Ken Thompson. Reflections on Trusting Trust
Review by Frank Fetters

This was an awesome paper which I thoroughly enjoyed reading. It is basically a speech that an award recipient gave reflecting on the values by which we trust things. It raised very interesting points regarding not trusting any code that you didn’t write.

My favorite part of this paper was the personal touches (i.e. the tax form comment). When I write I attempt to write papers as I would speak them. I feel that the more personal the paper is, the more the reader will associate and learn. (Similar to the way you learn more from the teachers who interact with you and are friendly). This paper was the perfect balance of starting as a funny friendly guy, who then delves into a very technical subject but is able to bring the reader along.

This paper poses the challenge of writing a program that output’s its own source code to the screen. While I had never thought of attempting this, I found it a very exciting idea. The paper then goes on to discuss how because the c compiler is written in c, it becomes a “learning” program, because you can teach it new tricks if you follow the stages outlined in this paper.

Being shown how one would add “\v” to become a new escape to a vertical tab was a great way to show how you can make changes to a compiler that will be undetectable by another user who trusts his compiler. This allows a malicious coder to introduce backdoors and trojans into all sorts of applications.

This paper showed how you should never trust any code you didn’t write. This distrust should extend to the compiler, since you didn’t write that. However, how do you write code without using a compiler that you didn’t write? The source for your new “safe” compiler has to be compiled somehow??