
35° | 14°	33° | 14°	32° | 14°	28° | 13°

Seattle, WA

19°C Current: Overcast
Wind: SE at 8 km/h
Humidity: 72%

Mon	Tue	Wed	Thu
20° | 14°	21° | 13°	22° | 14°	19° | 13°

Kirkland, WA 98033

20°C Current: Overcast
Wind: SE at 0 km/h
Humidity: 68%

Mon	Tue	Wed	Thu
19° | 13°	19° | 13°	22° | 13°	20° | 12°

msnbc.com: Travel

- 3 ways to avoid the newest travel 'gotcha'
- 47 spend 'nightmare' 9 hours in grounded plane
- World's strangest bars

Movies: 98033

G.I. Joe: The Rise of Cobra Trailer – IMDb
★★★☆☆ 3 reviews
AMC Cinerama 1 – 4:15 7:40 More »

The Ugly Truth Trailer – IMDb
★★☆☆☆ 22 reviews
Lincoln Square Cinemas – 4:05 7:05 More »

Funny People Trailer – IMDb
★★★★☆ 21 reviews
Regal Crossroads Stadium 8 – 3:45 7:00 More »

More movies and showtimes »

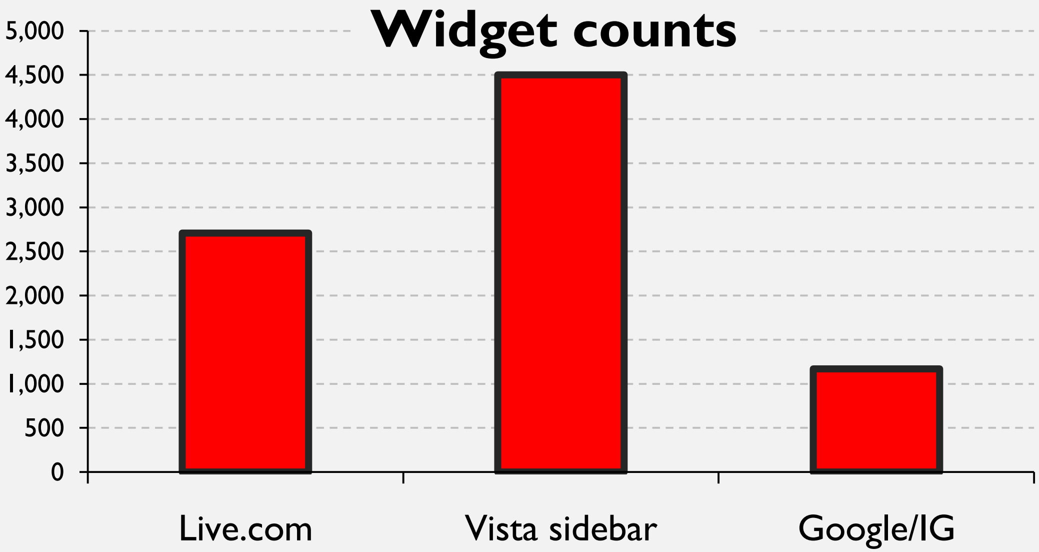
BBC News | News Front Page | World Edition

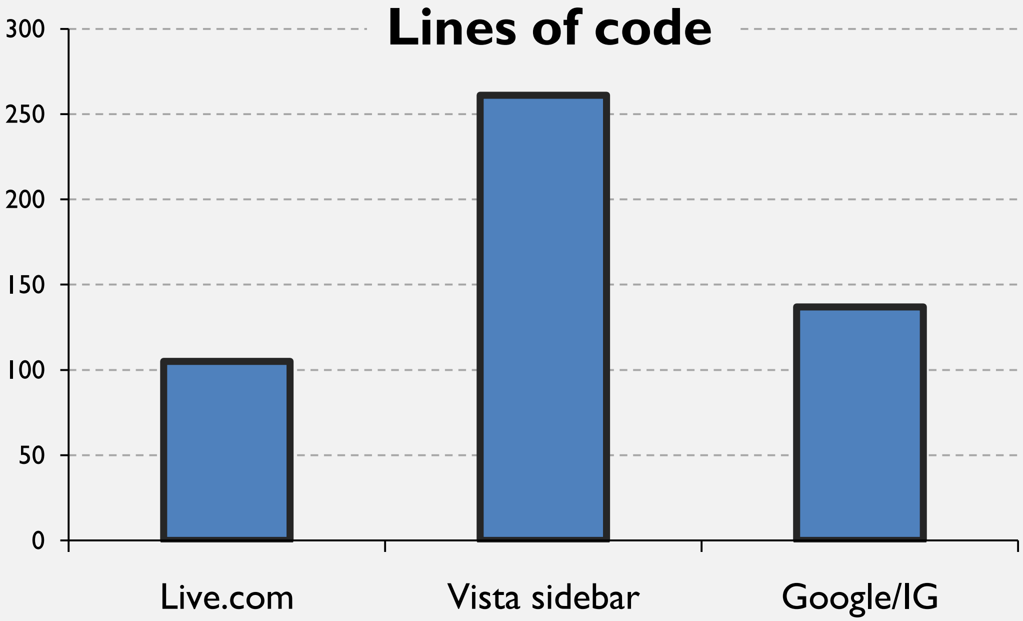
- 'Hundreds lost' in Taiwan typhoon
- Militants launch Afghan attacks
- Time 'runs short' on climate deal

```
// register your Gadget's namespace
registerNamespace("GadgetGamez");

// define the constructor for your Gadget (this must match the name in the manifest xml)
GadgetGamez.gg2manybugs = function(p_elSource, p_args, p_namespace) {
   // always call initializeBase before anything else!
   GadgetGamez.gg2manybugs.initializeBase(this, arguments);

   // setup private member variables
   var m_this = this;
   var m_el = p_elSource;
   var m_module = p_args.module;


   /****************************************
   **          initialize Method
   ****************************************/
   // initialize is always called immediately after your object is instantiated
   this.initialize = function(p_objScope)
   {
      // always call the base object's initialize first!
      GadgetGamez.gg2manybugs.getBaseMethod(this, "initialize", "Web.Bindings.Base").call(this, 
p_objScope);

      var url = "http://www.gadgetgamez.com/live/2manybugs.htm"

      m_iframe = document.createElement("iframe");
      m_iframe.scrolling = "yes";
      m_iframe.frameBorder = "0";
      m_iframe.src = url;
      m_iframe.width="95%";
      m_iframe.height="250px";
      p_elSource.appendChild(m_iframe);

   };
   GadgetGamez.gg2manybugs.registerBaseMethod(this, "initialize");


   /****************************************
   **          dispose Method
   ****************************************/
   this.dispose = function(p_blnUnload) {
      //TODO: add your dispose code here

      // null out all member variables
      m_this = null;
```

