DIRTY COW

Summary

This paper makes a very proper introduction to the thema: Dirty COW. It is quite fascinating as a topic as it was a significant problem in the Linux OS. The paper talks about how the linux kernel works and how it could be exploited before it was patched by the dirty COW attack. The attack is shown with some code parts as well as described with words. Some other types of attack paths are handled in the paper as well as Huge Dirty COW. Madvise is also a decent comparison to the dirty COW. Overall this is a very good topic and the paper is till now very well structured and covers the best topics one needs to know about the topic, regardless if he is a newbie in security/operating systems, with some experience or a professional.

Strengths

+ Very well structured overall, everything is logical and easy to follow

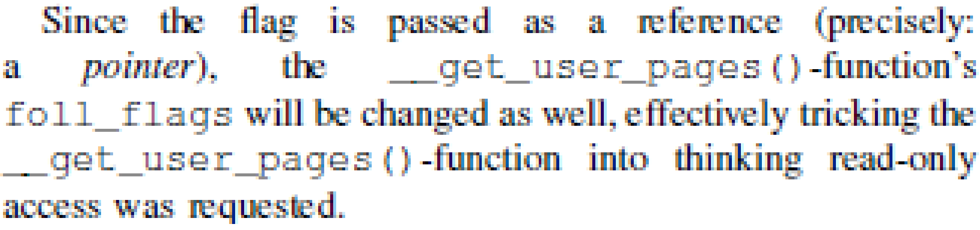
+ the language is very nice, easy and understandable

+ the overall amount of code doesn't prevent the understanding of the paper, but helps it even more

+Very good overall

Weaknesses

- Introduction could be a bit longer, maybe some more general, easy to understand facts/statements that could lead up to a good introduction to the topic.
- The code itself isn't a problem but there are many function names in the text, that make the text a bit harder to understand when some of them could be spared. For example: be broken by faultin_page() again and again. To break this loop, faultin_page() remove One of the faultin page function can be removed by using the word "it" or something. Same here:



Since the flag is passed as a reference (precisely: a *pointer*), the `__get_user_pages()`-function's `foll_flags` will be changed as well, effectively tricking the `__get_user_pages()`-function into thinking read-only access was requested.