Minutes of the: *2nd meeting of Ecma TC39 Special group on Secure ECMAScript*

held in: *Kona, HI, USA*

on: *18 November 2008*

Chairman: Mr. John Neumann (Microsoft/Ecma International)

Vice-Chairman: Vacancy

Secretary: Mr. John Neumann (Microsoft/Ecma International)

Attending: Mr. Douglas Crockford (Yahoo!), Mr. Alexc Daley (Microsoft), Mr. Brendan Eich (Mozilla), Mr. Cormac Flanagan (UCSC), Mr. Dave Herman (Northeastern University), Mr. Scott Idaacs (Microsoft), Mr. Waldemar Horwat (Google), Mr. Mark S. Miller (Google), Mr. Mike Samuel (Google), Mr. Rob Sayre (Mozilla) and Mr. Allen Wirfs-Brock (Microsoft).

On phone: Mr. Tyler Close (HP) and Mr. David-Sarah Hopwood (Industrial Systems, UK)

1 Opening, welcome and roll call

Introduction of the attendees.

2 Adoption of the agenda (08/092)

The agenda was adopted as presented.

3 Battle of the Bandaids: Caja vs Web Sandbox

Work it out in committee instead of doing a competition ?

Cajita and Valija levels: simple vs. current web-compatible.

valija == Microsoft web sandbox

They differ in that web sandbox passes out real references to objects but uses the current context to limit what you can access on them (an ACL paradigm). Valija restricts reference passing (a capability paradigm). In web sandbox the host can create channels between two sandboxes, but only primitive values can be passed over a channel.

Valija: all contexts see the own properties of an object in the same way. They see the basic protype properties in the same way but differ if one monkey-patches a prototype; only that context sees that patch.

Jacaranda: pure object capability

Recent web sandbox discovered and fixed security bugs: arguments.prototype.caller.caller to get at the Function constructor; Function() to execute arbitrary code (didn't realize that it did the same as new Function()).

Mozilla monitors greater than cubic complexity in regular expressions.

Doug Crockford: Get rid of all native prototypes, constructor fields, etc.

Problem with catch-alls/interceptors. Assertion was made that an interceptor is just like putting a getter and a setter on every property. However, this behaves materially differently for prototype objects: just the mere presence of a getter or a setter on a prototype prevents one from writing to create expandos in the derived object.

How do multiple contexts, as implemented by prototype inheritance on the built-in objects, interact with getters/setters and introspection?

other issues with interceptors are the ability to masquerade as other objects and run arbitrary user code for tests such as HasProperty. Also, the current spec assumes that the internal operations such as HasProperty, Put, etc. are consistent with each other.

How would iteration work with interceptors ?

Relying on an initial script to lock down/delete nasty properties from the global object + having eval do the evaluation in a virgin copy of the global object = oops !

More support for a stratified virtualization system where the outer program can do an "eval" in a virtual and separate inner universe with hooks for what happens on various property lookups, calls, etc.

Brendan: Catch-all introspection is extraordinarily difficult due to recursion suppresion (a simple flag won't work because the handlers may need to look up other things) and related complexities. Wouldn't want to go through that again.

4 Secure ECMAScript

Investigation of the minimal modification to ES3.1 to obtain a capability constrained language.

The first big problem is to make a language safe for ads in the web page. The first goal is security. We also need to make sure that we have complete and proper isolation. The second goal is to protect the applications. The third element is Mashups, and this takes solving the defensive code problem; Points raised include 1) primordial object freeze; 2) use lexical scope; 3) no overt channels not explicitly authorized; 4) eval operator; 5) elimination of constructor; 6) no prototypes (replaced by API); 7) “Use strict 3.1” is a required basis.

The question of threat models was brought up and discussion continued on that subject.

A suggestion was made to put in hooks to virtualize ECMAScript, and let several implementations solve the problem and see what works before standardizing solution. This approach was discussed and it was agreed that it was a simple thing that could be done quickly and get some early results. The approach is going to be called “webfoot”

Concern over work being done in W3C on HTML 5, and it may be desirable to send communication to W3C about what we are trying to solve and suggest that we try to work together to avoid conflict or “turf” issues.Also a desire to communicate with IAB. I will work with Douglas Crockford to craft the statements fro Ecma to W3C groups.

New name: "webfoot" for the concept of providing hooks for sandboxing ad code.

Some folks want to vastly reduce the scope of or delay HTML 5 to make securing it easier.

5 Romancing the DOM

Investigation of the minimal modifications to HTML and the Document Object Model to provide cooperative containment of widgets.

6 Any other business

7 Date and place of the next meeting(s)

1/27/2009 Sunnyvale

3/24/2009 Sunnyvale

8 Closure

Kumbaya.