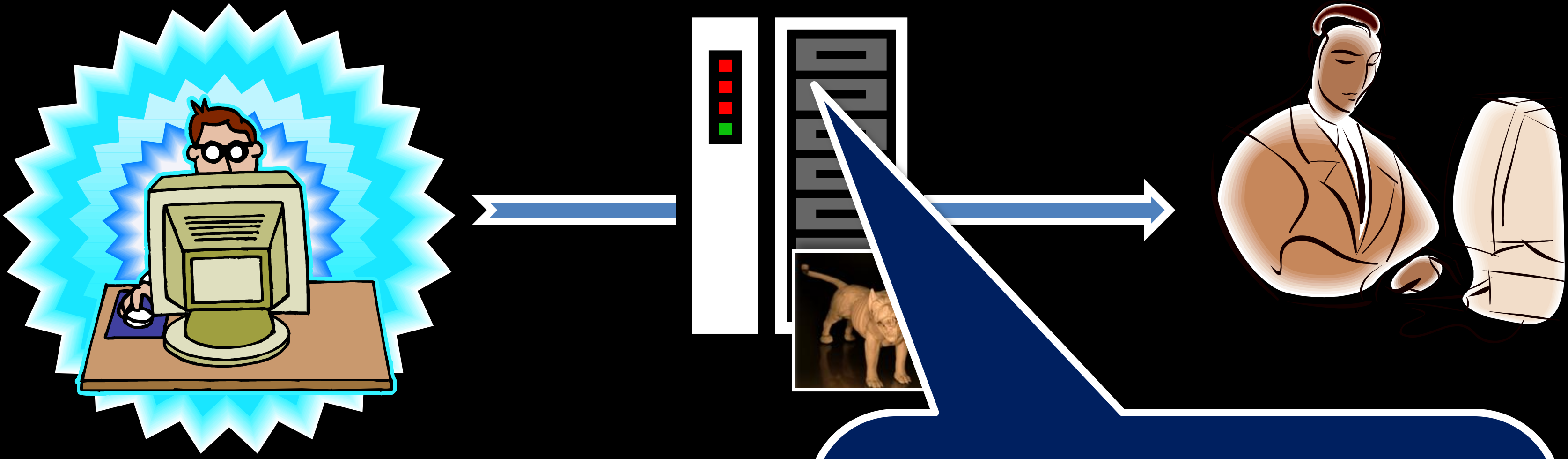
Widgets are everywhere…

We use over 8,500 widgets to evaluate Gatekeeper





Gatekeeper: Deployment Step on Widget Host



Widget:

```
...
alert('hi');
...
```

Hosting site: control widgets

by enforcing policies:

- **No alert**
- **No redirects**
- **No document.write**

Outline

- **Statically analyzable subset JavaScript$_{\mathbf{SAFE}}$**
- **Points-to analysis for JavaScript**
- **Formulate nine security & reliability policies**
- **Experiments**

