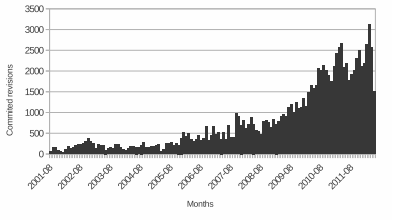
The Security History of the WebKit Browser Engine

Renáta Hodován

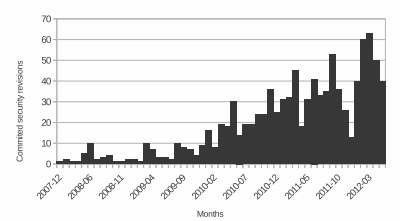
Over the years, more and more emphasis has been placed on the web domain. Customers are constantly requesting new features in web-applications and in the hosting browser itself as well. As these requests are fulfilled, the software becomes more and more complex and it's getting harder to take care of all side effects of the modifications. This can lead to undesired behaviour along with security vulnerabilities.

At the conference, I'll focus on the browser component and take a closer look at the security evolution of the WebKit layout engine. Among others, WebKit powers the Apple Safari and Google Chrome browsers and it holds nearly 36% of the browser market according to the survey of StatCounter in February 2012. Since WebKit is an open-source project, all implementation details and most bug entries are available. In this survey, I investigate a dataset that was generated from the publicly accessible security bug entries of WebKit. Furthermore, I determine a trend of the number of introduced security bugs over revisions and show the relation between the size of the code and the probability of the existing security holes.

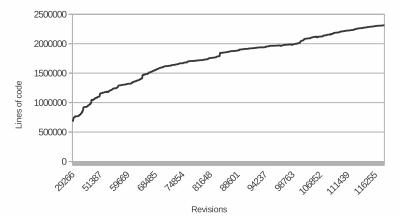
As Chart (a) shows, the number of committed revisions is growing increasingly over the months in the last few years. In line with this, the number of exposed security vulnerabilities follows the same trend, see Chart (b). This hangs together with the theorem that as a software is getting older and more complex [1], its maintenance and development turn into a hard challenge and we should count on undesired consequences. On Chart (c) we can see the way the size of the source code of the whole project is changing. And the last one shows how many lines of source code were modified during the months as a result of fixing security holes.



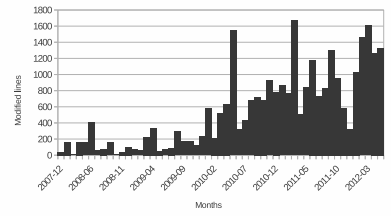
(a) Commited revisions over months



(b) Commited security bug fixes



(c) Code size over revisions



(d) The number of modified lines over security revisions

References

[1] Roger S. Pressman. *Software Engineering – A Practicioner's Approach*. McGraw-Hill, 5th edition, 2001.