Trusted Computing in Context

Fred B. Schneider
Associate Editor in Chief

Much has been said about what makes cyberspace insecure and who's at fault. Software producers are often singled out as a source of remedy. Actions these producers take in the name of improving the security of cyberspace, however, are sometimes viewed with suspicion.

The so-called *trusted computing* technology embodied in the Trusted Platform Module (TPM) secure-coprocessor from the Trusted Computing Group (https://www.trustedcomputinggroup.org) is a case in point. Here, a hardware-based root of trust makes it possible for the system designer—not the computer owner—to regulate which programs can run on a given computer. If system designers are evil, then they can use trusted computing to prevent a competitors' programs from being installed, thus creating or preserving a monopoly. When computer owners are incompetent, though, benevolent system designers can use trusted computing to prevent malware from being installed (perhaps unwittingly) and run. Unlike most defenses, which the system operator controls, trusted computing provides a way to save naïve or incompetent computer owners from themselves. Most computer owners aren't computer scientists, and thus need this help. (Many computer scientists need this help, too.)

Trusted computing introduces a tension between the rights of computer owners and the (presumed) responsibilities of system designers. Would that this tension could be avoided! Perhaps system designers could discharge the responsibility of securing cyberspace in some other way. Nobody has yet devised such a way, but one might exist; it would not only involve eliminating vulnerabilities but also preventing human users from being spoofed into unwittingly aiding attackers. Or perhaps system designers shouldn't feel any responsibility at all, but then I become skeptical that a secure cyberspace could be built from components that would be available.

The right of computer owners to control what their computers execute is seen as sacrosanct by critics of trusted computing. I don't think it's that simple, and I see analogies with other rights and responsibilities of individuals in a society. For example, we all benefit from the cleaner environment that comes from limiting how individuals use property they own. Impinging on the rights of individuals here produces benefits for all. And, we all benefit from vaccinating everyone against a disease, even if it involves relinquishing some control over our bodies (and carries some risk of side-effects) because the chances are reduced of the unvaccinated contracting the disease (herd immunity) and the costs are reduced for care and lost productivity when someone does. In short, there is a precedent and a tradition of relinquishing individual rights for the greater good.

In cyberspace, insecure machines can be attacked and co-opted to serve in armies of zombies, which then cause annoyance by sending spam or wreak havoc by participating in distributed denial-of-service attacks. All of us in cyberspace are put at risk when someone else's computer has been co-opted. The rights of computer owners to control what their computers execute thus comes with a responsibility—the responsibility not to execute malware that might pollute or infect other parts of cyberspace. Trusted computing helps to discharge that responsibility by transferring decision making to a presumably knowledgeable system designer from a likely naïve computer owner.

Trusted computing might not embody the best trade-off, but it does represent a plausible option in a world where you can't depend on everyone who operates a computer to do the right thing. In particular, trusted computing makes it possible to educate and depend on a relatively few software producers instead of the ever growing number of computer owner-operators, an argument about leverage—not technology. Overall, there has been disappointingly little discussion in the computer security community about assignments of rights and responsibilities. Much could be gained from formulating and evaluating security solutions in such terms, making assignments of rights and responsibilities in a way that best leverages the diverse capabilities of participating communities. □