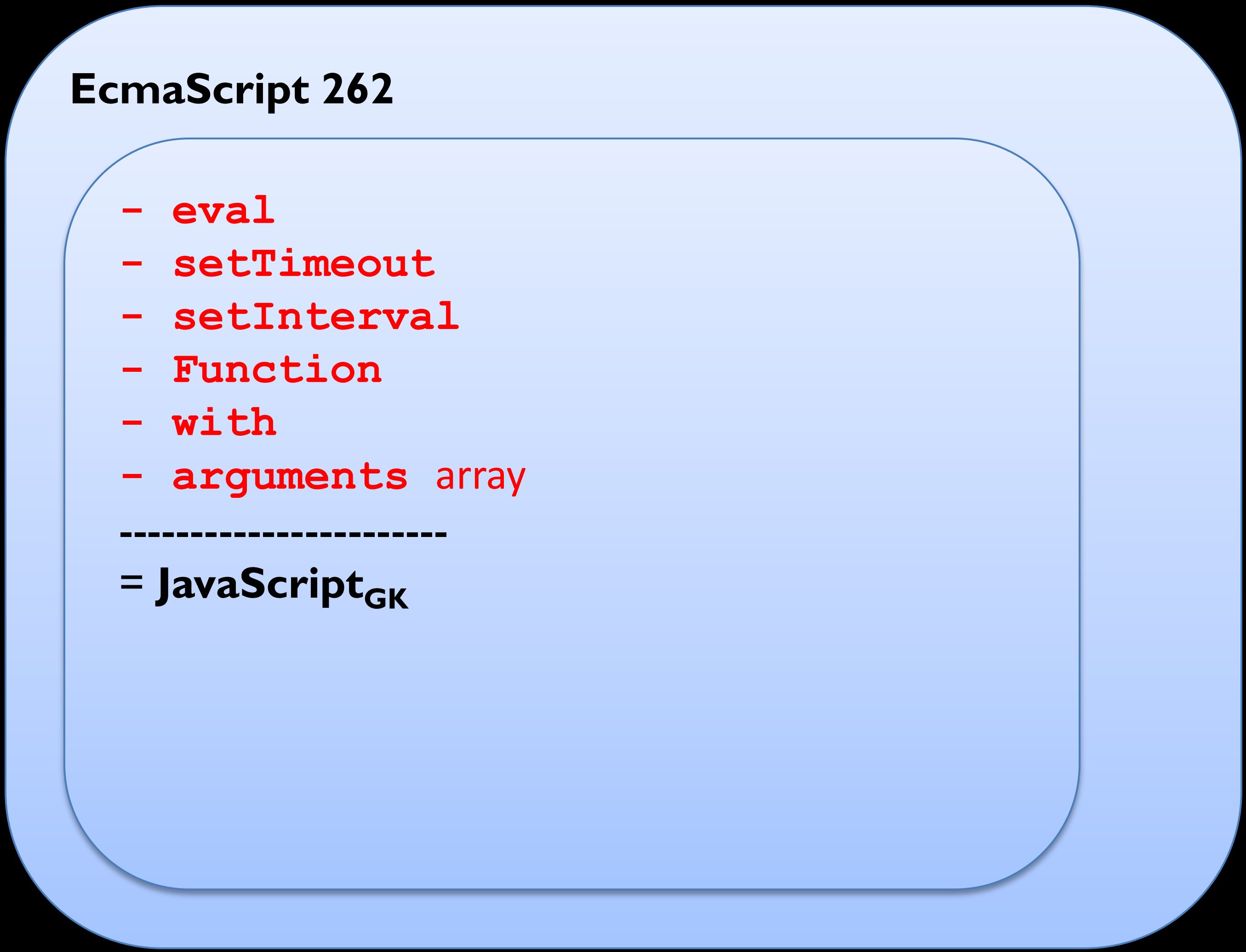
Techniques

Start with Entire JavaScript…

EcmaScript-262

```
var e = new Function(“eval”);
e.call(
    “do”+”cu”+”ment.write(”…”);

var e = new
  Function(unescape(“%65%76%61%6C”));
  e.call(“do”+”cu”+”ment.write(”…”);
```

