
PEER REVIEW MEMO

TO: CECELIA MUSSELMAN
FROM: NATHAN HUNT
SUBJECT: PROJECT 1 PEER REVIEW
DATE: JANUARY 25, 2016
CC: NICOLAS MEWS-SCHMUCK

Summary: The paper focuses on the different types of penetrative testing used to purposefully break software and expose bugs in the code. Tools like Metasploit make it convenient for developers to quickly and easily test various exploits in software; tools like this (and software componentization, explained later) are priceless in competitions like the Collegiate Cyber Defense Competition (CCDC) where programmers must be able to create code on the fly and adapt to unforeseen problems. Testing for all possible exploits reveals cracks in security, which must be patched up in order for software to function properly and predictably.

Major Points: Overall, the paper is solid. The topics flow into each other and the train of thought links logically across all the paragraphs. The paper also does well in covering the theory aspect of the prompt by including CCDC, HD Moore, and Douglas McIlroy. However, the author's personal connection to the topics mostly disappears after the first paragraph. I.e. the author mentions how they learned all the better ways to develop software, but does not mention how they feel about any of it or how it might have affected them. Overall, the paper is best in the first paragraph where it not only explains what is happening but also highlights the compelling aspects of the topic and reveals genuine interest from the author. I see why you like CCDC, but how do you feel about software componentization? Or McIlroy? The Wall-Street description is a good little example because it shows a real-life situation where software design is critical, and the paper discusses it in a way that is perfect for me to understand.

Is this a paper about how different situations led the author to discover how to smartly develop software, or is the paper about strictly about the different aspects of software design? Personally, I think the best way to reform the paper is to keep all the information, but re-frame it to focus on the author discovering it and understanding why it is important. The paper might have a stronger voice if it's read as *Here is how I became a better programmer and had a blast*, instead of *Here's a few things about programming*.

Minor Points: The grammar and punctuation seems mostly perfect throughout, a few proofreads will likely eliminate any lingering errors. The only thing I noticed as lacking was diction; in many places adjacent sentences contained the same/similar words to describe the same idea. Varied word choice should allow for easier reading and for the author to add more detail.