1) Technical details (approach/technique) that you found novel/ Something specific you learned that you didn't know before

The idea that uses AST to analyze javascript is not novel to me. However, it is interesting to see all the following works they have done based on the idea, including automatically dealing with dynamic code generated using “eval” and dealing with javascript packers.

2)Could I have done this work if I had the idea why or why not?

I think we could do part of the work. However, the authors make use of the an “Oracle”, the application Wepawet which is also from UC. With this they are able to observe the change of malwares corresponding to the improvement they have made, in other words, the game escalation. We are not able to do this part of work.

3)Is there anything I could do to repeat or validate?

One thing I want to validate is the effectiveness of Wepawet. I want to treat Wepawet as a blackbox and with a series of test check which part in a malicious javascript can trigger the alert.

4) What is my best idea for follow on work that I could personally do?

I feel like we can do some improvement to the Revolver system when dealing with packers. The author mentioned that they deal with packers by adjusting Minimum pattern size. Although this is a valid way to distinguish packed code between non-packed one, I didn’t see them mentioned that they have done works of unpacking the code, which is the reason all packed code samples are categorized as “Data-Dependency”. By recognizing some of the common pattern of packed code and unpack them accordingly, I think a better and more accurate result can be achieved.

5)What is my best idea for follow on work that I'd like to see the authors do?

I noticed that in the evaluation report, the count of Data-Dependency cases is much bigger than others. (6,996 scripts of injection, 101,039 scripts with data-dependencies, 4,147 evasive scripts, and 2,490 scripts as general evolutions). I want to see the author to elaborate more about their work on analyze the data-dependencies case. Such a big number may imply insufficient analysis to the case.

6)Any logistical experimental lessons I learned?

In this paper a lot of interesting resource that can be used to do similar analysis against javascript is mentioned. These are valuable resources.

HtmlUnit: http://htmlunit.sourceforge.net/

Wepawet: http://wepawet.iseclab.org/

7)How does this compare to the other papers we read? Most similar? How different? Other comparisons?

8)What is your biggest criticism of the paper?

Generally speaking this paper is good. However, I think I want to see more comparison result between the cases when different threshold values are chosen.