Remove `eval` & Friends…

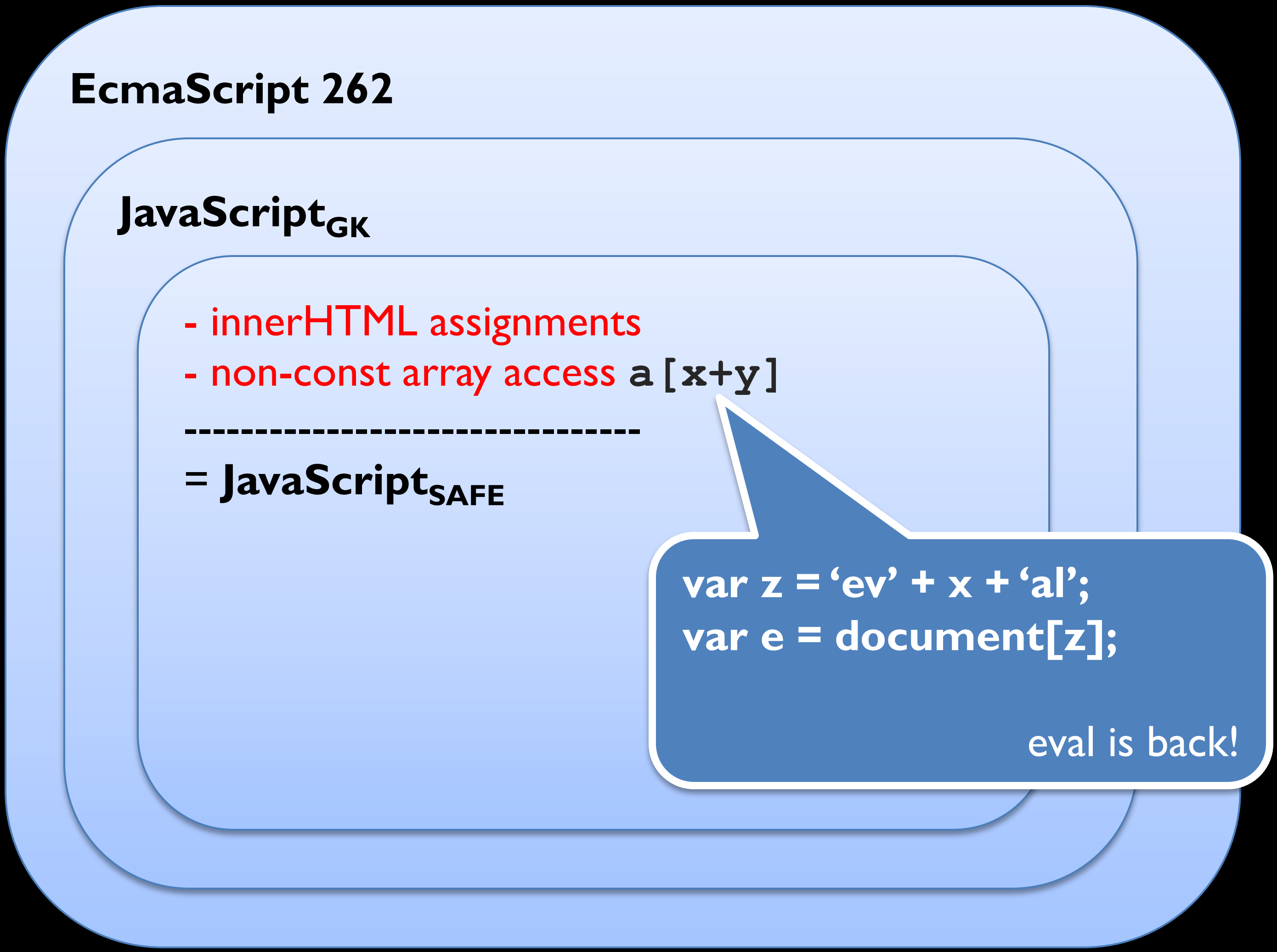


EcmaScript 262

- `eval`
- `setTimeout`
- `setInterval`
- `Function`
- `with`
- `arguments` array

= **JavaScript$_{GK}$**

Remove Unresolved Array Accesses…



EcmaScript 262

JavaScript$_{GK}$

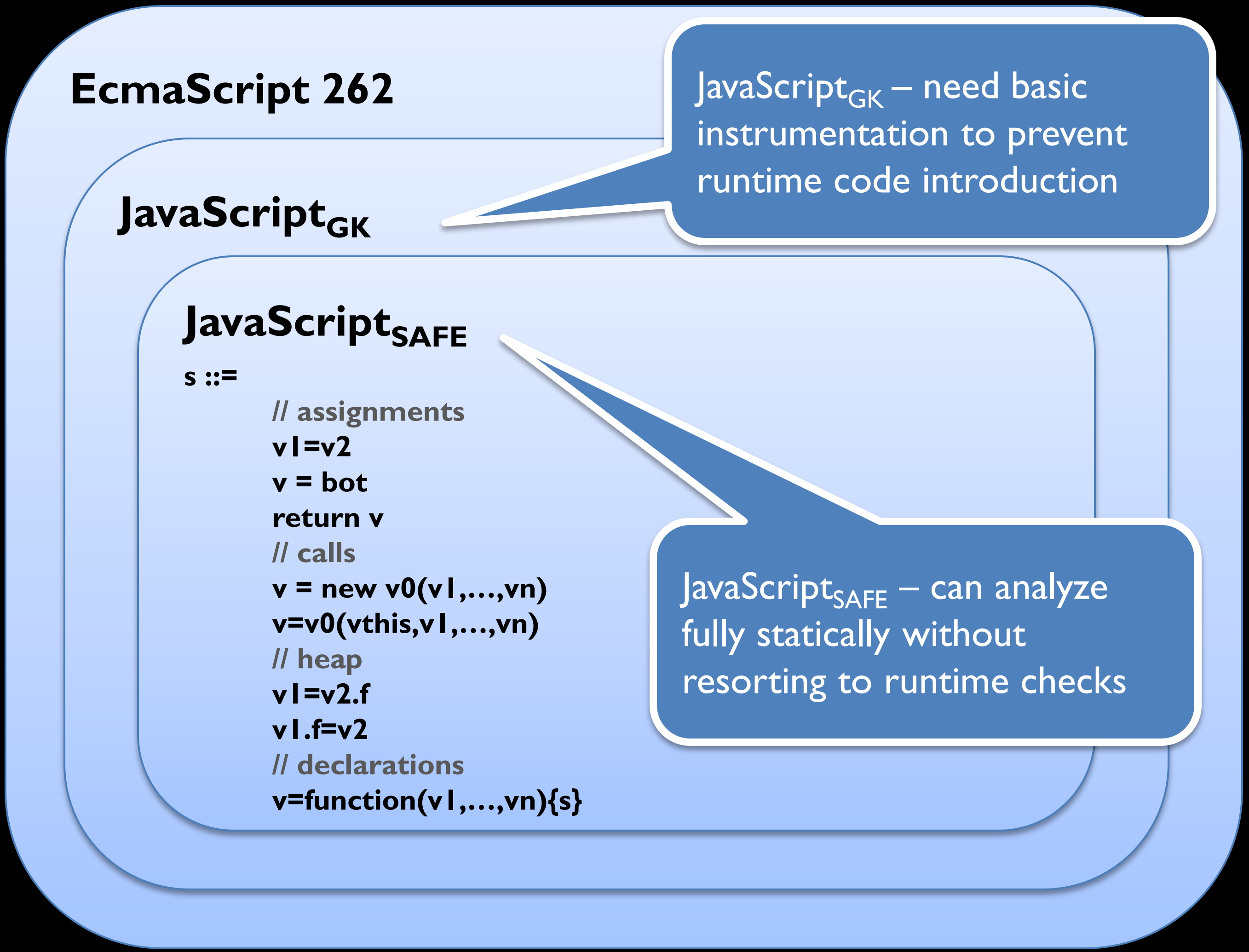
- innerHTML assignments
- non-const array access `a[x+y]`

= **JavaScript$_{SAFE}$**

```
var z = ‘ev’ + x + ‘al’;
var e = document[z];
```

eval is back!

