Obstacles to Freedom and Privacy by Design

Rebecca N. Wright
AT&T Labs – Research
Florham Park, NJ, USA
rwright@research.att.com

`http://www.research.att.com/~rwright`

Abstract: In this note, we describe some obstacles to designing technology that inherently protects civil liberties. We also suggest ways to overcome these obstacles.

Technology that inherently protects civil liberties is a nice idea in theory. In practice, however, it may be difficult to achieve. For example, Cranor and Wright [1] show that it can be difficult for software designers to control, or even influence, the ways in which their systems are used. Nonetheless, it is clear that generally a system is more likely to protect freedom and privacy if it is designed with that goal in mind than if it is not.

The most obvious civil liberty to discuss in this context is that of privacy, because the current Internet is an environment much more conducive to large-scale privacy invasion than the physical world. There are also others such as free speech and anonymous speech. The question of which rights to protect is complicated by the fact that different individuals have different ideas about which liberties are important, different governments support different laws about what are protected civil rights, and both corporations and governments often wish to engage in activities (for example, for the purpose of marketing in the case of governments and law enforcement in the case of governments) that while perhaps not strictly violating civil rights laws, come as close as possible without doing so.

In order to achieve the goal of liberty-protecting software, it is not enough to simply design and implement a system. Users will not necessarily use a particular technology just because it exists, even if it is better in some way than they technology they do use. It is tempting to think that users will have their own best interests at heart, and that therefore they will eagerly use any system that helps support their own civil liberties. However, this does not always appear to be the case. In particular, users may not always understand or care about the sometimes complex issues involved, and in fact convenience is often the main factor in determining what systems a user chooses, and how they are used. Furthermore, consumers' decisions can often be influenced by advertising, and large companies typically have substantially more resources available to devote to advertising than non-profit or public interest groups.

To some extent, these questions arise now because of the emergence of the "wired world". That is, as the use of computer technology grows, and in particular, the use of computer technology in a global network that includes all individuals, de facto standards and policies are made by virtue of what is possible and customary in this network ([3] and [4] discuss this at length). A fundamental question to be addressed is who should decide what the public policy properties of the computer world should be, and what will empower them to implement their decisions? In particular, what are and what should be the roles of and interplay between government, non-profit public interest groups, voluntary industrial standards, market forces driven by end consumers, and the technology itself?

Some have proposed market-based approaches, such as a system in which a privacy-protecting Internet service provider (ISP) may charge more than an ISP that does not provide privacy. However, as Shapiro [3] argues, such market-based approaches make it possible to sell away one's civil liberties, and also create a situation where the wealthy are more likely to retain their liberties than the poor.

The kind of technology solutions we seek must be:
- convenient: if it is not easy to obtain and easy to use, it will not be used.
- fair: the technology should be equally available to anyone, and the protected liberties should be protected for all users.

- (eventually) backed by industry, either for direct commercial reasons, or in response to government and/or consumer pressure.

There are several tools that can be used to help the creation and deployment of liberty-protecting software. Among these are social tools, such as consumer/voter education, and technical tools, such as cryptography and open source software.

Education: Many of the issues involved in understanding how using particular technological systems affects civil liberties are complex and subtle. People need to be educated in order to have the ability to make the right choices. If enough educated consumers are willing to buy and use certain technologies, while avoiding others, this provides incentives for other companies to provide similar solutions. Similarly, as voters, educated consumers can pressure politicians to be aware of these issues and act accordingly. To avoid overwhelming individuals by requiring them to learn too much about the intricacies of particular systems, it would be possible to use “seal of approval” approaches, where consumers and voters must learn who provides such approval, as well as something about the general issues, but need not learn and understand all the details.

Cryptography: Cryptography is the main existing tool that can be used to protect privacy. Furthermore, a significant amount of cryptographic software already exists. Why has this software not been more successfully and widely deployed? There are two significant reasons. First, users have a difficult time understanding how to use these tools, even when the user interface is one that would “traditionally” be considered good. For example, Whitten and Tygar [5] performed an experimental case study of PGP 5.0; their results suggest that security requires a different usability standard than other types of consumer software. They hypothesize that user interface design for effective security remains an open problem.

