

Is second-hand advice better than no advice at all?

 devblogs.microsoft.com/oldnewthing/20081118-00

November 18, 2008



Raymond Chen

Commenter Grow Up (if you're so grown up yourself, why not use your real name?) took issue with the second-hand advice I gave when the discussion of protecting sensitive data. In that discussion, I gave second-hand advice on how one could protect information, and one reader apparently thought I was trying to malign said second-hand advice or was holding it up as non-authoritative. (In case you forgot: Everything here is non-authoritative. It's all just my interpretation of the world around us. And that interpretation is often wrong. Don't make me bring back the nitpicker's corner.)

I added the second-hand advice only because upon re-reading, I noticed that I wrote a lot about what you shouldn't do but didn't write about what you should do. I didn't want to put my own neck on the line in an area I am not an expert, so I passed along second-hand advice instead, figuring second-hand advice was better than no advice at all.

It may surprise you that I am in fact an expert at very few things. I do have a rather extensive background in general programming principles, and I use that experience to "fill in the gaps" in places others may need help doing so. For example, I'm pretty good at the "Imagine what the world would be like if that were true" game because I've seen a good amount of the computer world and can think of scenarios that others may miss. On the other hand, I'm good at the "What if two people did this?" game only because I bother to play it at all. (Usually, the answer to the question "What if two people did this?" is obvious. People merely forget to ask it.) I'm also good at reverse-engineering history. I can see how something evolved and work out why it ended up that way.

So, suppose there's a topic that I know a little bit about but not enough to come up with an expert recommendation. For example, I may want to caution you against doing something but I don't have a good answer as to what you should be doing instead. Here are my options:

1. Become an expert in the topic and develop a personal recommendation. If it's worth doing, it's worth doing well.
2. Pass along somebody else's recommendation. Second-hand advice is better than none.
3. Develop an uninformed recommendation. Even though it's bad advice, at least it's my bad advice.

4. Don't provide any recommendation at all. If you can't stand by it, then don't write it.
5. Don't write the article in the first place. If you can't do a complete job, then don't do it.

The first option is not going to happen; I don't *want* to become an expert in the topic. The last one is also unpleasant, because I do want to warn against what I consider to be a bad practice, or at least I may have some thoughts that I think are worth sharing. The middle option isn't very pleasing either. Besides, this is a blog, not a textbook. If you want a thorough treatment of a topic, you're looking in the wrong place. There are plenty of other bloggers who throw out half-baked ideas. I should be able to do it, too.

Like I just did.

Raymond Chen

Follow