Now, this is Amenable to Analysis!



EcmaScript 262

JavaScript_{GK}

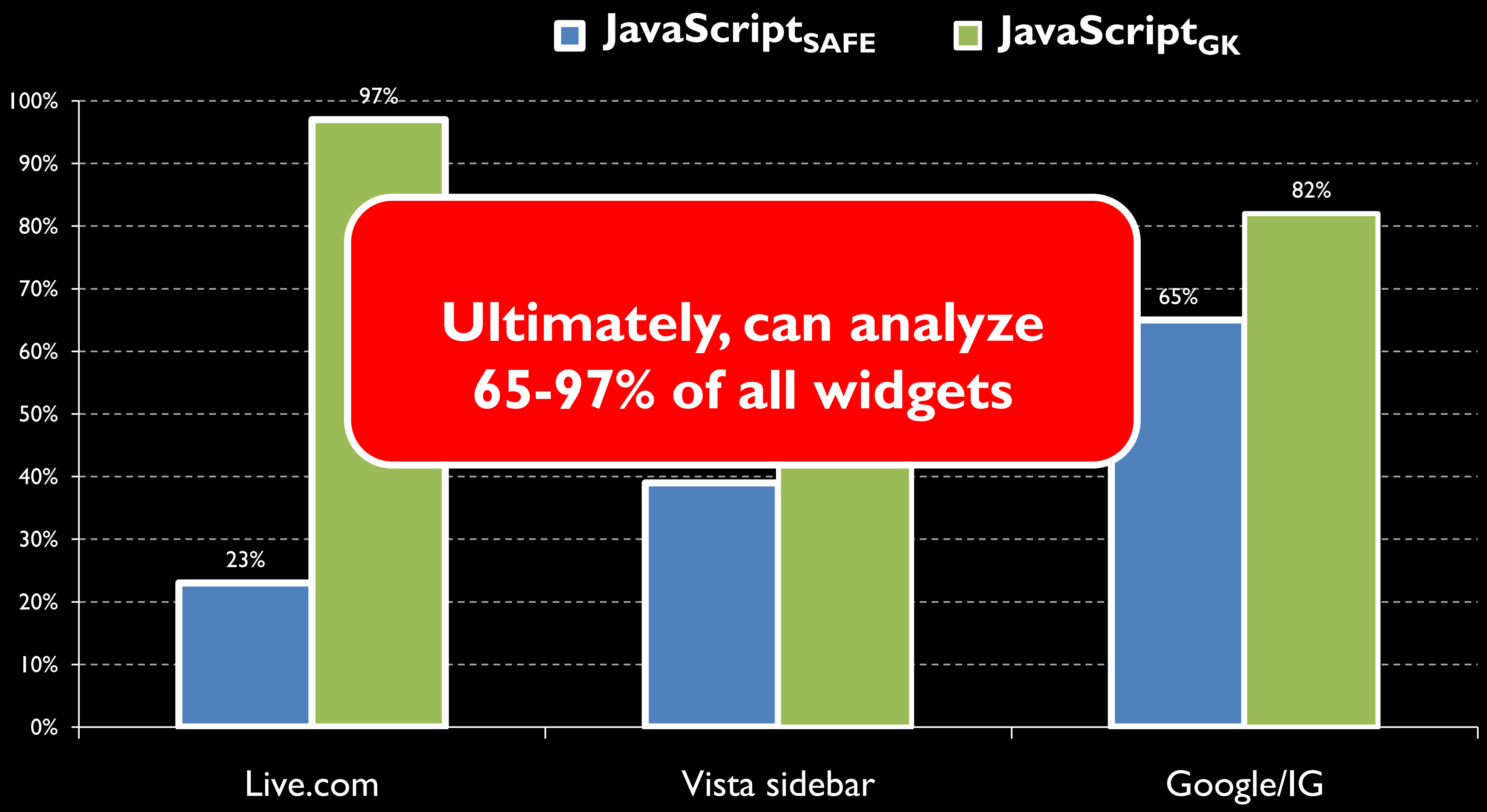
JavaScript_{SAFE}

```
s ::=
        // assignments
        v1=v2
        v = bot
        return v
        // calls
        v = new v0(v1,...,vn)
        v=v0(vthis,v1,...,vn)
        // heap
        v1=v2.f
        v1.f=v2
        // declarations
        v=function(v1,...,vn){s}
```

JavaScript_{GK} – need basic instrumentation to prevent runtime code introduction

JavaScript_{SAFE} – can analyze fully statically without resorting to runtime checks