Second is the problem of distributing and maintaining keys. Public key cryptography helps with the key distribution problem to a large degree, but leaves the problem of revocation. Certificate revocation lists (CRL’s) can be provided to solve this problem as part of a large-scale public-key infrastructure (PKI), but CRL’s do not work well in the absence of a large-scale PKI. In some sense, PGP [6] provides a “bottom-up” peer-to-peer building of a PKI, which can also make use of PKI servers where they exist. In this spirit, Millen and Wright [2] have proposed “dependers”, an alternative to certificate revocation lists (CRL’s) that are more suited to a “bottom-up” building of a public key infrastructure than CRL’s.

Furthermore, note that while a straightforward application of cryptography can protect the privacy of data in transit from eavesdroppers, it does not prevent the receivers of that data from using it in unwanted ways. For example, in the case of a Web commerce server, encryption does not prevent the server from collecting information about a user’s buying habits and selling that information to others.

Open Source Software: Open source software has several properties that make it well-suited to supporting a low-resource grass-roots technology building effort, as well as to specifically supporting civil liberties. First, it makes it possible for a number of different individuals to build on each other’s software to create larger scale systems that no one of them would have been able to build alone. Second, it is possible (though technically difficult in most cases) for anyone to look at the software and try to check what properties are being provided by a system, rather than having to trust that the software providers’ claims are true. It also makes it possible to add modifications that put in missing privacy and other such features.

Note, however, that open source software is not without its drawbacks. Just as it can be modified to add in liberty-protecting features, it can be modified to remove them. This could be done either by the actual users, perhaps in the interest of efficiency, or by malicious parties who provide the unprotected version to users who may not know any better. However, most users are unlikely to modify their own software because of the inconvenience. Another problem with open source software, especially in the case of communications software, is that a proliferation of different versions makes it harder to maintain compatibility between them.

Depending on the resources available to a software designer, different ways to influence software usage are most likely to succeed. Cranor and Wright [1] discuss four methods of exerting influence, listed here in decreasing order of resources generally necessary for success: “hard-wiring” it so that it only can be used in certain ways, licensing it so that those who use it are legally obligated to use it in certain ways, issuing guidelines for how it should be used, or providing resources that make it easier to use the technology as the designers intended than to use it in any other way.

Accordingly, since at least initially, this effort will be a low-resource, grass roots effort, the best approach may be to provide resources such as toolkits, documentation, and (open source) reference implementations to help users and particularly other implementors to use, create, improve, and grow liberty-protecting software.

In addition, potential opposition by industry and/or government must be dealt with. In particular, at present Microsoft controls the market for operating systems and many applications. In fact, to a large degree, Microsoft essentially controls the way people use their computers. Users who choose not use to use Microsoft are often limited in their ability to communicate with other users. How can one either compete with Microsoft or get them and other technology companies to adopt liberty-protecting solutions? Coupled with the grass roots software-growing effort proposed above that will make liberty-protecting solutions available, there should be a grass roots political and educational campaign to create sufficient pressure on politicians and companies that the companies will create and adhere to voluntary industrial standards in order to avoid forced governmental regulation and create consumer goodwill that they can cash in on.

Acknowledgments

We thank Barbara Wright and Lorrie Cranor for helpful discussions.

References

[1] Lorrie F. Cranor and Rebecca N. Wright, "Influencing Software Usage", *these proceedings*. Also available at `http://xxx.lanl.gov/abs/cs.CY/9809018`.

[2] Jonathan K. Millen and Rebecca N. Wright, "Certificate Revocation the Responsible Way", *Proceedings of Computer Security, Dependability, and Assurance: From Needs to Solutions (CSDA'98)*, IEEE Computer Society, 1999, pp. 196--203. Available at `http://www.research.att.com/~rwright/csda98.ps`.

[3] Andrew L. Shapiro, *The Control Revolution: How The Internet is Putting Individuals in Charge and Changing the World We Know*, A Century Foundation Book, PublicAffairs, New York, 1999.

[4] Langdon Winner, "Do Artifacts Have Politics?", *Daedalus* 109/1 (Winter 1980), pp. 121-136.

[5] Alma Whitten and J. D. Tygar, "Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0", *Proceedings of Usenix Security Symposium*, 1999.

[6] Philip R. Zimmermann, *The Official PGP User's Guide*, MIT Press, 1995.