Caja: Defending Against Untrusted Javascript

Summary

Caja is a Javascript-to-Javascript translator that effectively sandboxes untrusted Javascript code. It allows the containing application to completely control what the untrusted script can access but does this so that well-behaved Javascript is unaware that it has been confined.

Caja will run in any browser without a plugin, since the output is standard Javascript.

The Translation

The core of the Caja translation is to transform the untrusted script such that there are no free variables - instead, all variables that were originally free become properties of an object supplied by the container. This means that the container can supply fake or "tamed" versions of the objects that are normally globally available. This means that any API or other global object can be completely controlled by the container, either by omitting it entirely, diluting it (for example, omitting certain functions) or wrapping it so that the container can control fine details of the API's use.

The container can also intercept gets and sets of properties of objects, replacing them with functions, so that even these can be controlled in a fine-grained way.

Applications

The Caja team's initial target has been the world of Web Gadgets, such as are seen on OpenSocial platforms, iGoogle, Facebook and Yahoo!'s Application Platform. In this world, the containing page (for example, the iGoogle page) is assumed to be "trusted" (the user chose to visit it, after all), but gadgets displayed within it are not - for example, they might try to steal cookies, navigate the page in order to perform phishing, replace password boxes with controls of their own and so forth.

Currently the best defence against this is to use iframes, but they are far from a complete defence, they are also very much all-or-nothing and they impose some painful overheads on gadgets, particular for interframe communication.

In contrast, Caja allows the container to choose in a very fine-grained way what the application can do - for example, it can be allowed access to the entire OpenSocial API, but only permitted to see its own subtree in the DOM. Inter-gadget communication becomes very cheap and easy - one gadget gives the other an object with methods on it, and the other calls those methods to communicate with the first gadget. Because of Caja, it is possible for this object to be a normal rich Javascript object that the second gadget cannot "see" inside. Because of this Caja also allows mutual suspicion between gadgets whilst still permitting rich interaction.

Some other ways to deploy Caja include

- Browser-based deployment - this would require a plugin, of course, but then the user could choose which pages had what capabilities. This mode would seem most relevant to the mobile API case. See below.

- Server-side execution. An increasingly widely offered service is to run untrusted user code on machines "in the cloud". Various sandboxing schemes are used to keep the servers safe. Caja provides a detailed, principled and fine-grained way to achieve this sandboxing.

- "Macros" - in a web application a user might want to add custom functionality - but to protect themselves from coding errors or attack by malicious other users of the application, they might choose to use Caja to reduce the authority of their macros.

Application to Mobile APIs

Clearly the client side variant could be used to control access to APIs, using a wide variety of permissioning schemes - for example, policy-based, via user interaction, by consulting a server, or whatever - essentially anything you can write in Javascript. It can also limit the damage malicious Javascript can cause in general. All that is needed is the ability to hook some Java into the browser, to pre-render the page before the browser gets access to it.

Some Details

Caja is an open source project, written in Java, licensed under the Apache Licence. Currently most of the work has been done by Google engineers, with some contribution from Yahoo! and also from Tobie Langel, one of the Prototype.js team (Prototype is a very widely used Javascript library). It can be downloaded from http://code.google.com/p/google-caja/. It can be played with in our testbed, http://cajadores.com/demos/testbed/ and it can also be seen live in various platforms, such as Yahoo!'s new Application Platform (where it is mandatory), iGoogle and Orkut (where it is optional).