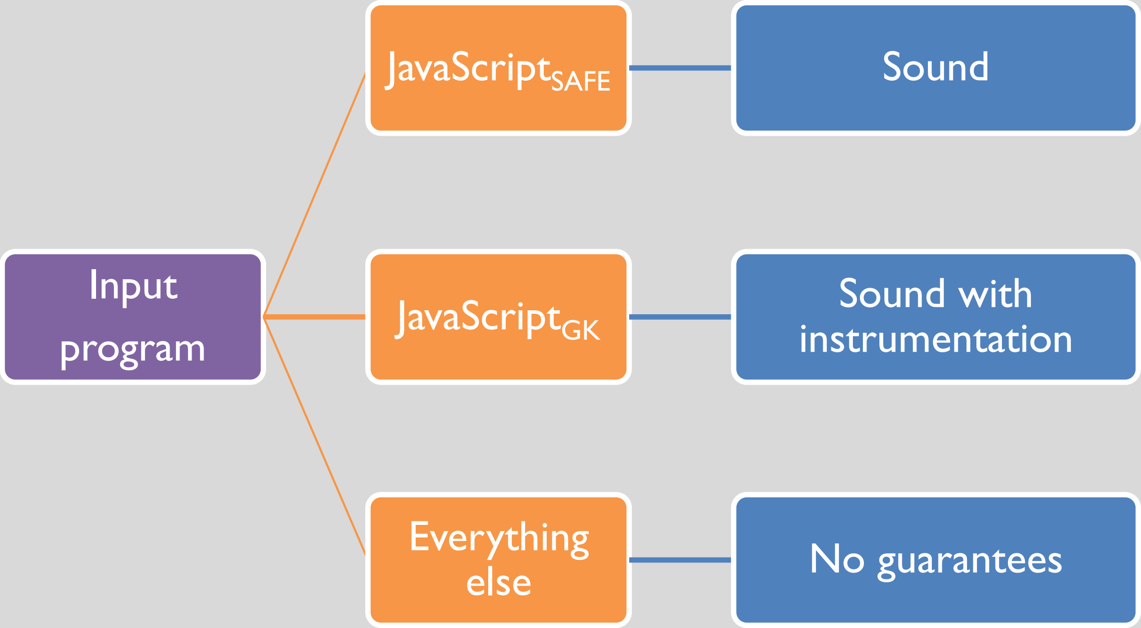
How Many Widgets are in the Subsets?



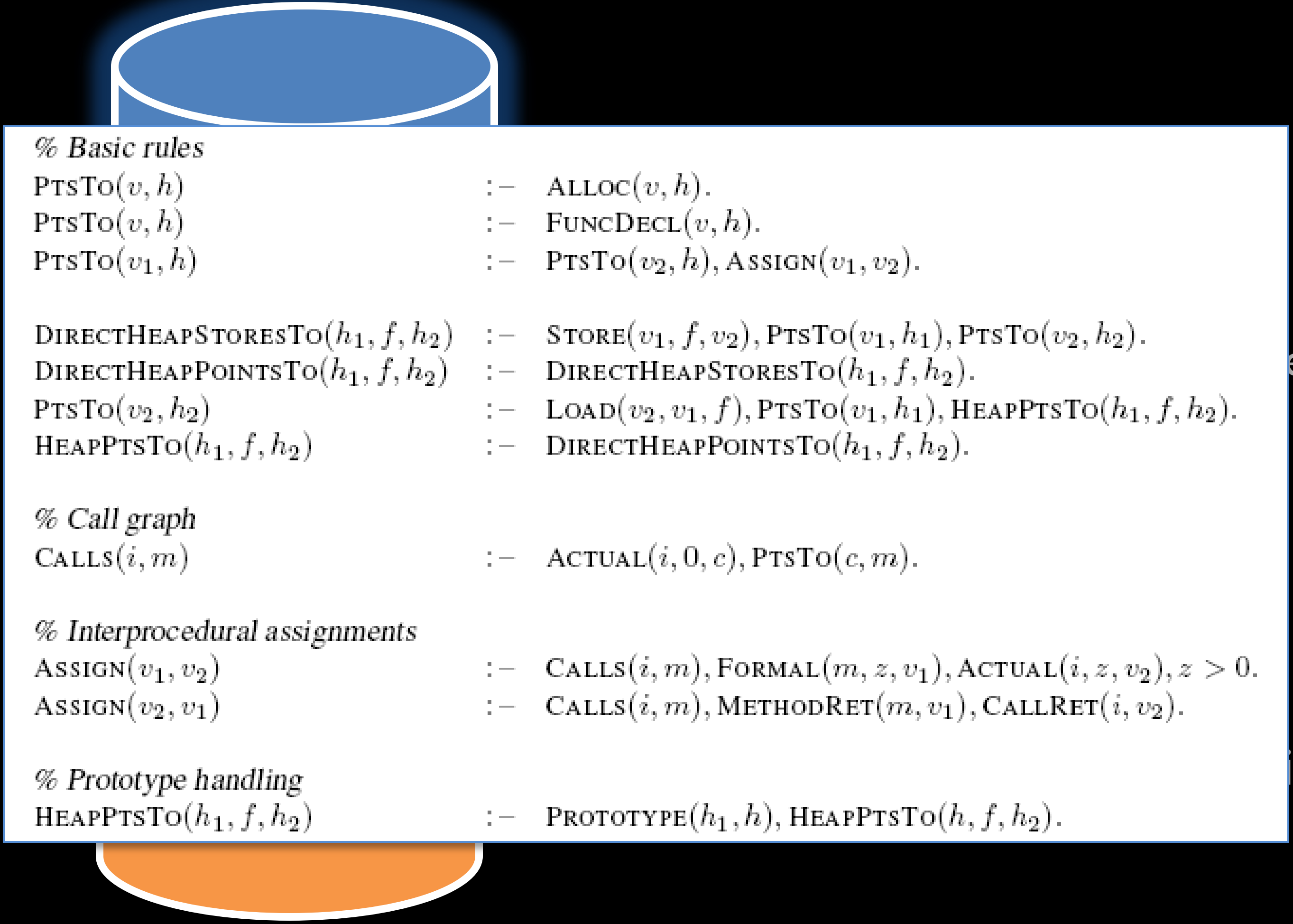
Ultimately, can analyze 65-97% of all widgets

Sound analysis:

ensures that our policy checkers find *all* violations



Points-to Analysis in Gatekeeper



% Basic rules

$\textsc{PtsTo}(v, h) \quad :- \quad \textsc{Alloc}(v, h).$

$\textsc{PtsTo}(v, h) \quad :- \quad \textsc{FuncDecl}(v, h).$

$\textsc{PtsTo}(v_1, h) \quad :- \quad \textsc{PtsTo}(v_2, h), \textsc{Assign}(v_1, v_2).$

$\textsc{DirectHeapStoresTo}(h_1, f, h_2) \quad :- \quad \textsc{Store}(v_1, f, v_2), \textsc{PtsTo}(v_1, h_1), \textsc{PtsTo}(v_2, h_2).$

$\textsc{DirectHeapPointsTo}(h_1, f, h_2) \quad :- \quad \textsc{DirectHeapStoresTo}(h_1, f, h_2).$

$\textsc{PtsTo}(v_2, h_2) \quad :- \quad \textsc{Load}(v_2, v_1, f), \textsc{PtsTo}(v_1, h_1), \textsc{HeapPtsTo}(h_1, f, h_2).$

$\textsc{HeapPtsTo}(h_1, f, h_2) \quad :- \quad \textsc{DirectHeapPointsTo}(h_1, f, h_2).$

% Call graph

$\textsc{Calls}(i, m) \quad :- \quad \textsc{Actual}(i, 0, c), \textsc{PtsTo}(c, m).$

% Interprocedural assignments

$\textsc{Assign}(v_1, v_2) \quad :- \quad \textsc{Calls}(i, m), \textsc{Formal}(m, z, v_1), \textsc{Actual}(i, z, v_2), z > 0.$

$\textsc{Assign}(v_2, v_1) \quad :- \quad \textsc{Calls}(i, m), \textsc{MethodRet}(m, v_1), \textsc{CallRet}(i, v_2).$

% Prototype handling

$\textsc{HeapPtsTo}(h_1, f, h_2) \quad :- \quad \textsc{Prototype}(h_1, h), \textsc{HeapPtsTo}(h, f, h_2).$

e fly

in Datalog

Datalog Policy for Preventing `document.write`

```
DocumentWrite(i) :-
   PointsTo("global", h1),
   HeapPointsTo(h1, "document", h2),
   HeapPointsTo(h2, "write", h3),
   Calls(i, h3).
```

```
document.write('<iframe id="dynstuff" src=""
'+iframeprops+'></iframe>')
```

Experimental Evaluation

Policies for Widget Security & Reliability

36 lines

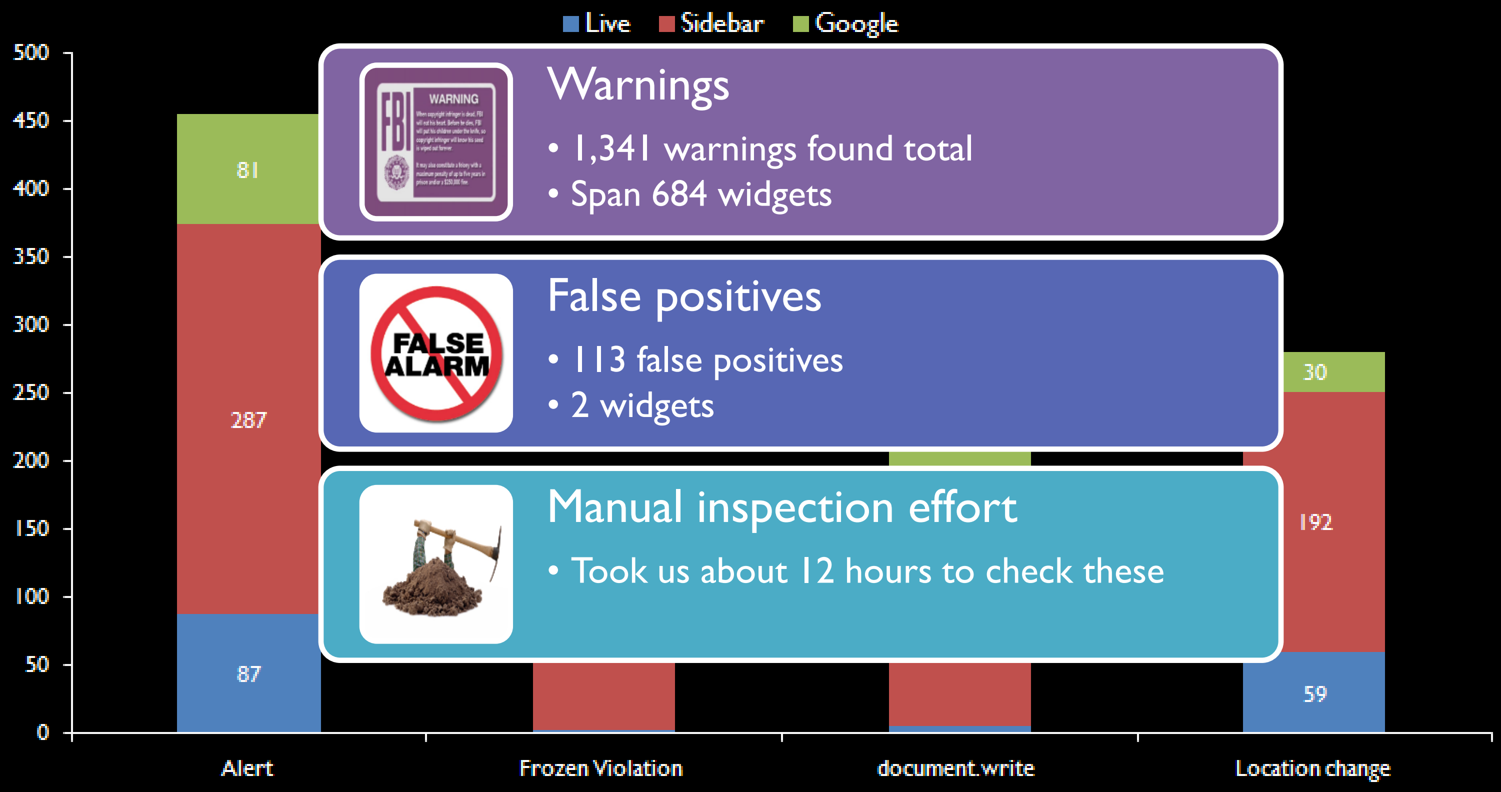
```
AlertCalls(i)    :- PointsTo("global", h), HeapPointsTo(h, "alert", h2), Calls(i, h2) .

DocumentWrite(i)    :- PointsTo("global", h1), HeapPointsTo(h1, "document", h2), HeapPointsTo(h2, "write", h3), Calls(i, h3) .
DocumentWrite(i)    :- PointsTo("global", h1), HeapPointsTo(h1, "document", h2), HeapPointsTo(h2, "writeln", h3), Calls(i, h3) .
InnerHTML(v)        :- Store(v, "innerHtml", _) .

BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "String", h) .
BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Date", h) .
BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Array", h) .
BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Boolean", h) .
BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Math", h) .

BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Function", h) .
BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Document", h) .
BuiltinObject(h) :- PointsTo("global", h1), HeapPointsTo(h1, "Window", h) .

Reaches(h1, f, h2) :- HeapPointsTo(h1, f, h2) .
Reaches(h1, f, h2) :- HeapPointsTo(h1, _, h), Reaches(h, f, h2) .

FrozenViolation(v, h1) :- Store(v, _, _), PointsTo(v, h1), BuiltinObject(h1) .
FrozenViolation(v, h1) :- Store(v, _, _), PointsTo(v, h1), BuiltinObject(h2), Reaches(h2, f, h1) .

LocationObject(h)         :- PointsTo("global", h1), HeapPointsTo(h1, "location", h) .
WindowObject(h)             :- PointsTo("global", h1), HeapPointsTo(h1, "window", h) .

StoreToLocationObject(h)  :- PointsTo("global", h1), HeapPointsTo(h1, "window", h2), DirectHeapStoreTo(h2, "location", h) .
StoreToLocationObject(h)  :- PointsTo("global", h1), HeapPointsTo(h1, "document", h2), DirectHeapStoreTo(h2, "location", h) .
StoreToLocationObject(h)  :- PointsTo("global", h1), DirectHeapStoreTo(h1, "location", h) .

StoreInLocationObject(h)  :- LocationObject(h1), DirectHeapStoreTo(h1, _, h) .

CallLocationMethod(i)     :- LocationObject(h), HeapPointsTo(h, "assign", h1), Calls(i, h1) .
CallLocationMethod(i)     :- LocationObject(h), HeapPointsTo(h, "reload", h1), Calls(i, h1) .
CallLocationMethod(i)     :- LocationObject(h), HeapPointsTo(h, "replace", h1), Calls(i, h1) .

WindowOpenMethodCall(i)   :- WindowObject(h1), HeapPointsTo(h1, "open", h2), Calls(i, h2) .
```

pply to all widgets

e.com only

a Sidebar only

Policy Checking Results



Warnings

- 1,341 warnings found total
- Span 684 widgets

False positives

- 113 false positives
- 2 widgets

Manual inspection effort

- Took us about 12 hours to check these

False Positives

common.js:

```
function MM_preloadImages() {
  var d=m_Doc;
  if(d.images){
    if(!d.MM_p) d.MM_p=new Array();
    var i,j=d.MM_p.length,

a=MM_preloadImages.arguments;
    for(i=0; i<a.length; i++)
      if (a[i].indexOf("#")!=0){
        d.MM_p[j]=new Image;
        d.MM_p[j++].src=a[i];
      }
    }
}
```

- Why not more false positives?
  - Most violations are local
  - But this is policy-specific – a global taint policy might produce other results

Conclusions

Gatekeeper: Static analysis for JavaScript

Technique: points-to analysis

Focus: analyzing widgets

Results:

- **1,341 policy violations**
- **false positives affect 2 widgets**

Contact us

Gatekeeper security project MSR _

GATEKEEPER: Mostly Sta
Reliability Po

Salvatore
Universi

Abstract

The advent of W
client-side co
This code
other c